RESEARCH CENTRE

Inria Centre at Université Côte d'Azur

IN PARTNERSHIP WITH:

CNRS, Université Côte d'Azur

2023

ACTIVITY REPORT

Project-Team KAIROS

Multiform Logical Time for Formal Cyber-Physical System Design

IN COLLABORATION WITH: Laboratoire informatique, signaux systèmes de Sophia Antipolis (I3S)

DOMAIN

Algorithmics, Programming, Software and Architecture

THEME

Embedded and Real-time Systems

Inria

Contents

Project-Team KAIROS

Creation of the Project-Team: 2019 July 01

Keywords

Computer sciences and digital sciences

A1.1.1. – Multicore, Manycore

A1.1.2. – Hardware accelerators (GPGPU, FPGA, etc.)

A1.2.5. – Internet of things

A1.2.7. – Cyber-physical systems

A1.5.2. – Communicating systems

A2.2. – Compilation

A2.3. – Embedded and cyber-physical systems

A2.4. – Formal method for verification, reliability, certification

A2.5.1. – Software Architecture & Design

Other research topics and application domains

B5.1. – Factory of the future

B5.4. – Microelectronics

B6.1. – Software industry

B6.4. – Internet of things

B6.6. – Embedded systems

B6.7. – Computer Industry (harware, equipments...)

B7.2. – Smart travel

B8.1. – Smart building/home

B8.2. – Connected city

B9.5.1. – Computer science

1 Team members, visitors, external collaborators

Research Scientists

- Robert De Simone [Team leader, INRIA, Senior Researcher, HDR]
- Luigi Liquori [INRIA, Senior Researcher, HDR]
- Eric Madelaine [INRIA, Researcher, until Mar 2023, HDR]
- Dumitru Potop-Butucaru [INRIA, Researcher, HDR]

Faculty Members

- Julien Deantoni [UNIV COTE D'AZUR, Professor, HDR]
- Nicolas Ferry [UNIV COTE D'AZUR, Associate Professor]
- Frédéric Mallet [UNIV COTE D'AZUR, Professor, HDR]
- Marie-Agnès Peraldi-Frati [UNIV COTE D'AZUR, Associate Professor]
- Sid Touati [UNIV COTE D'AZUR, Professor, HDR]

Post-Doctoral Fellow

- Frédéric Fort [IRT ST EXUPERY, Post-Doctoral Fellow, until Oct 2023]

PhD Students

- Baptiste Allorant [INRIA]
- Joao Cambeiro [UNIV COTE D'AZUR, until Sep 2023]
- Barbara Da Silva Oliveira [UNIV COTE D'AZUR, from Oct 2023]
- Arseniy Gromovoy [UNIV COTE D'AZUR]
- Maksym Labzhaniia [UNIV COTE D'AZUR]
- Fabien Siron [KRONO SAFE (CIFRE)]
- Pavlo Tokariev [INRIA]
- Enlin Zhu [CNRS, until Sep 2023]

Technical Staff

- Barbara Da Silva Oliveira [CNRS, Engineer, from Aug 2023 until Sep 2023]
- Luc Hogie [CNRS]

Interns and Apprentices

- Antoine Fadda Rodriguez [IUT NICE, Intern, from Apr 2023 until Jun 2023]
- Antoine Maistre [IUT NICE, Intern, from Apr 2023 until Jun 2023]
- Alexis Rosset [IUT NICE, Intern, from Apr 2023 until Jun 2023]

Administrative Assistant

- Patricia Riveill [INRIA]

Visiting Scientists

- Joelle Despeyroux [INRIA]
- Tim Krauter [UNIV APPLIED SCIENCES WES NOR, from Feb 2023 until Jun 2023]

External Collaborator

- François Revest [UNIV COTE D'AZUR, until Apr 2023]

2 Overall objectives

The Kairos ambitions are to deal with the Design of Cyber-Physical Systems (CPS), at various stages, using Model-Based techniques and Formal Methods. Design here stands for co-modeling, co-simulation, formal verification and analysis activities, with connections both ways from models to code (synthesis and instrumentation for optimization). Formal analysis, in turn, concerns both functional and extra-functional correctness properties. Our goal is to link these design stages together, both vertically along the development cycle, and horizontally by considering the interactions between cyber/digital and physical models. These physical aspects comprise both physical environments and physical execution platform representations, which may become rather heterogeneous as in the cases of the Internet of Things (IoT) and computing at the edges of the gateways. The global resulting methodology can be tagged as Model-Based, Platform-Based CPS Design, see Figure 1.

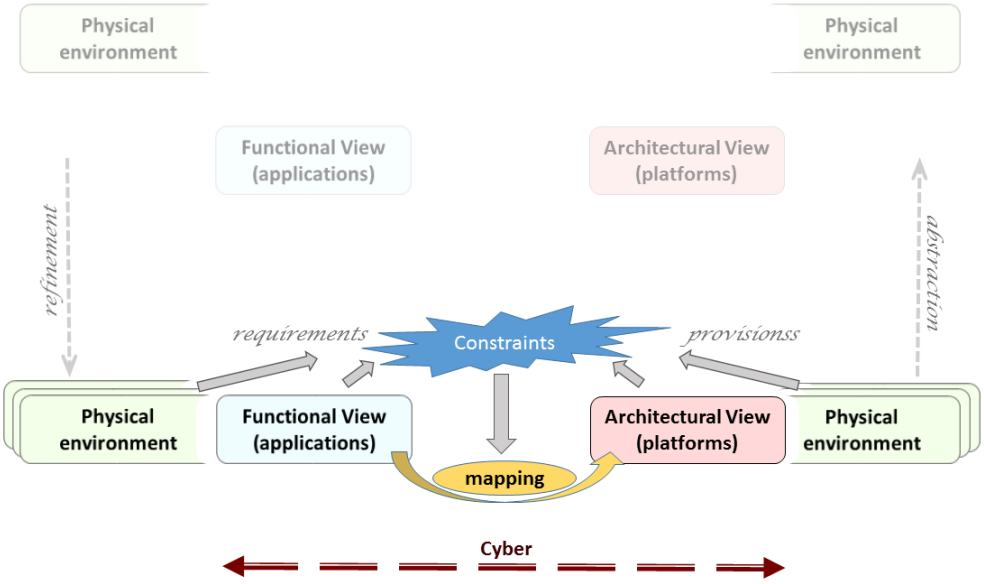

Figure 1: Cyber-Physical generic architectural features

CPS design must take into account all 3 aspects of application requirements, execution platform guarantees and contextual physical environment to establish both functional and temporal correctness. The general objective of Kairos is thus to contribute in the definition of a corresponding design methodology, based on formal Models of Computation for joint modeling of cyber and physical aspects, and using the

important central concept of Logical Time for expressing the requirements and guarantees that define CPS constraints.

Logical Multiform Time. It may be useful to provide an introduction and motivation for the notion of Logical Multiform Time (and Logical Clocks), as they play a central role in our approach to Design. We call Logical Clock any repetitive sequence of occurrences of an event (disregarding possible values carried by the event). It can be regularly linked to physical time (periodic), but not necessarily so: fancy processors may change speeds, simulation engine change time-integration steps, or much more generally one may react with event-driven triggers of complex logical nature (do this after 3-times that unless this...). It is our belief that user specifications are generally expressed using such notions, with only partial timing correlations between distinct logical clocks, so that the process of realization (or "model-based compilation") consists for part in establishing (by analysis or abstract simulation) the possible tighter relations between those clocks (unifying them from a partial order of local total orders to a global total order).

Kairos defined in the past a small language of primitives expressing recognized constraints structuring the relations between distinct logical clocks [1, 9]. This language (named CCSL for Clock Constraint Specification Language), borrows from notions of Synchronous Reactive Languages [11], Real-Time Scheduling Theory, and Concurrent Models of Computations and Communication (MoCCs) in Concurrency Theory [10] altogether. Corresponding extensions of Timed Models originally based on single (discrete or continuous) time can also be considered. Logical Time is used in our approach to express relation constraints between heterogeneous models, of cyber or physical origin, and to support analysis and co-simulation. Addressing cyber-physical systems demands to revisit logical time to deal with the multi-physical and sometimes uncertain environments.

Kairos is also active in Standardisation of the above mentioned Cyber-Physical Systems and Internet of Things research fields.

3 Research program

3.1 Cyber-Physical co-modeling

In Real-Time embedded systems, timing criticality imposes to take time predictivity as much into account as functional determinism, and this from the very beginning design phases onward. In addition, cyber-Physical System modeling requires joint representation of digital/cyber controllers and natural physics environments. Heterogeneous modeling must then be articulated to support accurate (co-)simulation, (co-)analysis, and (co-)verification, with multiple logical time sources and scales.

Figure 1 sketches the overall design framework. It comprises functional requirements, to be met provided surrounding platform guarantees, in a contract approach. All relevant aspects are modeled with proper Domain Specific Languages (DSL), so that constraints can be gathered globally, then analyzed to build a mapping proposal with both a structural aspect (functions allocated to platform resources), but also behavioral ones, scheduling activities. Mapping may be computed automatically or not, provably correct or not, obtained by static analytic methods or abstract execution.

Physical phenomena (in a very broad acceptance of the term) are usually modeled using continuous-time models and differential equations. Then the "proper" discretization opportunities for numerical simulation form a large spectrum of mathematical engineering practices. Note that is not at all the domain of expertise of Kairos members, but it should not be a limitation as long as one can assume a number of properties from the discretized version. On the other hand, we do have a strong expertise on modeling of both embedded processing architectures and embedded software (i.e., the kind of usually concurrent, sometimes distributed software that reacts to and control the physical environment). This is important as, unlike in the "physical" areas where modeling is common-place, modeling of software and programs is far from mainstream in the Software Engineering community. These domains are also an area of computer science where modeling, and even formal modeling, of the real objects that are originally of discrete/cyber nature, takes some importance with formal Models of Computation and Communications. It seems therefore quite natural to combine physical and cyber modeling in a more global design approach (even multi-physic domains and systems of systems possibly, but always with software-intensive aspects involved).

Our objective is certainly not to become experts in physical modeling and/or simulation process, but to retain from it only the essential and important aspects to include them into System-Level Engineering design, based on Model-Driven approaches allowing formal analysis.

This sets an original research agenda: Model-Based System Engineering environments exist, at various stages of maturity and specificity, in the academic and industrial worlds. Formal Methods and Verification/Certification techniques also exist, but generally in a point-wise fashion. Our approach aims at raising the level of formality describing relevant features of existing individual models, so that formal methods can have a greater general impact on usual, "industrial-level", modeling practices. Meanwhile, the relevance of formal methods is enhanced as it now covers various aspects in a uniform setting (timeliness, energy budget, dependability, safety/security...).

Directions on formalizing CPS should focus on the introduction of uncertainty (stochastic models) in our particular framework, on relations between (logical) real-time and security, and on accounting for resource discovery also in presence of mobility inherent to connected objects and Internet of Things [2].

3.2 Cyber-Physical co-simulation

The FMI standard (Functional Mock-Up Interface) has been proposed for "purely physical" (i.e., based on persistent signals) co-simulation, and then adopted in over 100 industrial tools including frameworks such as Matlab/Simulink and Ansys, to mention two famous model editors. With the recent use of co-simulation to cyber-physical systems, dealing with the discrete and transient nature of cyber systems became mandatory.

Together with other people from our community, we showed that FMI and other frameworks for co-simulation badly support co-simulation of cyber-physical systems; leading to bad accuracy and performances. More precisely, the way to interact with the different parts of the co-simulation require a specific knowledge about its internal semantics and the kind of data exposed (e.g., continuous, piecewise-constant). Towards a better co-simulation of cyber-physical systems, we are looking for conservative abstractions of the parts and formalisms that aim to describe the functional and temporal constraints that are required to bind several simulation models together.

3.3 Formal analysis and verification

Because the nature of our constraints is specific, we want to adjust verification methods to the goals and expressiveness of our modeling approach [13]. Quantitative (interval) timing conditions on physical models combined with (discrete) cyber modes suggest the use of SMT (Satisfiability Modulo Theories) automatic solvers, but the natural expressiveness requested (as for instance in our CCSL constructs) shows this is not always feasible. Either interactive proofs, or suboptimal solutions (essentially resulting of abstract run-time simulations) should be considered.

Complementarily to these approaches, we are experimenting with new variants of symbolic behavioural semantics, allowing to construct finite representations of the behaviour of CPS systems with explicit handling of data, time, or other non-functional aspects [4].

3.4 Relation between model and code

While models considered in Kairos can also be considered as executable specifications (through abstract simulation schemes), they can also lead to code synthesis and deployment. Conversely, code execution of smaller, elementary software components can lead to performance estimation enriching the models before global mapping optimization [3].

CPS introduce new challenging problems for code performance stability. Indeed, two additional factors for performance variability appear, which were not present in classical embedded systems: 1) variable and continuous data input from the physical world and 2) variable underlying hardware platform. For the first factor, CPS software must be analyzed in conjunction with its data input coming from the physics, so the variability of the performance may come from the various data. For the second factor, the underlying hardware of the CPS may change during the time (new computing actors appear or disappear, some actors can be reconfigured during execution). The new challenge is to understand how these factors

influence performance variability exactly, and how to provide solutions to reduce it or to model it. The modeling of performance variability becomes a new input.

3.5 Code generation and optimization

A significant part of CPS design happens at model level, through activities such as model construction, analysis, or verification. However, in most cases the objective of the design process is implementation. We mostly consider the implementation problem in the context of embedded, real-time, or edge computing applications, which are subject to stringent performance, embedding, and safety *non-functional requirements*.

The implementation of such systems usually involves a mix of synthesis—(real-time) scheduling, code generation, compilation—and performance (*e.g.* timing) analysis, as introduced in [7]. One key difficulty here is that synthesis and performance analysis depend on each other. As enumerating the various solutions is not possible for complexity reasons, heuristic implementation methods are needed in all cases. One popular solution here is to build the system first using unsafe performance estimations for its components, and then check system *schedulability* through a global analysis. Another solution is to use safe, over-approximated performance estimations and perform their mapping in a way that ensures by construction the schedulability of the system.

In both cases, the level of specification for the compound design space -including functional application, execution platform, extra-functional requirements, implementation representation is a key problem. Another problem is the definition of scalable and efficient mapping methods based on both "exact" approaches (ILP/SMT/CP solving) and compilation-like heuristics.

3.6 Extensions for spatio-temporal modeling and mobile systems

While Time is clearly a primary ingredient in the proper design of CPS systems, in some cases Space, and related notions of local proximity or conversely long distance, play also a key role for correct modeling, often in part because of the constraints this puts on interactions and time for communications. Once space is taken into account, one has to recognize also that many systems will request to consider mobility, originated as change of location through time. Mobile CPSs (or mCPS) occur casually in real-life, e.g., in the case of Intelligent Transportation Systems, or roaming connected objects of the IoT.

Spatio-temporal and mobility modeling may each lead to dynamicity in the representation of constraints, with the creation-deletion-discovering of new components in the system. This opportunity for new expressiveness will certainly cause new needs in handling constraint systems and topological graph locations. The new challenge is to provide an algebraic support with a constraint description language that could be as simple and expressive as possible, and of use in the semantic annotations for mobile CPS design. We also aim to provide fully distributed routing protocols to manage Semantic Resource Discovery in IoT and to standardize it.

3.7 Towards foundations for synchronous algorithmic & proof languages

The challenge is to find new process algebras with related type systems able to specify dynamic (operational semantics) and static (type systems) semantics of synchronous languages and algorithms. Adding types to such algebra would allow to certify them through Interactive Theorem Provers (ITP).

Luigi Liquori holds expertise in working in and extending the Edinburgh Logical Framework (LF), itself the basis of almost all ITPs [5]. New notions of *synchronizing proofs* could be introduced in LF to cope with these timed process algebras, and so help reasoning on (future evolutions of?) synchronous programming languages.

3.8 IoT and CPS standardization

Under the shield of the ETSI standardization consortium, we study protocols, models and performances of widely used CPS, mCPS and IoT frameworks and protocols: we focus on oneM2M ETSI interoperability standards in many aspects, see relevant sections on application domains 4.4, new results 7.15 7.16, and contracts 9.3.2 9.3.2.

4 Application domains

4.1 Cyber-Physical and embedded system design

System Engineering for CPS systems requires combinations of models, methods, and tools owing to multiple fields, software and system engineering methodologies as well as various digitalization of physical models (such as "Things", in Internet of Things (IoT)). Such methods and tools can be academic prototypes or industry-strength offers from tool vendors, and prominent companies are defining design flow usages around them.

We have historical contacts with industrial and academic partners in the domains of avionics and embedded electronics (Airbus, Thales, Safran). We also have new collaborations in the fields of satellites (Thales_Alenia_Space) and connected cars driving autonomously (Renault Software Factory). These provide us with current use cases and new issues in CPS co-modeling and co-design (Digital Twins) further described in New Results section. The purpose here is to insert our formal methods into existing design flows, to augment their analysis power where and when possible.

4.2 Safe driving rules for automated driving

Self-driving cars will have to respect roughly the same safety-driving rules as currently observed by human drivers (and more). These rules may be expressed syntactically by temporal constraints (requirements and provisions) applied to the various meaningful events generated as vehicles interact with traffic signs, obstacles and other vehicles, distracted drivers and so on. We feel our formalisms based on Multiform Logical Time to be well suited to this aim, and follow this track in several collaborative projects with automotive industrial partners. This domain is an incentive to increase the expressiveness of our language and test the scalability of our analysis tools on real size data and scenari.

4.3 Smart contracts for IoT objects

In collaboration with local industrial and standardisation partners, we have considered Smart Contracts (SC), as a way to formally establish specification of behavioral system traces, applied to connected objects in an IoT environment.

The ANR project SIM (see contract section) is based on the definition of formal language to describe services for autonomous vehicles that would execute automatically based on the observation of what is happening on the vehicle or the driver. The key focus is on the design of a virtual passport for autonomous cars that would register the main events occurring on the car and would use them to operate automatic but trustworthy and reliable services.

In a distinct effort, we investigate ways to "cleanly" extend smart contract languages with the capacity of modification/update at run-time, for the dynamic correction of errors or vulnerabilities (see next subsection on standardization). This direction is based on former work by L. Liquori and coauthors [41, 36].

4.4 oneM2M and ETSI standards

We participate to several ETSI Standardisation Committees:

- Based on our skills in IoT Semantic Discovery Protocols [39, 38] and content-based network protocols [35, 37] we contribute to the SmartM2M standard enhancement.
- In the new TTF (Testing Task Force) 019 project we consider as performance evaluation, analysis, planning and deployment for some (but not all) oneM2M open source implementations.
- A new STF (Specialist Task Force) ETSI project will explore the requirements of Smart Contrats from the UE Data Act and the UE EIDAS2 proposal. We contribute in definition of Smart Contracts that support exchange of data and their remuneration, with compliance versus new European Digital Identity and Electronic Ledger frameworks as target.

- Following our previous standardization effort of *Asynchronous Contact Tracing* (ACT) [40] in the Covid19 virus context, we now extend to the standardization of a new Asynchronous Forecast Standard, still at ETSI. The ambitious goal of designing an IoT eHealth standard, that can enable multiple detection and surveillance platforms to communicate and cooperate together, requires the multidisciplinary interaction between public agencies and biotech companies.

5 Social and environmental responsibility

5.1 Footprint of research activities

- Julien DeAntoni and Frédéric Mallet are members of the I3S Working Group on new practical ways to measure and reduce the impact of our research activity on the environment.

6 New software, platforms, open data

6.1 New software

6.1.1 VerCors

Name: VERification of models for distributed communicating COmponants, with safety and Security

Keywords: Software Verification, Specification language, Model Checking

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.

Release Contributions: It includes integrated graphical editors for GCM component architecture descriptions, UML classes, interfaces, and state-machines. The user diagrams can be checked using the recently published validation rules from, then the corresponding GCM components can be executed using an automatic generation of the application ADL, and skeletons of Java files.

The experimental version (VerCors V4, 2019-2022) also includes algorithms for computing the symbolic semantics of Open Systems, and testing bisimulation equivalences in a compositional way, using symbolic methods based on the Z3 SMT engine.

URL: https://team.inria.fr/scale/software/vercors/

Contact: Eric Madelaine

Participants: Antonio Cansado, Bartlomiej Szejna, Eric Madelaine, Ludovic Henrio, Marcela Rivera, Nassim Jibai, Oleksandra Kulankhina, Siqi Li, Xudong Qin, Zechen Hou, Biyang Wang, Viacheslav Herasymov, Rabea Ameur-boulifa

Partner: East China Normal University Shanghai (ECNU)

6.1.2 TimeSquare

Keywords: Profil MARTE, Embedded systems, UML, IDM

Scientific Description: TimeSquare offers six main functionalities:

1) graphical and/or textual interactive specification of logical clocks and relative constraints between them, 2) definition and handling of user-defined clock constraint libraries, 3) automated simulation of concurrent behavior traces respecting such constraints, using a Boolean solver for consistent trace extraction, 4) call-back mechanisms for the traceability of results (animation

of models, display and interaction with waveform representations, generation of sequence diagrams...). 5) compilation to pure java code to enable embedding in non eclipse applications or to be integrated as a time and concurrency solver within an existing tool. 6) a generation of the whole state space of a specification (if finite of course) in order to enable model checking of temporal properties on it

Functional Description: TimeSquare is a software environment for the modeling and analysis of timing constraints in embedded systems. It relies specifically on the Time Model of the Marte UML profile, and more accurately on the associated Clock Constraint Specification Language (CCSL) for the expression of timing constraints.

URL: http://timesquare.inria.fr

Contact: Julien DeAntoni

Participants: Benoît Ferrero, Charles André, Frédéric Mallet, Julien DeAntoni, Nicolas Chleq

6.1.3 GEMOC Studio

Name: GEMOC Studio

Keywords: DSL, Language workbench, Model debugging

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

1) Melange, a tool-supported meta-language to modularly define executable modeling languages with execution functions and data, and to extend (EMF-based) existing modeling languages. 2) MoCCML, a tool-supported meta-language dedicated to the specification of a Model of Concurrency and Communication (MoCC) and its mapping to a specific abstract syntax and associated execution functions of a modeling language. 3) GEL, a tool-supported meta-language dedicated to the specification of the protocol between the execution functions and the MoCC to support the feedback of the data as well as the callback of other expected execution functions. 4) BCOoL, a tool-supported meta-language dedicated to the specification of language coordination patterns to automatically coordinates the execution of, possibly heterogeneous, models. 5) Monilog, an extension for monitoring and logging executable domain-specific models 6) Sirius Animator, an extension to the model editor designer Sirius to create graphical animators for executable modeling languages.

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

URL: http://gemoc.org/studio.html

Publications: hal-00850770, hal-01355391, hal-01609576, hal-01651801, hal-01152342, hal-03374955, hal-01614561, hal-01616154

Contact: Benoît Combemale

Participants: Didier Vojtisek, Dorian Leroy, Erwan Bousse, Fabien Coulon, Julien DeAntoni

Partners: IRIT, ENSTA, I3S, OBEO, Thales TRT

6.1.4 Lopht

Name: Logical to Physical Time Compiler

Keywords: Real time, Compilation

Scientific Description: The Lopht (Logical to Physical Time Compiler) has been designed as an implementation of the AAA methodology. Like SynDEx, Lopht relies on off-line allocation and scheduling techniques to allow real-time implementation of dataflow synchronous specifications onto multiprocessor systems. But there are several originality points: a stronger focus on efficiency, which results in the use of a compilation-like approach, a focus on novel target architectures (many-core chips and time-triggered embedded systems), and the possibility to handle multiple, complex non-functional requirements covering real-time (release dates and deadlines possibly different from period, major time frame, end-to-end flow constraints), ARINC 653 partitioning, the possibility to preempt or not each task, and finally SynDEx-like allocation.

Functional Description: Compilation of high-level embedded systems specifications into executable code for IMA/ARINC 653 avionics platforms. It ensures the functional and non-functional correctness of the generated code.

Contact: Dumitru Potop-Butucaru

Participants: Dumitru Potop-Butucaru, Manel Djemal, Thomas Carle, Zhen Zhang

6.1.5 LoPhT-manycore

Name: Logical to Physical Time compiler for many cores

Keywords: Real time, Compilation, Task scheduling, Automatic parallelization

Scientific Description: Lopht is a system-level compiler for embedded systems, whose objective is to fully automate the implementation process for certain classes of embedded systems. Like in a classical compiler (e.g. gcc), its input is formed of two objects. The first is a program providing a platform-indepedent description of the functionality to implement and of the non-functional requirements it must satisfy (e.g. real-time, partitioning). This is provided under the form of a data-flow synchronous program annotated with non-functional requirements. The second is a description of the implementation platform, defining the topology of the platform, the capacity of its elements, and possibly platform-dependent requirements (e.g. allocation).

From these inputs, Lopht produces all the C code and configuration information needed to allow compilation and execution on the physical target platform. Implementations are correct by construction. Resulting implementations are functionally correct and satisfy the non-functional requirements. Lopht-manycore is a version of Lopht targeting shared-memory many-core architectures.

The algorithmic core of Lopht-manycore is formed of timing analysis, allocation, scheduling, and code generation heuristics which rely on four fundamental choices. 1) A static (off-line) real-time scheduling approach where allocation and scheduling are represented using time tables (also known as scheduling or reservation tables). 2) Scalability, attained through the use of low-complexity heuristics for all synthesis and associated analysis steps. 3) Efficiency (of generated implementations) is attained through the use of precise representations of both functionality and the platform, which allow for fine-grain allocation of resources such as CPU, memory, and communication devices such as network-on-chip multiplexers. 4) Full automation, including that of the timing analysis phase.

The last point is characteristic to Lopht-manycore. Existing methods for schedulability analysis and real-time software synthesis assume the existence of a high-level timing characterization that hides much of the hardware complexity. For instance, a common hypothesis is that synchronization and interference costs are accounted for in the duration of computations. However, the high-level timing characterization is seldom (if ever) soundly derived from the properties of the platform

and the program. In practice, large margins (e.g. 100%) with little formal justification are added to computation durations to account for hidden hardware complexity. Lopht-manycore overcomes this limitation. Starting from the worst-case execution time (WCET) estimations of computation operations and from a precise and safe timing model of the platform, it maintains a precise timing accounting throughout the mapping process. To do this, timing accounting must take into account all details of allocation, scheduling, and code generation, which in turn must satisfy specific hypotheses.

Functional Description: Accepted input languages for functional specifications include dialects of Lustre such as Heptagon and Scade v4. To ensure the respect of real-time requirements, Lopht-manycore pilots the use of the worst-case execution time (WCET) analysis tool (aiT from AbsInt, https://www.absint.com/ait/index.htm). By doing this, and by using a precise timing model for the platform, Lopht-manycore eliminates the need to adjust the WCET values through the addition of margins to the WCET values that are usually both large and without formal safety guarantees. The output of Lopht-manycore is formed of all the multi-threaded C code and configuration information needed to allow compilation, linking/loading, and real-time execution on the target platform.

Contact: Dumitru Potop-Butucaru

Participants: Dumitru Potop-Butucaru, Keryan Didier

6.1.6 mlirlus

Name: Lustre-based reactive dialect for MLIR

Keywords: Machine learning, TensorFlow, MLIR, Reactive programming, Real time, Embedded systems, Compilers

Scientific Description: We are interested in the programming and compilation of reactive, real-time systems. More specifically, we would like to understand the fundamental principles common to general-purpose and synchronous languages—used to model reactive control systems—and from this to derive a compilation flow suitable for both high-performance and reactive aspects of a modern control application. To this end, we first identify the key operational mechanisms of synchronous languages that SSA does not cover: synchronization of computations with an external time base, cyclic I/O, and the semantic notion of absent value which allows the natural representation of variables whose initialization does not follow simple structural rules such as control flow dominance. Then, we show how the SSA form in its MLIR implementation can be seamlessly extended to cover these mechanisms, enabling the application of all SSA-based transformations and optimizations. We illustrate this on the representation and compilation of the Lustre dataflow synchronous language. Most notably, in the analysis and compilation of Lustre embedded into MLIR, the initialization-related static analysis and code generation aspects can be fully separated from memory allocation and causality aspects, the latter being covered by the existing dominance-based algorithms of MLIR/SSA, resulting in a high degree of conceptual and code reuse. Our work allows the specification of both computational and control aspects of high-performance real-time applications. It paves the way for the definition of more efficient design and implementation flows where real-time ressource allocation drives parallelization and optimization.

Functional Description: The Multi-Level Intermediate Representation (MLIR) is a new reusable and extensible compiler infrastructure distributed with LLVM. It stands at the core of the back-end of the TensorFlow Machine Learning framework. mlirlus extends MLIR with dialects allowing the representation of reactive control needed in embedded and real-time applications.

Release Contributions: First open-source and public version.

URL: https://github.com/dpotop/mlir-lus-public

Publication: hal-03043623

Authors: Hugo Pompougnac, Dumitru Potop-Butucaru

Contact: Dumitru Potop-Butucaru

Partner: Google

6.1.7 Idawi

Keywords: Java, Distributed computing, Web Services, Parallel computing, Component models, Software Components, P2P, Dynamic components, Internet of things, Distributed Applications

Functional Description: Idawi is a middleware for the development and experimentation of distributed applications for mobile multi-hop networks, such as the IoT, the Edge, Mobile Ad hoc Networks, etc. Its development was initially motivated by our need to deploy scientific applications in clusters of computers, in order to run large experimentation campaigns of graph algorithms.

Idawi is an innovative arrangement of many features found in existing tools into a fresh Open Source Java reference implementation.

Idawi defines applications elements as components organized into a multi-hop overlay network on top of TCP/UDP and SSH, to be able to communicate even in the presence of NATs and firewalls. In the usual use case, there will be only one component per device. But, in order to enable the simulation/emulation of large systems, components can deploy other components in their Java Virtual Machine (JVM) or in another JVM(s) in the same device.

Idawi proposes a structuring model of distributed applications, which then must conform to a specific Object-Oriented model in the style of SOA: it defines that components expose their functionality via services. Services hold data and implement functionality about the specific concern they are about. Functionality is then exposed via operations, which can be triggered remotely from anywhere in the component overlay.

The decentralized communication model of Idawi matches the very nature of mobile multi-hop networks. It defines that components communicate with each other via messages of bounded size. Messaging can be both synchronous (imperative) and asynchronous (reactive/event-driven). It is powered by a default routing scheme and APIs that are tailored to collective communication, so as to offer native support of parallel processing.

Idawi comes with a set of built-in fully decentralized services for automatized quick deployment/bootstrapping of components through SSH, interoperability through a REST-based web interface, service provisioning and discovery, overlay management, and many other system-level functionality.

URL: https://github.com/lhogie/idawi

Publications: hal-03863333, hal-03886521, hal-03562184

Contact: Luc Hogie

6.1.8 ACT

Name: Asynchronous Contact Tracing Framework

Keyword: Contact tracing

Scientific Description: Implementation of standard ETSI TS 103757

Functional Description: ACT consists in 3 modules: 1) an ETSI/oneM2M communication infrastructure, 2) a mobile application (android), and 3) a web application.

Release Contributions: First open-source and public version available on gitlab inria

URL: https://gitlab.inria.fr/act/act

Publications: hal-02989793, hal-02989404, hal-03127890, hal-03935906

Contact: Luigi Liquori

Participants: Luigi Liquori, Alessio Di Dio, Antoine Maistre, Antoine Fadda Rodriguez, Pascal Tempier, Enrico Scarrone

Partners: ETSI, Université Côte d'Azur (UCA)

7 New results

7.1 Decision process for Logical Time

Participants: Frédéric Mallet, Pavlo Tokariev, Robert de Simone.

We are progressing the study of efficient methods for (syntactic) model-based expressiveness and (semantic) analysis/verification of multiform Logical Time (LT) formalisms, centered around our original CCSL constraint language (with possibly relevant extensions/restrictions.

This is a topic of collaboration with Chinese partners, mostly in the context of our associated-team Plot4IoT (see collaboration section). This year we focused on automatic synthesis of *safe-timing* CCSL constraints [14], as well as the use of reinforcement learning methods for such synthesis [15].

In the continuation of his PhD thesis Pavlo Tokariev studied descriptions of imprecise logical clocks, with notions of logical jitter (accumulative or not), with the impact of such constructs on operational semantics and event-driven simulation sc

7.2 Formalizing and extending Smart Contracts languages with temporal and dynamic features

Participants: Frédéric Mallet, Enlin Zhu.

The PhD of Enlin Zhu is funded on the SIM ANR project (see section 9.4), started January 2021. We initially focused mainly on smart contracts and their acceptance by lawyers, but we decided afterwards to consider the distributed application as a whole, since the smart contract itself in isolation is not sufficient to have a global end-to-end control of the services.
The main result is a state-based modeling of the recovery mechanism used in distributed ledger applications. There our formalisms are very relevant to capture both state-dependent properties and (logical) time requirements. This is also a place where the actual implementation and the temporality of the orchestration has a direct impact on the function to be achieved.
It is indeed very good to show the actual benefit in our setting of the model-based approach, allowing to reason and conduct verification at model level, and then hope to synthetize correct-by-construction code. We mainly target the generation of the control part that puts into music the purely functional parts deployed either on smart contracts or on some of the actors of the distributed service (including oracles).
While this result has been submitted twice, it was not yet accepted by a reviewing committee.

7.3 CCSL extension to stochastic Logical Time

Participants: Robert de Simone, Arseniy Gromovoy.

During this second year of Arseniy Gromovoy PhD thesis, we tried to draw connections between stochastic extensions of our CCSL constraint language and formal models of Functional Safety Analysis. The idea is to combine an abstract modeling of the system predicted nominal behavior, then a superposed stochastic model of rare, faulty behaviors. The purpose of analysis is then to guarantee sufficient dependability probabilities for the compound global system, given hypothesis on the potential local failures (all expressed in logical time terms).

7.4 Safety rules for autonomous driving

Participants: Frédéric Mallet, Maksym Labzhaniia, Marie-Agnès Peraldi-Frati, Julien Deantoni, Robert De Simone.

We had addressed the topic of formal modeling of automotive driving Safety Rules in a previous PhD thesis. In the current PhD of Maksym Labzhaniia we are revisiting this language in the light of logical multi-dimensions, mostly time and space (linked together by speed).

7.5 Take-over between automated and human automotive driving

Participants: Robert de Simone, François Revest, Ankica Barisic, Julien Deantoni.

This work was conducted as part of PSPC project ADAVEC, see contracts section 9.5.

The objective of ADAVEC is to study disengagement principles, where driving responsibility may be handed over between either a human driver or an artificial intelligence. This requires the formalization of take-over rules and models, with environmental observations both of the outside road environment, but also the cockpit inside and the human driver state. The postdoctoral period of Ankica Barisic and the temporary research engineer François Revest worked on this project, integrating their results in a general demonstrator developed by the industrial partners [19, 26].

7.6 On the generation of compiler friendly C code from CCSL specification

Participant: Baptiste Allorant, Sid Touati, Frédéric Mallet.

The objective of this work was to produce efficient sequential C codes from a CCSL description. A current prototype tool, named ECOGEN, was completed and intensively tested. ECOGEN can run on four prevalent compilers: GNU gcc, INTEL icc, Clang/LLVM, as well as CompCert by Inria. In addition to the small benchmarks of CCSL at hand, we produce synthetic ones with random generation techniques as well. The next step would be to extend ECOGEN in order to generate tuned *parallel* C code.

Sadly, the PhD thesis was interrupted in 2023 after Baptiste Allorant resigned for private medical reasons. But his source code and experimental data have been correctly saved on INRIA github.

7.7 Behavioral semantics in Model Driven Engineering

Participants: Julien Deantoni, Joao Cambeiro.

In the last years, in collaboration with other academic researchers, we developed the GEMOC studio, a language workbench to develop new modeling or programming languages and their interactions, based on a formal concurrency model. We recently used it for platform based design [21]

Two new results emerged this year. First, we figured out that it may be difficult for a user to debug concurrency since depending on the systems, some interleaving are not of interest and make difficult the exploration of significant ones. From these experiments, we decided to built up upon the Debugger Adapter Protocol to allow concurrency debugging. It implies among other a place where *exploration strategies* can be defined by the user (the user being either the language designer or the system designer). The concurrency strategies hide the irrelevant interleaving to highlight the important ones [17]. These strategies are defined independently of the execution engine and implemented into the GEMOC studio. The companion webpage is accessible here.

Second, we investigated further the notion of fidelity of a model with respect to the (knowledge of) reality and its impact in digital twins modeling. It clarifies the notion of model substituability during the development and/or verification process [22]. This is specifically addressed in the PhD thesis of Joao Cambeiro, in which we aim to characterize the capacity of distinct models to address a common reality. Defense is expected in the next months.

7.8 Efficient parallelism in shared memory

Participant: Dumitru Potop Butucaru.

We progressed further the technology underlying the LoPhT tool 6.1.4, partly inside the ANR *Caotic* project launched this year. We were also granted a bilateral collaborative funding from Airbus for continued evaluation of LoPhT in industrial context.

7.9 A language and compiler for embedded and Real-Time Machine Learning programming

Participants: Dumitru Potop Butucaru.

This research axis connects three research and engineering fields: Real-Time/Embedded programming, High-Performance Compilation, and Machine Learning. Initiated with the PhD thesis of Hugo Pompougnac (2022), it was extended this year with new exciting ideas allowing to add training features to the previous inference modeling in MLIR internal format representation extended with synchronous notions. This work was conducted with researchers from Google (including Albert Cohen, formerly DR Inria) [31].

The Static Single Assignment (SSA) form has proven an extremely useful tool in the hands of compiler builders. First introduced as an intermediate representation (IR) meant to facilitate optimizations, it became a staple of optimizing compilers. More recently, its semantic properties[1] established it as a sound basis for High-Performance Computing (HPC) compilation frameworks such as MLIR, where different abstraction levels of the same application[2] share the structural and semantic principles of SSA, allowing them to co-exist while being subject to common analysis and optimization passes (in addition to specialized ones).

But while compilation frameworks such as MLIR concentrate the existing know-how in HPC compilation for virtually every execution platform, they lack a key ingredient needed in the high-performance embedded systems of the future—the ability to represent reactive control and real-time aspects of a system. They do not provide first-class representation and reasoning for systems with a cyclic execution model, synchronization with external time references (logical or physical), synchronization with other systems, tasks and I/O with multiple periods and execution modes.

[1] Functional determinism while still allowing for limited concurrency.

[2] Ranging from ML dataflow graphs and linear algebra specifications down to affine loop nests and optimized (tiled, vectorized,...) low-level code.

And yet, while the standard SSA form does not cover these aspects, it shares strong structural and semantic ties with one of the main programming models for reactive real-time systems: dataflow synchrony, and its large and structured corpus of theory and practice of RTE systems design.

Relying on this syntactic and semantic proximity, we have extended the SSA-based MLIR framework to open it to synchronous reactive programming of real-time applications. We illustrated the expressiveness of our extension through the compilation of the pure dataflow core of the Lustre language (see software section for mlirlus 6.1.6). This allowed us to model and compile all data processing, computational and reactive control aspects of a signal processing application[3]. In the compilation of Lustre (as embedded in MLIR), following an initial normalization phase, all data type verification, buffer synthesis, and causality analysis can be handled using existing MLIR SSA algorithms. Only the initialization analysis specific to the synchronous model (a.k.a. clock calculus or analysis) requires specific handling during analysis and code generation phases, leading to significant code reuse.

The MLIR embedding of Lustre is non-trivial. As modularity based on function calls is no longer natural due to the cyclic execution model, we introduced a node instantiation mechanism. We also generalized the usage of the special undefined/absent value in SSA semantics and in low-level intermediate representations such as LLVM IR. We clarified its semantics and strongly linked it to the notion of absence and the related static analyses (clock calculi) of synchronous languages.

Our extension remains fully compatible with SSA analysis and code transformation algorithms. It allows giving semantics and an implementation to all correct SSA specifications. It also supports static analyses determining correctness from a synchronous semantics point of view.

An ANR project proposal has been submitted, and we participate to the `Confiance.AI` national initiative (in project EC7 - embedded AI) continues.

7.10 Formal verification of Logical Execution Time applications

Participants: Robert de Simone, Dumitru Potop Butucaru, Fabien Siron.

Fabien Siron defended his PhD on this topic in December 2023 [28].

The work, that was started by providing full operational semantics as a synchronous Logical Execution Time formalism (sLET) to the PsyC language [32], was satisfactorily concluded by applying it for model-checking automatic verification, with effective tool support and experimental evaluation on case study [25].

The thesis was funded on a CIFRE contract with the Krono-Safe company, now renamed Asterios Technologies after its acquisition by Safran. The proof-of-concept prototype should be integrated to the PsyC toolsuite.

More checkers targeted were NuXMV, Prover, Uppaaal and Kind2. Translators for interfacing were devised.

7.11 Interference analysis in Real-Time task mapping

Participants: Julien Deantoni, Robert de Simone.

In this work, funded by the *Institut de Recherche Technologique (IRT) Saint-Exupery*, and in connection with local R&D labs of Renault Software Factory and Thales Alenia Space, we consider model-based formal engineering of Interface Description Languages (IDL) to support logical time annotations and abstract temporized functional behavior representations. A very pragmatic simulator has been developped. In this simulator, applications are characterized by a temporal profil of their resource usage, the resources are characterized by the number of simultaneous access they support and the result is an evaluation of time spent waiting for a resource for each application.

[3] A pitch tuning vocoder.

7.12 Trustworthy Fleet deployment and management

Participants: Nicolas Ferry, Marie-Agnès Peraldi Frati, Julien Deantoni.

This activity is a follow-up of Nicolas Ferry previous activities in ENACT H2020 project, now renewed since his arrival in Kairos in the TrustFleet PHC 9.3.2 and DYNABIC 9.3.1 HEU projects. Continuous and automatic software deployment is still an open question for IoT systems, especially at the Edge and IoT ends. The state-of-the-art Infrastructure as Code (IaC) solutions are established on a clear specification about which part of the software goes to which types of resources. This is based on the assumption that, in the Cloud, one can always obtain the exact computing resources as required. However, this assumption is not valid on the Edge and IoT levels. In production, IoT systems typically contain hundreds or thousands of heterogeneous and distributed devices (also known as a *fleet of IoT/Edge devices*), each of which has a unique context, and whose connectivity and quality are not always guaranteed. In ENACT, we both investigated the challenge of automating the deployment of software on heterogeneous devices and of managing variants of the software which fit different types or contexts of Edge and IoT devices in the fleet [24]. In 2022, GeneSIS (a part of the aforementioned results) was recognized by the EC innovation radar as highly innovant with high maturity level. The natural next step is to investigate how to guarantee the trustworthiness of the deployment when (i) the quality of the devices is not guaranteed, (ii) the context of each device is continuously changing in an unanticipated manner, and (iii) software components are frequently evolving in the whole software stack of each device. In such context, ensuring the proper ordering and synchronization of the deployment actions is critical to improve the quality and trustworthiness and to minimize the downtime.

7.13 Model-Based serverless platform for the Cloud-Edge-IoT continuum

Participants: Nicolas Ferry, Barbara de Oliveira.

One of the most prominent implementation of the serverless programming model is Function-as-a-Service (FaaS). Using FaaS, application developers provide source code of serverless functions, typically describing only parts of a larger application, and define triggers for executing these functions on infrastructure components managed by the FaaS provider. The event-based nature of the FaaS model is a great fit for Internet of Things (IoT) event and data processing. However, there are still challenges that hinder the proper adoption of the FaaS model on the whole IoT-Edge-Cloud continuum. These include (i) vendor lock-in, (ii) the need to deploy and adapt serverless functions as well as their supporting services and software stack to the cyber-physical context in which they will execute, and (iii) the proper orchestration and scheduling of serverless functions deployed on the whole continuum. These observations are being confirmed by our ongoing systematic literature review initiated as part of the TrustFleet PHC aurora project in collaboration with SINTEF (Oslo, Norway), and performed as part of Barbara de Oliveira internship. In parallel, and to address the aforementioned challenges we initiated a first prototyping platform for the design, deployment, as well as maintenance of applications over the IoT-Edge-Cloud continuum. In particular, our platform enables the specification and deployment of serverless functions on Cloud and Edge resources as well as the deployment of their supporting services and software stack over the whole IoT-Edge-Cloud continuum. Next step will be to investigate solutions for the proper and timely orchestration and scheduling of these functions.

7.14 Behavioral equivalence of Open Systems

Participant: Eric Madelaine.

This work is conducted in part jointly with Rabea Ameur-Boulifa, associate professor at Telecom Paris. We consider Open (concurrent) Systems where the environment is represented as a number of processes whose behavior is unspecified. Defining their behavioral semantics and equivalences from a Model-Based Design perspective naturally implies model transformations. To be proven correct, they require equivalence of "Open" terms, in which some individual component models may be omitted. Such models take into account various kind of data parameters, including, but not limited to, time.This year we focused on notions of refinements for Open Terms [29, 18].

A formal framework is developed, as well as related effective tool set 6.1.1, for the compositional analysis of such programs.

After the retirement of Eric Madelaine, activities are still carrying on at Telecom Paris.

7.15 Performance evaluation in ETSI oneM2M standard

Participants: Luigi Liquori, Marie-Agnès Peraldi Frati.

In the context of the ETSI Testing Task Force (TTF 019) project, we studied performance evaluation of some (but not all) open source implementations of the oneM2M IoT interoperability standard, such as OM2M, OCEAN and ACME.

We first setup few cases studies [33] representating paradigmatic IoT/ CPS systems to be simulated/deployed and finally measured on the basis on a selected choices of KPIs. Then, we contribute with a multi-level model of a oneM2M system including the application behaviour, the oneM2M platform and services, and the oneM2M implementation [34]. This model will be injected and measured in the discrete event simulator OMNeT++.

7.16 Raising a contact tracing standard to a forecast standard

Participants: Luigi Liquori, Antoine Fadda Rodriguez, Antoine Maistre, Alessio Di Dio, Pascal Tempier, Enrico Scarrone.

In the recent past, we standardized [40] a novel contact tracing protocol, called Asynchronous Contact Tracing (ACT). ACT traces the presence of Covid19 virus via the IoT connected sensors and makes those informations available anonymously. This year we extended ACT in order to add a global detection and inter-communication IoT network focused on rapid response to bio-emergencies. This would potentially change the current services already deployed by the National Public Health Institutes to monitor, alert and advise political decision makers, public and military services, environmental institutes and citizens. This extension could promote an ETSI European standard capable of a real-time "forecasting" of any kind of pathogens and pollutions to people in the EU space, in an anonymous, resilient, and secure way; the standard should be sufficiently flexible and customizable nation-by-nation, according to their peculiar laws.

Sources of the web application (front-end and back-end) and the running android app are fully available on gitlab inria, while the web front-end can be runned by now on act.inria.fr. A rich documentation is also available. We are currently in the process of standardizing the extended protocol at the ETSI TC SmartM2M, and we are actively searching for fully fledged realistic big open-data sets in order to stress our proof of concept and our oneM2M architecture on different use cases. We also envisage to scale up the application to a bigger TRL scale by looking for internal (Inria) and external funding.

7.17 Strong priority and determinacy in timed CCS

Participants: Luigi Liquori, Michael Mendler, Robert De Simone.

Building on the classical theory of process algebra with priorities, we identify a new scheduling mechanism, called *sequentially constructive reduction* which is designed to capture the essence of synchronous programming. The distinctive property of this evaluation strategy is to achieve determinism-by-construction for multi-cast concurrent communication. In particular, it permits us to model shared memory multi-threading with reaction to absence as it lies at the core of the programming language Esterel.

In the technical setting of CCS extended by clocks and priorities, we prove for a large class of processes, which we call *structurally coherent* the Church-Rosser confluence property for constructive reductions. We further show that under some syntactic restrictions, called *pivotable* the operators of prefix, summation, parallel composition, restriction and hiding preserve structural coherence. This covers a strictly larger class of processes compared to those that are confluent in Milner's classical theory of CCS without priorities [30].

7.18 A middleware for the experimentation on IoT mobile networks

Participant: Luc Hogie.

In the context of studies on decentralized algorithms from mobile dynamic networks, we investigated the state of the art of the experimentation tools. We discovered that existing solutions, either coming from labs or companies, do not match the requirements of experimentation as it is usually done by Researchers. Indeed commercial products focus on reliability and interoperability at the expense of versatility, while lab tools most often serve as proof of concepts. The experimental study of algorithms requires the availability, in a single solution, of the following features: support for both synchronous and asynchronous communication, simulation/emulation of large systems, fast deployment, Web interoperability, and full decentralization, just to name a few. Idawi was designed and implemented to the very purpose of providing the Research community with a tool tailored to its needs [20]. See also the Software section of this report 6.1.7.

8 Bilateral contracts and grants with industry

Participants: Robert de Simone, Frédéric Mallet, Julien Deantoni, Dumitru Potop Butucaru, Marie-Agnès Peraldi Frati, Frédéric Fort.

Airbus. This collaboration provided us a funding grant for the extension of the Real-Time Systems Compilation method to allow parallelization onto multi-cores with classical ARM or POWER architecture. See Section 7.8 for results in 2024

IRT Archeocs. This collaboration deals with identification of real-time interferences in the context of concurrent tasks mapped to a common multicore architecture. Industrial partners come from both the Toulouse and Sophia areas, and we have special links with local Thales_Alenia_Space and Renault Software Labs. Frédéric Fort was hired in October 2022 as postdoc to conduct work under our supervision. See 7.11 for technical content.

Krono-Safe. The collaboration on the CIFRE PhD of Fabien Sironcame with financial back-up support. Technical results are reported in 7.10.

9 Partnerships and cooperations

Participants: Robert de Simone, Frédéric Mallet, Dumitru Potop-Butucaru, Julien Deantoni, Nicolas Ferry, Marie-Agnès Peraldi Frati, Luigi Liquori.

9.1 International initiatives

9.1.1 Inria Associate Teams not involved in an IIL or an international program

PLoT4IoT The Probabilistic Logical Time for Internet of Things associated team, together with the Key Laboratory of Trustworthy Computing of East China Normal University (ECNU) Shanghai, resumed its activities after 3 years stand-by due to CoVid confinement. We thus entered its second operational year (while calendar fifth). See the PLoT4IoT web site for details.

Some results are described in 7.1, while others are still unpublished. We also discussed potential registration of Chinese students in UniCA international Master programmes, in relation with our collaboration. There is hope to invite Chinese researchers for medium-size periods in 2024.

9.2 International research visitors

9.2.1 Visits of international scientists

ECNU delegation

Status ECNU delegation

Institution of origin: ECNU

Country: China

Dates: 1 week, December 2023

Context of the visit: workshop on the future collaboration program with ECNU (December 7th-8th)

Mobility program/type of mobility: funded by PLoT4IoT program

Michael Mendler

Status Professor

Institution of origin: University of Bamberg

Country: Germany

Dates: 1 months, Mai 2023

Context of the visit: working with Luigi Liquori on studying a deterministic process algebras accounting clocks and priorities. It gives also a talk at I3S

Mobility programming: funded by I3S/UniCA

Robert Harper

Status Professor

Institution of origin: Carnegie Mellon University

Country: USA

Dates: 1 week, October 2023

Context of the visit: invited by Luigi Liquori to give a talk to the Inria/UniCA Morgenstein Colloquium, presenting last advances on Logical Frameworks accounting resources

Mobility program: funded by I3S/UniCA and Inria

Tim Krauter

Status PhD student

Institution of origin: Bergen University

Country: Norway

Dates: 4 months, Feb to May 2023

Context of the visit: Mobility research period favored in the norwegian PhD system.

Mobility program: funded by his University.

9.2.2 Visits to international teams

Robert De Simone

Visited institution: ECNU

Country: China

Dates: 2 weeks, October 2023

Context of the visit: discussion and work with Prof. Jing Liu and her team

Mobility program/type of mobility: funded by PLoT4IoT program and partly by Chinese funding

Frédéric Mallet

Visited institution: ECNU

Country: China

Dates: 1 week, September 2023

Context of the visit: around the jubilee commemoration of Prof. He Jifeng 80th birthday, made an invited contribution.

Mobility program/type of mobility: funded by PLoT4IoT program

9.3 European initiatives

9.3.1 Horizon Europe

DYNABIC We participate to the DYNABIC HE project, jointly with I3S/UniCA Sparks team. DYNABIC stands for: *Dynamic business continuity of critical infrastructures on top of adaptive multi-level cybersecurity*. The project aims at delivering socio-technical methods, models and tools for resilience management. It will produce and validate a framework that enables system operators to forecast, assess and mitigate in real time business continuity risks and their possible cascading effects. Nicolas Ferry is WP7 leader and contributor to WP4 and WP5 (Critical infrastructure monitoring and security adaptation).

9.3.2 Other european programs/initiatives

PHC Aurora - TrustFleet project TrustFleet was a two-year PHC Aurora project, which concluded in December 2023. The project partners were SINTEF and the Kairos team. The main project results are the following. On the one side we investigated the latest state-of-the-art advances in the domain of software deployment over the Cloud-Edge-IoT Computing Continuum. In particular, we analyzed the latest trends and current limitations in the deployment of software functions (following the Function-as-a-Service principle) over Cloud, Edge and IoT infrastructures. On the other side, we investigated the novel concept of Digital Security Pair, a virtual security expert working in tandem with a SecDevOps team that overlooks the continuous security testing and improvement, on the fly and in parallel with the system development and operation, in particular for applications on the computing continuum. This specific aspect is an ongoing collaborative work, which will continue after the end of the project.

ETSI TTF 019 The TTF (Testing Task Force) 019 is funded by ETSI (2022-2024) and integrated within the Work Programme of the Technical Committee SmartM2M (TC SmartM2M). We received support to conduct the performance evaluation, analysis, planning and deployment for some (but not all) oneM2M open source initiatives. A systematic comparative study will be done to compare connectivity, interoperability, data management, security, and complex architecture issues.

ETSI STF 655 The STF (Specialist Task Force) 655 with ETSI members, is funded by ETSI (2023-2025) will explore the requirements of Smart Contrats according to the UE Data Act and the UE EIDAS2 proposal where Smart Contracts should support the exchange of data and their remuneration. Compliance with the new European Digital Identity and Electronic Ledger frameworks will be also addressed.

9.4 National initiatives

ANR Project SIM The ANR SIM (Smart IoT for Mobility) is a PRCE project co-funded by ANR (AAPG 2019) and DGA for 42 months. The national coordinator is the LEAT (UMR CNRS) and the other partners are Renault Software Labs and Symag. The goal is to provide a formal meta-language to describe Smart Contracts that can be used in the context of autonomous vehicles to provide services to the users. The services are related to the combined use of multi-model transportation systems by having a single Smart Contract that can enforce all the intermediate transactions with all the actors involved (car manufacturing, parking lease, highway toll companies, insurances, bike rental companies).

Competitivity Clusters The Kairos team is involved in the actions of the cluster SCS (Systèmes Communicants Sécurisés) and Frédéric Mallet is elected in the steering committee of SCS. One of the most prominent actions is to build, in partnership with Aix-Marseille University, a Digital Innovation Hub, to open the access (with actions of transfer and valorization) to Digital Innovations for companies that would benefit from it, like public institutions (hospitals, human resources, employment institutions) or private companies that could use IoT for agriculture, tourism, smart infrastructures (harbours, buildings, cities).

CNRS GDRs We are registered members of three GDR funded by CNRS : **SoC**^{2}, on topics of Hardware/Software codesign and Non-Functional Property modeling for co-simulation; **LTP**, on verification and language design for reactive CPS systems; **GPL**, on software engineering and Domain-Specific Languages.

Grand Défi - Confiance.AI We participate to the Confiance.AI programme, in the project "Embedded AI". It is the technological pillar of the Grand Défi “Securing, certifying and enhancing the reliability of systems based on artificial intelligence” launched by the Innovation Council. It is the largest technological research programme in the AIforHumanity plan, which is designed to make France one of the leading countries in artificial intelligence (AI). Our technical contributions to this project are described in 7.9.

9.5 Regional initiatives

PSPC ADAVEC This consortium, coordinated for UniCA and Inria by Amar Bouali, and with industrial partners the local companies EpicnPoc and Avisto, was intended to promote activities around automotive automated driving in Sophia-Antipolis science park. it was backed by Renault Research Factory and SCS cluster. Results for 2023 are described in 7.5.

10 Dissemination

Participants: Robert De Simone, Frédéric Mallet, Julien DeAntoni, Dumitru Potop-Butucaru, Luigi Liquori, Marie-Agnès Peraldi-Frati, Nicolas Ferry, Sid Touati.

10.1 Promoting scientific activities

10.1.1 Committes

Member of the standardisation committee

- Luigi Liquori is member of
  - ETSI TC SmartM2M - Smart Machine-2-Machine Communications and ETSI TC eHealth - ICT affecting the health sector.
  - oneM2M Concortium, hosted by ETSI - Global community developping standards for IoT.
  - ad hoc oneM2M Group on Academia Standardisation Relationship.
  - ISO groups ICS 35.60, Languages used in information technology ; “C and C++ liaison study group” of WG14 (C) and WG21 (C++).
  - ECMA, Industry association group for standardizing information and communication systems, Comité de Standardisation Technical Committee 39: Specifying JavaScript.
- Marie-Agnès Peraldi-Frati is member of the ETSI TC SmartM2M - Smart Machine-2-Machine Communications and oneM2M global partnership, hosted by ETSI - Global community developping standards for IoT.

Chair

- Luigi Liquori is Workshop Chair of the 9th International Conference on FSCD, Formal Structures for Computation and Deduction 2024 (co-located with ICALP and LICS).

Steering

- Robert de Simone was in the Steering Committee of the EmSoft conference, part of ESWeek federation.
- Luigi Liquori is elected member of the following steering committees: IFIP WG 1.6 Rewriting and of the International Conference on Formal Structures for Computation and Deduction (FSCD).

Conferences and workshops

- Luigi Liquori and Marie-Agnès Peraldi-Frati were members of the ETSI IoT Conference 2023.
- Nicolas Ferry was member of the following program committees: SCC’2023 - International workshop on Secure Cloud Continuum (co-located with CloudCom’23), STAM’23 - International workshop on Safety and Security Testing and Monitoring (co-located with ARES’23).
- Julien Deantoni was member of the following program committees: MPM4CPS’2023 - International workshop on Multi Paradigm Modeling for Cyber Physical Systems (co-located with Models’23), MeSS’23 - International workshop on MDE for Smart IoT Systems (co-located with STAF’23).

Organizing

- Nicolas Ferry organized MeSS’23 - International workshop on MDE for Smart IoT Systems (co-located with STAF’23).

10.1.2 Reviewing

Member of the editorial boards

- Julien Deantoni was guest editor for the ACM Transactions on Embedded Computing Systems (ACM TECS) Special Issue on Specification and Design Languages.
- Frédéric Mallet was editor of the Proceedings of the 22nd Day of AFADL [27].

Reviewer - reviewing activities

- Luigi Liquori was reviewer for the Logical Methods in Computer Science (LMCS) journal and for the international conferences Computer Science Logic (CSL) and of Foundations of Software Science and Computation Structures (FOSSACS).
- Nicolas Ferry was reviewer for the Software and Systems Modeling (SoSyM) journal.
- Julien Deantoni was reviewer for the Software and Systems Modeling (SoSyM) journal.

10.1.3 Invited talks

- Luigi Liquori. Raising the new Rewriting Web site. At IFIP Working Group 1.6 on Rewriting, Rome, Italy, 2023.

10.1.4 Scientific expertise

- Robert de Simone is jury member for the ANR CES25 project funding selection committee.
- Luigi Liquori is UE Cost funding selection committee member.

10.1.5 Research administration

- Frédéric Mallet is Director of the I3S UMR.
- Robert de Simone was member of the CSD (*Comité de Suivi Doctoral*) Inria committee until the end of 2023.
- Marie-Agnès Peraldi-Frati was member of the UniCA CAC (Conseil Académique).

10.2 Teaching - Supervision - Juries

10.2.1 Teaching

- Master: Dumitru Potop-Butucaru, A synchronous approach to the design of embedded real-time systems, 30h eq TD, EPITA Engineering School, Paris.
- Master: Dumitru Potop-Butucaru, Real-time embedded systems, 42h eq TD, EIDD, École d'Ingenieur Denis Diderot, Université Paris Cité.
- Master: Sid Touati, Architectures de processeurs hautes performances, 30 eq TD, Master 1 informatique, Université Côte d'Azur.
- Master: Sid Touati, Advanced operating systems, 30h eq TD, Master 1 informatique, Université Côte d'Azur.
- Master: Luigi Liquori, Peer-to-peer systems, 32h eq TD, M2, Polytech Nice Sophia, Université Côte d'Azur.
- Master: Julien Deantoni, Domain Specific Languages, 32h eq TD, M2, Polytech Nice Sophia, Université Côte d'Azur.

- Master: Julien Deantoni, Architecting IoT systems, Beyond Functional Correctness, 32h eq TD, Polytech Nice Sophia, Université Côte d'Azur.
- Master: Nicolas Ferry, Architecting IoT systems, Beyond Functional Correctness, 8h eq TD, Polytech Nice Sophia, Université Côte d'Azur.
- Master: Nicolas Ferry, Web services for the Internet of Things, 4h eq TD, M2 International Ubinet, Université Côte d'Azur.
- Master: Frédéric Mallet, Programmation Synchrone, 32h eq TD, M1, Université Côte d'Azur.
- International Master: Frédéric Mallet, Safety-Critical Systems, 32h eq TD, M1, Université Côte d'Azur.
- International Master: Frédéric Mallet, Software Engineering, 32h eq TD, M1, Université Côte d'Azur.
- License: Luc Hogie, Distributed programming, 28h eq TD, DUT Informatique, Université Côte d'Azur.
- Licence: Sid Touati, Architecture machine, 50 eq TD, L3 informatique, Université Côte d'Azur.
- Licence: Sid Touati, Compilation, 87h eq TD, L3 informatique, Université Côte d'Azur.
- Licence: Sid Touati, Systèmes d'exploitation, 18h eq TD, L2 informatique, Université Côte d'Azur.
- Licence: Marie-Agnès Peraldi Frati, Web security, 20h eq TD, Université Côte d'Azur.
- Licence: Marie-Agnès Peraldi Frati, Security of connected objects, 20h eq TD, Université Côte d'Azur.
- Licence: Marie-Agnès Peraldi Frati, IoT Infrastructure deployment, 20h eq TD, Université Côte d'Azur.
- Licence: Marie-Agnès Peraldi Frati, Large scale plateform for IoT, 20h eq TD, Université Côte d'Azur.
- Licence: Julien Deantoni, Introduction à l'informatique par le Web, 130h eq TD, DS4H portail science, Université Côte d'Azur.
- BUT1: Nicolas Ferry, Software Quality, 18h eq TD, Software Quality 2, 20h eq TD.
- BUT2: Nicolas Ferry, Software Architecture, 50h eq TD, IUT Nice Côte d'Azur, Université Côte d'Azur.
- BUT2: Nicolas Ferry, Web Programming, 30h eq TD, IUT Nice Côte d'Azur, Université Côte d'Azur.
- BUT3: Nicolas Ferry, Software Quality, 20h eq TD, Université Côte d'Azur.
- BUT3: Nicolas Ferry, Advanced Programming, 20h eq TD, Université Côte d'Azur.

10.2.2 Supervision

- Frédéric Mallet supervises the PhD theses of Pavlo Tokariev, Enlin Zhu, Maksym Labzhaniia, and co-supervises the PhD thesis of Baptiste Allorant.
- Sid Touati co-supervises the PhD thesis of Baptiste Allorant.
- Julien Deantoni supervises the PhD thesis of Joao Cambeiro, co-supervises the PhD thesis of Maksym Labzhaniia and co-supervises the PhD thesis of Barbara Da Silva Oliveira.
- Nicolas Ferry co-supervises the PhD thesis of Barbara Da Silva Oliveira.
- Robert de Simone supervises the PhD thesis of Arseniy Gromovoy, and co-supervised the PhD thesis of Fabien Siron.
- Dumitru Potop-Butucaru co-supervised the PhD thesis of Fabien Siron.
- Luigi Liquori supervises the PhD thesis of Mansur Khazeev. At this time, the thesis is suspended because of geo-political reasons (no visa for France).

10.2.3 Juries

- Robert de Simone was reviewer for the PhD thesis of Basile Pesin, Parkas Inria team, Paris (October).
- Julien Deantoni was examinator for the PhD thesis of Jayanth Krishnamurthy, Indes Inria team, Sophia Antipolis (June).
- Julien Deantoni was examinator for the PhD thesis of Yasmine Assioua, Institut polytechnique de Paris (April).

10.3 Popularization

10.3.1 Articles and contents

- Luigi Liquori published a popularization article on how research meet standardization [16].

11 Scientific production

11.1 Major publications

[1] C. André, J. Deantoni, F. Mallet and R. De Simone. 'The Time Model of Logical Clocks available in the OMG MARTE profile'. In: *Synthesis of Embedded Software: Frameworks and Methodologies for Correctness by Construction*. Ed. by S. K. Shukla and J.-P. Talpin. Chapter 7. Springer Science+Business Media, LLC 2010, July 2010, p. 28. URL: https://hal.inria.fr/inria-00495664.

[2] Y. Bao, M. Chen, Q. Zhu, T. Wei, F. Mallet and T. Zhou. 'Quantitative Performance Evaluation of Uncertainty-Aware Hybrid AADL Designs Using Statistical Model Checking'. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 36.12 (Dec. 2017), pp. 1989–2002. DOI: 10.1109/TCAD.2017.2681076. URL: https://hal.inria.fr/hal-01644285.

[3] T. Carle, D. Potop-Butucaru, Y. Sorel and D. Lesens. 'From Dataflow Specification to Multiprocessor Partitioned Time-triggered Real-time Implementation *'. In: *Leibniz Transactions on Embedded Systems* (Nov. 2015). DOI: 10.4230/LITES-v002-i002-a001. URL: https://hal.inria.fr/hal-01263994.

[4] L. Henrio, E. Madelaine and M. Zhang. 'A Theory for the Composition of Concurrent Processes'. In: *36th International Conference on Formal Techniques for Distributed Objects, Components, and Systems (FORTE)*. Ed. by E. Albert and I. Lanese. Vol. LNCS-9688. Formal Techniques for Distributed Objects, Components, and Systems. Heraklion, Greece, 2016, pp. 175–194. DOI: 10.1007/978-3-319-39570-8_12. URL: https://hal.inria.fr/hal-01432917.

[5] F. Honsell, L. Liquori, P. Maksimovic and I. Scagnetto. 'LLFP : A Logical Framework for modeling External Evidence, Side Conditions, and Proof Irrelevance using Monads'. In: *Logical Methods in Computer Science*. Special Issue in honor of Pierre Louis Curien (23rd Feb. 2017). URL: https://hal.inria.fr/hal-01146059.

[6] F. Honsell, L. Liquori, C. Stolze and I. Scagnetto. 'The Delta-framework'. In: 38th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, (FSTTCS) 2018. Vol. 122. 38th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS. Ahmedabad, India, 2018, 37:1–37:21. DOI: 10.4230/LIPIcs.FSTTCS.2018.37. URL: https://hal.archives-ouvertes.fr/hal-01701934.

[7] F. Jebali and D. P. Butucaru. 'Ensuring Consistency between Cycle-Accurate and Instruction Set Simulators'. In: *18th International Conference on Application of Concurrency to System Design, ACSD 2018, Bratislava, Slovakia, June 25-29, 2018*. 2018, pp. 105–114. DOI: 10.1109/ACSD.2018.00019. URL: https://doi.ieeecomputersociety.org/10.1109/ACSD.2018.00019.

[8] L. Liquori and C. Stolze. 'The Delta-calculus: syntax and types'. In: FSCD 2019 - 4th International Conference on Formal Structures for Computation and Deduction. Dortmund, Germany, 2019. DOI: 10.1007/978-3-030-44411-2_6. URL: https://hal.archives-ouvertes.fr/hal-01963662.

[9] F. Mallet and R. De Simone. 'Correctness issues on MARTE/CCSL constraints'. In: *Science of Computer Programming* 106 (Aug. 2015), pp. 78–92. DOI: 10.1016/j.scico.2015.03.001. URL: https://hal.inria.fr/hal-01257978.

[10] J.-V. Millo and R. De Simone. 'Periodic scheduling of marked graphs using balanced binary words'. In: *Theoretical Computer Science* 458.2 (Nov. 2012), pp. 113–130. DOI: 10.1016/j.tcs.2012.08.012. URL: https://hal.inria.fr/hal-00764076.

[11] D. Potop-Butucaru, R. De Simone and J.-P. Talpin. 'Synchronous hypothesis and polychronous languages'. In: *Embedded Systems Design and Verification*. Ed. by R. Zurawski. CRC Press, 2009, pp. 6-1-6–27. DOI: 10.1201/9781439807637.ch6. URL: https://hal.inria.fr/hal-00788473.

[12] C. Stolze and L. Liquori. 'A Type Checker for a Logical Framework with Union and Intersection Types'. In: FSCD 2020 - 5th International Conference on Formal Structures for Computation and Deduction. Paris, France, 2020. DOI: 10.4230/LIPIcs.FSCD.2020. URL: https://hal.archives-ouvertes.fr/hal-02573605.

[13] M. Zhang, F. Dai and F. Mallet. 'Periodic scheduling for MARTE/CCSL: Theory and practice'. In: *Science of Computer Programming* 154 (Mar. 2018), pp. 42–60. DOI: 10.1016/j.scico.2017.08.015. URL: https://hal.inria.fr/hal-01670450.

11.2 Publications of the year

International journals

[14] M. Hu, J. Xia, M. Zhang, X. Chen, F. Mallet and M. Chen. 'Automated Synthesis of Safe Timing Behaviors for Requirements Models using CCSL'. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (12th June 2023), pp. 1–1. DOI: 10.1109/TCAD.2023.3285412. URL: https://inria.hal.science/hal-04178061.

[15] M. Hu, M. Zhang, F. Mallet, X. Fu and M. Chen. 'Accelerating Reinforcement Learning-Based CCSL Specification Synthesis Using Curiosity-Driven Exploration'. In: *IEEE Transactions on Computers* 72.5 (1st May 2023), pp. 1431–1446. DOI: 10.1109/TC.2022.3197956. URL: https://inria.hal.science/hal-04178227.

[16] L. Liquori. 'How Research meets Standardization: the Asynchronous Contact Tracing ETSI Standard and the PANDESYS Research Activity'. In: *Enjoy ! the ETSI Mag* janvier (1st Jan. 2023). URL: https://inria.hal.science/hal-03940889.

[17] S. Zschaler, E. Bousse, J. Deantoni and B. Combemale. 'A Generic Framework for Representing and Analysing Model Concurrency'. In: *Software and Systems Modeling* 22 (2023), pp. 1319–1340. DOI: 10.1007/s10270-022-01073-2. URL: https://inria.hal.science/hal-03921704.

International peer-reviewed conferences

[18] R. Ameur-Boulifa, Q. Corradi, L. Henrio and E. Madelaine. 'Refinements for Open Automata'. In: *Lecture Notes in Computer Science*. SEFM 2023 - Software Engineering and Formal Methods. Vol. LNCS-14323. Software Engineering and Formal Methods 21st International Conference, SEFM 2023, Eindhoven, The Netherlands, November 6-10, 2023, Proceedings. Eindhoven, Netherlands: Springer Nature Switzerland, 31st Oct. 2023, pp. 11–29. DOI: 10.1007/978-3-031-47115-5_2. URL: https://inria.hal.science/hal-04271300.

[19] A. Barisic, P. Sigrist, S. Oliver, A. Sciarra and M. Winckler. 'Driver Model for Take-Over-Request in Autonomous Vehicles'. In: UMAP 2023 - 31st ACM Conference on User Modeling, Adaptation and Personalization. UMAP '23 Adjunct: Adjunct Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization. Limassol, Cyprus: ACM, 2023, pp. 317–324. DOI: 10.1145/3563359.3596994. URL: https://hal.science/hal-04205128.

[20] L. Hogie. 'A Decentralized Web Service Infrastructure for the Interoperability of Applications in Multihop Dynamic Networks'. In: CIoT 2023 - 6th Conference on Cloud and Internet of Things. Lisbon, Portugal: IEEE, 20th Mar. 2023, pp. 211–218. DOI: 10.1109/CIoT57267.2023.10084876. URL: https://hal.science/hal-04075895.

[21] J. Holtmann, J. Deantoni and M. Fockel. 'Early Timing Analysis based on Scenario Requirements and Platform Models (Extended Abstract)'. In: *Lectures Notes in Informatics*. Software Engineering 2023. Vol. 332. Software Engineering 2023 - kompletter Tagungsband. Paderborn, Germany: Gesellschaft für Informatik e.V., 2023, pp. 69–70. URL: https://inria.hal.science/hal-04259779.

[22] B. J. Oakes, C. Gomes, P. Gorm Larsen, J. Denil, J. Deantoni, J. Cambeiro and J. Fitzgerald. 'Examining model qualities and their impact on digital twins'. In: *IEEE Computer society digital library*. ANNSIM 2023 - Annual Modeling and Simulation Conference. Vol. 2023. 2023 Annual Modeling and Simulation Conference (ANNSIM) 5. Hamilton, Ontario, Canada, 26th June 2023, pp. 220–232. URL: https://inria.hal.science/hal-04259726.

[23] B. Oliveira, N. Ferry, H. Song, R. Dautov, A. Barišić and A. R. D. Rocha. 'Function-as-a-Service for the Cloud-to-Thing Continuum: A Systematic Mapping Study'. In: *Proceedings of the 8th International Conference on Internet of Things, Big Data and Security - IoTBDS*. IoTBDS 2023 - 8th International Conference on Internet of Things, Big Data and Security. Proceedings of the 8th International Conference on Internet of Things, Big Data and Security. Prague, Czech Republic, 21st Apr. 2023, pp. 82–93. DOI: 10.5220/0011982600003482. URL: https://hal.science/hal-04081180.

[24] E. Rios, E. Iturbe, A. Rego, N. Ferry, J.-Y. Tigli, S. Lavirotte, G. Rocher, P. Nguyen, H. Song, R. Dautov, W. Mallouli and A. R. Cavalli. 'The DYNABIC approach to resilience of critical infrastructures'. In: *ARES '23: Proceedings of the 18th International Conference on Availability, Reliability and Security*. ARES 2023 - 18th International Conference on Availability, Reliability and Security. Benevento, Italy: ACM, 29th Aug. 2023, p. 136. DOI: 10.1145/3600160.3605055. URL: https://hal.science/hal-04191589.

[25] F. Siron, D. Potop-Butucaru, R. de Simone, D. Chabrol and A. Methni. 'Semantics foundations of PsyC based on synchronous Logical Execution Time'. In: *ACM digital library*. CPS-IoT Week 2023 - Cyber-Physical Systems and Internet of Things Week 2023. Vol. 2023. CPS-IoT Week '23: Proceedings of Cyber-Physical Systems and Internet of Things Week 2023. San Antonio TX USA, France: ACM, 9th May 2023, pp. 319–324. DOI: 10.1145/3576914.3587495. URL: https://inria.hal.science/hal-04355453.

Scientific book chapters

[26] A. Barisic and M. Winckler. 'Towards User Profile Meta-Ontology'. In: *Human-Centered Software Engineering. HCSE 2023. Lecture Notes in Computer Science. LNCS*. 31st Oct. 2023. URL: https://hal.science/hal-04210148.

Edition (books, proceedings, special issue of a journal)

[27] N. Kushik and F. Mallet, eds. *Proceedings of the 22nd Day of AFADL*. Journées AFADL. 9th Aug. 2023, p. 60. URL: https://inria.hal.science/hal-04179353.

Doctoral dissertations and habilitation theses

[28] F. Siron. 'Methodology for the formal verification of temporal properties for real-time safety-critical applications based on logical time'. Université cote d'azur, 11th Dec. 2023. URL: https://inria.hal.science/tel-04355316.

Reports & preprints

[29] L. Henrio, E. Madelaine, R. Ameur-Boulifa and Q. Corradi. *Refinements for Open Automata (Extended Version)*. RR-9517. Inria - Research Centre Grenoble – Rhône-Alpes, 1st Sept. 2023. URL: https://inria.hal.science/hal-04193421.

[30] L. Liquori and M. Mendler. *Strong Priority and Determinacy in Timed CCS*. Inria, 30th Dec. 2023. URL: https://inria.hal.science/hal-04367635.

[31] D. Potop-Butucaru, A. Cohen, G. Plotkin and H. Pompougnac. *Bidirectional Reactive Programming for Machine Learning*. 2023. DOI: 10.48550/arXiv.2311.16977. URL: https://inria.hal.science/hal-04354071.

[32] F. Siron, D. Potop-Butucaru, R. de Simone, D. Chabrol and A. Methni. *Formal Semantics of the PsyC language*. RR-9506. Inria - Sophia Antipolis, May 2023, p. 32. URL: https://inria.hal.science/hal-04088177.

Other scientific publications

[33] B. Flynn, L. Liquori, M.-A. Peraldi-Frati, S. Medjiah and T. Monteil. *SmartM2M: Scenarios for evaluation of oneM2M deployments*. 1st Sept. 2023. URL: https://inria.hal.science/hal-04229449.

[34] S. Medjiah, T. Monteil, L. Liquori, M.-A. Peraldi-Frati and B. Flynn. *SmartM2M; Model for oneM2M Performance Evaluation*. 1st Aug. 2023. URL: https://inria.hal.science/hal-04229478.

11.3 Cited publications

[35] R. Chand, M. Cosnard and L. Liquori. 'Powerful Resource Discovery for Arigatoni Overlay Network'. In: *Future Generation Computer Systems*. Future Generation Computer Systems, FGCS 24.1 (Jan. 2008), pp. 31–48. DOI: 10.1016/j.future.2007.02.009. URL: https://inria.hal.science/hal-00909630.

[36] A. Ciaffaglione, P. Di Gianantonio, F. Honsell and L. Liquori. 'A prototype-based approach to object evolution'. In: *The Journal of Object Technology* 20.4 (Dec. 2020), pp. 1–24. URL: https://inria.hal.science/hal-01646168.

[37] L. Liquori, D. Borsetti, C. Casetti and C.-F. Chiasserini. 'An Overlay Architecture for Vehicular Networks'. In: *NETWORKING 2008. Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet 7th International IFIP-TC6 Networking Conference Singapore, May 5-9, 2008 Proceedings*. Ed. by A. Das, H. K. Pung, F. B. S. Lee and L. W. C. Wong. Vol. 4982. Lecture Notes in Computer Science. Singapore, Singapore: Springer Verlag, May 2008, pp. 60–71. DOI: 10.1007/978-3-540-79549-0_6. URL: https://inria.hal.science/hal-00909585.

[38] L. Liquori, M.-A. Peraldi-Frati, A. Cimmino, R. García Castro, A. Q. Khan, S. El Khatab, O. Khramov, E. Scarrone and J. Koss. *ETSI SmartM2M Technical Report 103716; oneM2M Discovery and Query solution(s) simulation and performance evaluation*. oneM2M has currently native discovery capabilities that work properly only if the search is related to specific known sources of information (e.g. searching for the values of a known set of containers) or if the discovery is well scoped and designed (e.g. the lights in a house). When oneM2M is used to discover wide sets of data or unknown sets of data, the functionality is typically integrated by ad hoc applications that are expanding the oneM2M functionality. This means that this core function may be implemented with different flavours and this is not optimal for interworking and interoperability.The objective of the present document [i.3] in conjunction with three other ones ETSI TR 103 714 [i.1], ETSITR 103 715 [i.2] and ETSI TR 103 717 [i.4] is the study and development of semantic Discovery and Query capabilities for oneM2M and its contribution to the oneM2M standard.The goal is to enable an easy and efficient discovery of information and a proper interworking with external source/consumers of information (e.g. a distributed data base in a smart city or in a firm), or to directly search information in the oneM2M system for big data purposes.A simulation phase is conducted in parallel and "circular" feedback with respect to the study phase, with the goal to provide a proof of concept, run suitable scenarios provided by previous phases and a performance evaluation to support the selection/development of the Discovery and Query solution. The simulator and the simulation results are documented in ETSI TR 103 716 [i.3] (the present document). An extract of the simulation results is included ETSI TR 103 715 [i.2] and ETSI TR 103 717 [i.4]. A selection of the use cases includes a set of oneM2M relevant configurations scenarios to be considered for the simulation activity described below. June 2021. URL: https://inria.hal.science/hal-03261059.

[39] L. Liquori, E. Scarrone, M.-A. Peraldi-Frati, S. M. Jeong, A. Cimmino, R. García Castro, J. Koss, A. Q. Khan, S. Kumar and S. El Khatab. *ETSI SmartM2M Technical Report 103715; Study for oneM2M; Discovery and Query solutions analysis & selection*. Ed. by ETSI. The oneM2M system has implemented basic native discovery capabilities. In order to enhance the semantic capabilities of the oneM2M architecture by providing solid contributions to the oneM2M standards, four Technical Reports have been developed. Each of them is the outcome of a special study phase: requirements, study, simulation and standardization phase. The present document covers the second phase and provides the basis for the other documents. It identifies, defines and analyses relevant approaches with respect to the use cases and requirements developed in ETSI TR 103 714 The most appropriate one will be selected. Jan. 2021. URL: https://inria.hal.science/hal-03115497.

[40] L. Liquori, E. Scarrone, S. Wood, L. Cees, F. Dasilva, M. Maass, F. Bob, T. Kessler, H. Taras and M. Vanetti. *ETSI Technical Specification TS 103757. SmartM2M; Asynchronous Contact Tracing System*. Ed. by ETSI. Dec. 2021. URL: https://hal.inria.fr/hal-02989793.

[41] L. Liquori and A. Spiwack. 'FeatherTrait: A Modest Extension of Featherweight Java'. In: *ACM Transactions on Programming Languages and Systems (TOPLAS)*. ACM Transactions on Programming Languages and Systems (TOPLAS) 30.2 (Mar. 2008), 11:1–11:32. DOI: 10.1145/1330017.1330022. URL: https://inria.hal.science/inria-00432538.