INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team Mosel

Proof-oriented development of computer-based systems

Nancy - Grand Est

THEME SYM

Table of contents

1. Team

Research Scientist

Stephan Merz [Research Director, INRIA, HdR]

Faculty Member

Dominique Méry [Team leader, Professor, University Henri Poincaré Nancy 1, HdR]
Dominique Cansell [Assistant Professor, University of Metz, HdR]
Pascal Fontaine [Assistant Professor, University Nancy 2]
Jacques Jaray [Professor, Institut National Polytechnique de Lorraine and École des Mines de Nancy, HdR]
Denis Roegel [Assistant Professor, University Nancy 2]

External Collaborator

Mahmoud BenHacine [Université de TUNIS, Tunis, Intern, February-August]
Paul Gibson [Institut National des Télécommunications Evry]
Rosemary Monahan [NUI Maynooth, Maynooth, Visitor, July]
Cristian Rosa [University Rosario, Argentina, Intern, May–July]
Pablo Speciale [University Rosario, Argentina, Intern, May–July]
Hehua Zhang [University Tsinghua, Beijing, PhD Student, June–November]

Technical Staff

Thomas Bouton

PhD Student

Sabina Akhtar [University Henri Poincaré Nancy 1, SFERE grant and complement from the Lorraine Region, since October 2008, defense planned for 2011]
Nazim Benaïssa [University Henri Poincaré Nancy 1, ANR project RIMEL, defense planned for 2009]
Diego Caminha Barbosa de Oliveira [University Nancy 2, CORDI-S contract, defense planned for 2010]
Loïc Fejoz [University Henri Poincaré Nancy 1, Microsoft ERO grant, defense planned for February 2009]
Joris Rehm [University Henri Poincaré Nancy 1, grant of French ministery of Higher Education and Research, defense planned for 2009]
Cristian Rosa [University Henri Poincaré Nancy 1, since November 2008, joint supervision with Martin Quinson of AlGorille team, ANR project UMSSimGrid, defense planned for 2011]
Mouna Saad [joint supervision, University of Tunis and Institut National Polytechnique de Lorraine, defense planned for 2010]
Neeraj Kumar Singh [University Henri Poincaré Nancy 1, since October 2008, grant of French ministery of Higher Education and Research, defense planned for 2011]

Administrative Assistant

Roxane Auclair [Administrative assistant]

2. Overall Objectives

2.1. Introduction

Proof-oriented system development focuses on formally describing and analyzing design models for computer-based systems. Because the descriptions are based on sound semantic models, they aim at ensuring higher levels of reliability and correctness. The MOSEL research team develops such concepts, and applies them, focusing on reactive, real-time, distributed, and mobile systems that may contain both hardware and software components. Key concepts in the approach advocated by our group are *refinement* and *(de-)composition* that support the development of complex systems across several layers of abstraction. Our work is structured along the following lines of research:

Foundations and methodology. The theoretical underpinnings of formal methods have been firmly established since several decades, and we refrain from developing completely new approaches. However, novel concepts of system design or novel application domains, including the security of computerized systems, mobile systems or hardware/software codesign require extensions and adaptations of existing formalisms (B and TLA$^+$ are the two main frameworks used by our group). Moreover, formal methods need to be integrated in standard industrial development cycles, requiring serious attention to the methodology of their application. For example, specifications and proofs represent proper artefacts of system design, and we engage in work on their representation, management, and reuse, based on composition and genericity.

Notation and tools. We are studying notations that aid system engineers for representing their concepts, and that integrate different methods and tools of system design. Where necessary, we also engage in developing support tools or—whenever possible—in interfacing existing tools to facilitate their use or support their application in novel contexts.

Applications. Industrial and academic case studies serve to validate our concepts and theories and lay the foundation for their transfer to use by practitioners in industry. They also force us to recognize deficiencies of our concepts, stimulating further theoretical advances and tool development. We are therefore maintaining active cooperations with partners in industry, namely ClearSy, Microsoft, and academia, including neighboring disciplines such as circuit design. We also use our methods in the courses we teach to evaluate their applicability.

2.2. Highlights of the year

- The project MOSEL is stopped on the 31. of December 2008 and has been fruitful for demonstrating that formal methods [33], [39] could be applied and that both abstractiona nd refinement [34] are two central concepts of computer science; the team MOSEL is still going ahead and will go further in the research started in the MOSEL EPI; this year has been very active in discussions for proposing a new project in the INRIA framework. We hope that the success will be on the way of the new promotor. Finally, the ANR RIMEL project provides other illustrative examples of the refinement technique, when developing distributed algorithms [34] like Mazurkiewicz's one.
- Springer Verlag published a book on Temporal Logic and State Systems of which Stephan Merz is a co-author [18].
- This year has seen a complete revision of our SMT solver haRVey, a new release of which is scheduled before the end of the year (see section 5.1).

3. Scientific Foundations

3.1. Foundations and Methodology

Keywords: *abstraction*, *composition*, *concurrency*, *distributed systems*, *formal methods*, *reactive systems*, *refinement*.

The MOSEL team investigates methods to develop provably correct computer-based systems. The class of systems we are interested in includes reactive, distributed, embedded, and mobile systems. In contrast to classical sequential algorithms that can be characterized in terms of their input-output relation, the correctness of such systems is described in terms of their executions (traces). The choice of an adequate formal language depends on which properties are of interest for a given system. For example, methods based on pre- and postconditions suffice for expressing and proving safety properties, while temporal logics can also express liveness.

We are particularly interested in processes and methodologies that underly system *development*, as opposed to the verification of an existing system a posteriori. This view is formally reflected by the notion of *refinement*, which ensures that descriptions produced in later stages of system design preserve earlier, more abstract descriptions; in particular, all properties proven earlier remain valid for the refined model. In this way, the effort of verification is spread over the entire development process, and this helps us to achieve a significant degree of automatisation in our verification efforts. Crucially, errors can be detected very early, when they are relatively cheap to correct. The formalisms that we are most familiar with are the B method due to Abrial [32], [31] and the Temporal Logic of Actions and the TLA$^+$ language introduced by Lamport [37]. Members of MOSEL have been invited to write tutorials on these methods [17], [20].

The second cornerstone of system development is composition and decomposition [30]. In effect, monolithic system development methods do not scale to realistic systems. Composition refers to the assembly of complex systems from independently developed, possibly pre-existing components. Dually, an entire system (or its specification) can be decomposed into separate subsystems that are then refined individually. Decomposition is a fundamental structuring principle of the event-based B method.

The contributions of the MOSEL team to the foundations of this area concern extensions of the semantic models for particular types of systems such as real-time, mobile or security-sensitive systems. We also study ways to make developments more easily reusable by focusing on generic theories and proofs that can later be instantiated for reuse.

3.2. Notation and tools

Keywords: *B*, *TLA*, *interface*, *model checking*, *proof obligations*, *theorem proving*, *tool integration*.

The development of provably correct systems [2] relies on languages with a precise, mathematically defined semantics, in which system specifications are written and proof obligations are stated. For all but toy systems, formal development methods generate a huge number of proof obligations, and highly automated tools become essential to successfully apply the methods. Whereas automated deduction has made substantial progress, each tool typically covers a restricted domain, and the combination of different tools is an active area of research.

We believe that beyond the sheer capacity of provers for carrying out deductions, adequate interfaces are an important element in order to promote their actual use in system development. Dominique Cansell has developed an interface for the interactive prover of Atelier B, the primary support tool for the B method, freely available for academic users within the B4Free tool. Its development is based on a thorough study of interactive provers are used for system verification.

Members of MOSEL are also developing the SMT solver haRVey (described in section 5.1), which can output proofs for certification by interactive proof assistants.

3.3. Applications

Keywords: *embedded systems*, *hardware-software codesign*, *security*, *services*, *telecommunications*, *transportation systems*.

A substantial part of our research is driven by work on concrete applications and case studies, as well as by courses that we teach. Several applications currently studied by MOSEL concern hardware-software codesign for embedded systems. Within these industrial projects, we have applied the B method to produce a series of models, starting from standard requirement documents as inputs for the abstract models. As a result of several refinement steps, with accompanying proofs, we were obtained models at a level of granularity that allowed us to mechanically synthesize hardware descriptions.

Another interesting field of study is the application of formal description techniques to problems of information security. We have worked on extending logics and formalisms that we are familiar with for the specification of access control requirements, and we have studied problems that arise in electronic voting.

4. Application Domains

4.1. Application Domains

Keywords: *critical systems*, *embedded systems*, *networks*, *protocols*, *telecommunications*.

Our work mainly targets critical systems whose malfunctioning may endanger the health or life of persons, their privacy and security, or that may lead to serious financial consequences. We enjoy working on concrete examples that are developed in the context of industrial or cooperative projects, including telecommunications, embedded systems, networks and their protocols, problems of information security, and mobile systems.

5. Software

5.1. The haRVey reasoner

Keywords: *SMT prover*, *arithmetic*, *combination of decision procedures*, *congruence closure*, *difference logic*, *proof trace*.

Participants: Thomas Bouton, Diego Caminha, Pascal Fontaine [correspondant].

haRVey is an SMT (Satisfiability Modulo Theories) prover. It was initiated in cooperation with Silvio Ranise and Christophe Ringeissen of the CASSIS project team of INRIA Lorraine. It is developed in cooperation with David Déharbe from the Federal University of Rio Grande do Norte in Natal, Brazil. haRVey can handle large quantifier-free formulas containing uninterpreted predicates and functions, and some arithmetics. It also has some support for user-defined theories, quantifiers, and lambda-expressions. This allows users to easily express properties about "higher-order" concepts involving sets, relations, etc. The prover can produce an explicit proof trace when it is used as a decision procedure for quantifier-free formulas with uninterpreted symbols. This feature has been used to integrate it with the proof assistant Isabelle.

Since 2006, the tool has been available at http://harvey.loria.fr. The main efforts in 2008 have gone into preparing a major new release of the tool, which will provide a better integration of the different available features, and a better support for arithmetic over rationals and integers. A public release is planned for the end of 2008, after extensive testing and improvement of the documentation.

The capabilities of the tool to handle arithmetic have been extended, with the implementation of a highly efficient decision procedure for difference logic. Its integration as a component in the overall tool required some theoretical foundations, and a deeper understanding of the cooperation process with non-convex theories (see section 6.3).

Future research and implementation efforts will be directed to furthermore extend the accepted language, increase the efficiency, and provide an optimal interface (including providing explicit proof traces) for the prover to be used within larger verification tools. We target applications where validation of formulas is crucial, such as validation of TLA$^+$ and B specifications.

6. New Results

6.1. Incremental development of distributed algorithms

Keywords: *distributed algorithms*, *event B*, *refinement*.

Participants: Nazim Benaïssa, Dominique Cansell, Dominique Méry, Joris Rehm.

The development of distributed algorithms and, more generally, of distributed systems, is a complex, delicate, and challenging process. The approach based on refinement helps to gain formality by using a proof assistant, and proposes to apply a design methodology that starts from the most abstract model and leads, in an incremental way, to the most concrete model, for producing a distributed solution. Our works help to formalize pre-existing algorithms as well as to develop new algorithms, as well as developing models for distributed systems.

Our research is supported by an ANR grant namely the RIMEL project and we have got many interesting and challenging results:

- Nazim Benaïssa and Dominique Méry consider the proof-based development of cryptographic protocols satisfying security properties. The main motivation is that cryptographic protocols are very complex systems to prove and to design, since they are based on specific assumptions. For instance, the model of Dolev-Yao provides a way to integrate a description of possible attacks when designing a protocol. They use existing protocols and want to provide a systematic way to prove but also to design cryptographic protocols; moreover, they would like to provide patterns for integrating cryptographic elements in an existing protocol. Communication channels are supposed to be unsafe. Analysing cryptographic protocols requires precise modelling of the attacker's knowledge. They present a technique for analysing key establishment protocols. The attacker's behaviour conforms to the Dolev-Yao model. In the Dolev-Yao model, the attacker has full control of the communication channel, and the cryptographic primitives are supposed to be perfect. We illustrate the technique on the well known Needham-Schroeder public key protocol and Blake-Wilson-Menezes key transport protocol. The underlying modelling language is Event B and is supported by the RODIN platform, which is used to validate models. Publications on this work include deliverables [22], [24], [10].
- Dominique Cansell has developed several distributed algorithms that were supposed to be complex by memebers of the distributed algorithms community. He has especially developed Dijkstra's distributed algorithm for stabilizing the state of distributed systems defined on a ring. Publications on this work include deliverable [22].

6.2. Proof-based patterns for developing system models

Keywords: *event B*, *modelling*, *proof*, *proof-based pattern*, *real time*, *refinement*, *theorem prover*.

Participants: Nazim Benaïssa, Dominique Cansell, Dominique Méry, Joris Rehm.

The development of programs or systems is carried out either using bottom-up techniques, or top-down techniques; we show how refinement and proof can be used to help in the proof-based development of systems or programs. If we consider a classical program development, When a problem is stated, the incremental proof-based methodology of event B starts by stating a very abstract model and further refinements lead to finer-grained event-based models which are used to build an algorithm. We have considered several case studies to understand what are the useful patterns when developing Event B models and distributed system. We summarize the different works:

- Dominique Cansell and Joris Rehm[29], [23] have proposed patterns to integrate the develoment of time-sensitive systems. The application was the ongoing development of the IEEE 1394 lmeader election protocol by integrating the notion of time to solve questions of contention.
- Nazim Benaissa and Dominique Méry have studied the way to integrate a model of attacks on the development of cryptographic protocols by minimizing the number of proof obligations to redo. Main results are found in the delivrable 3 of the RIMEL project[24], [10].
- Dominique Méry has addressed questions on teaching the development of classical algorithms and has supervised the development of a tool to produce automatically an algorithm from an Event B model. It appears that the method can be generalized to handle many programming paradigms. Two papers are summarizing the experiments [15], [14].

We describe more carefully the work on time constraint patterns for event B development. Real-time constraints are frequent requirements for distributed applications. In order to express such constraints in (mathematical) models, we intend to integrate time constraints in the modeling process based on event B models and refinement. The starting point of our work is the development in event B of the IEEE 1394 leader election protocol. From the documents of the standard, we derive temporal requirements to solve the contention problem and we propose a method for introducing time constraints using a pattern. The pattern captures time constraints in a generic event B development and it is applied to the IEEE 1394 case study. Joris Rehm proposed to introduce time constraints (propagation, sleeping time) in the B development of the IEEE 1394 protocol to solve the contention problem. To attain this goal he has defined a pattern in event B to introduce time constraint. This pattern is useful to carry out verification by theorem proving. In order to carry out verification by model-checking, another pattern for real-time modelling is defined. This new pattern uses only relative values about time and timing. The equivalence with the previous version is studied, and interesting properties for model-checking are reviewed. This work is explained in [28]. Finally a third pattern is proposed to handle duration over a predicate. This last pattern was used to provide the last model of a formal construction of Simpson's "2-slot algorithm". This algorithm is a reader-writer algorithm which uses only two slots and there are some time constraints on the reader and the writer to allow the asynchronous communication mechanism. This work was presented in [16].

6.3. Combination of theories through the exchange of (model-)equalities

Keywords: *SMT solvers*, *combination of decision procedures*, *decision procedures*, *theorem proving*.

Participants: Diego Caminha, Pascal Fontaine [correspondant], Stephan Merz.

The sound and complete combination of decision procedures for disjoint quantifier-free first-order languages requires the decision procedures to be able to produce entailed disjunctions of equalities: given a set of literals, the decision procedures should not only state whether the set is satisfiable or not, but also – in case the set is satisfiable – if there are (disjunctions of) equalities between terms that are deducible from the set. Finding such entailed (disjunctions of) equalities may be expensive, their number may be huge and handling all of them may becomes impractical. Furthermore, inspecting disjunctions of several equalities requires some case splitting on equalities in the disjunction; this may also be a factor of inefficiency.

We found out that the cooperation of decision procedure could not only be done through the exchange of deduced (disjunctions of) equalities, but also with the exchange of model-equalities that are guessed (with possible backtracking) by inspecting tentative models. Using these guessed equalities the expensive deduction and treatment of disjunctions of equalities is not required anymore. The case splitting is done at the SAT solver level, and only equalities are exchanged between decision procedures. These results were presented in [11].

6.4. Verification of Consensus algorithms

Keywords: *Heard-Of model*, *consensus algorithms*, *distributed algorithms*, *model checking*, *refinement*, *theorem proving*, *verification*.

Participants: Jacques Jaray, Stephan Merz, Mouna Saad.

Distributed algorithms are notoriously difficult to verify formally, be it through model checking or theorem proving, due to the absence of a global control state, asynchronous communication, and the high number of possible interleavings. Few interesting algorithms have been completely verified, and the hand proof of a moderately difficult algorithm such as Disk Paxos [35] in a formalism such as TLA$^+$ takes about 30 pages, which translate into more than 100 pages [36] when the proof is formalized in an interactive proof assistant such as Isabelle/HOL. The remedy proposed by the Mosel team is to describe and verify such algorithms at a higher level of abstraction, and then to prove a relation of (correctness-preserving) refinement.

Based on the observation that many distributed algorithms proceed in communication-closed rounds, Bernadette Charron-Bost and André Schiper proposed the Heard-Of (HO) model for describing such algorithms. In this model, executions of asynchronous distributed algorithms can be represented as sequences of global rounds, thus eliminating the need to consider fine-grained interleavings of actions of individual processes.

Consensus in the presence of failures of processes or communication is one of the fundamental problems of distributed computing. In joint work with Bernadette Charron-Bost from the LIX laboratory, Stephan Merz has written TLA$^+$ models of simple consensus algorithms, finite instances of which be verified by model checking. He has also proved one such algorithm in Isabelle/HOL; the proof of the HO variant of Paxos is ongoing work.

Mouna Saad's PhD thesis, co-supervised by Jacques Jaray and Samir Ben Ahmed from the Institut des Sciences Appliquées de Tunisie, also focuses on the HO model. In particular, she addresses the formal relationship between the HO representation of algorithms and their fine-grained model by state machines that communicate asynchronously. The objective is to prove that the conventional (fine-grained) model preserves the properties of interest (validity, agreement, and termination in the case of Consensus algorithms), if they have been established for the HO model.

6.5. A proof environment for TLA+

Keywords: *Isabelle*, *TLA*, *Zenon*, *proof assistant*, *proof language*, *theorem proving*.

Participant: Stephan Merz.

Within a joint project between INRIA and Microsoft Research (see section 8.4) that aims at developing a verification environment for TLA$^+$, Stephan Merz has further contributed to the design of the proof language for TLA$^+$, tentatively called TLA^{+2}. He has also encoded a core fragment of TLA$^+$ as a new object logic in the interactive proof assistant Isabelle. Together with Kaustuv Chaudhuri, Damien Doligez, and Leslie Lamport, he has worked on the TLA^{+2} Proof Manager and its integration with Isabelle and the Zenon tableau prover. A first prototype is now operational and allows us to validate the language design and perform machine-checked TLA^{+2} proofs. A presentation of the overall design and the current state of the project has been published at the Workshop on Knowledge Exchange between Automated Provers and Proof Assistants [13].

6.6. Work on +CAL and model checking Grid algorithms

Keywords: *TLA*, *distributed algorithms*, *model checking*, *proof.*

Participants: Sabina Akhtar, Cristian Rosa, Stephan Merz, Martin Quinson [of project team AlGorille].

In joint research with Martin Quinson of the AlGorille team of INRIA Nancy and LORIA, we are interested in the verification (essentially via model checking) of distributed and peer-to-peer algorithms. Whereas model checking is now routinely used for concurrent and embedded systems, existing algorithms and tools can rarely be effectively applied for the verification of asynchronous distributed algorithms and systems.

We have started to explore two approaches to this problem. In the first approach, Sabina Akhtar has studied in her Master thesis [27] an extension of the $^+$CAL language [38] defined by Lamport. The extension is intended for describing and verifying models of distributed algorithms, whereas the original language is geared towards shared-memory concurrent programming. Most importantly, the extensions include nested processes and scoped variables; this is useful for modeling threads that communicate via shared memory versus processes that communicate by message passing. Other extensions lift ad-hoc restrictions on the use of labels and the atomicity of instruction blocks imposed by the original language. Sabina has implemented a new compiler for the language and validated her approach on several examples. In her thesis, she plans to continue to work on this language and in particular to optimize the existing model checking algorithm for $^+$CAL so that it takes advantage of the locality information in order to reduce the number of interleavings that must be explored.

On the other hand, we are interested in adding model checking support to the SimGrid platform and specifically the GRAS API for simulating and implementing Grid algorithms. During his stay as an INRIA Internship student in the summer, Cristian Rosa explored this possibility and implemented a first prototype. During his PhD thesis supervised by Martin Quinson and Stephan Merz, Cristian Rosa will study dedicated model checking techniques and algorithms for this platform. In contrast to similar projects that aim at verification of essentially unrestricted C programs, we believe that significant optimizations are possible due to the clearly specified programming model of SimGrid and GRAS.

6.7. Provably correct lock-free data structures

Keywords: *automatic proof*, *linearizability*, *lock-free algorithm*, *verification*.

Participants: Loïc Fejoz, Pascal Fontaine, Stephan Merz.

The development of multi-threaded programs is no longer restricted to operating system experts and specialised application areas, but is entering mainstream programming. A central problem is to ensure that different threads can access shared data structures without interference. The traditional solutions are based on locks, which can be either coarse-grained (applying to the entire data structure) or fine-grained (applying to individual elements of the data structure). Both cases have severe drawbacks: coarse-grained locks limit the possible degree of parallelism, while fine-grained locks are hard to control and prone to deadlocks. Several mechanisms of concurrent programming without locks have been proposed in the literature. In this project, funded by a Microsoft ERO PhD grant, we investigate verification techniques for lock-free data structures. A dedicated refinement framework has been elaborated and formalized in Isabelle/HOL. It results in proof obligations that are tailored to this class of algorithms and ensures that concurrent accesses are linearizable. In particular, a user of a lock-free data structure can reason as if all accesses by the different processes occurred atomically, in some unpredictable order.

A generator of proof obligations has been implemented that can output the verification conditions for a given set of algorithm specifications in the format of automatic theorem provers and SMT solvers. The method has been validated for several lock-free algorithms, including an implementation of the RDCSS algorithm based on a CAS_1 compare-and-swap instruction. In cooperation with Pascal Fontaine and David Déharbe, who have implemented a tailored instantiation heuristic for haRVey, all proof obligations have been discharged successfully for this and other examples, whereas previous verification attempts were restricted to model checking hand-tailored abstractions. The method has been presented at a workshop co-located with CAV [12]; a more complete presentation has been submitted to a conference. The PhD thesis of Loïc Fejoz has been submitted for defense in early 2009.

6.8. Computer science history

Keywords: *history of computer science*.

Participant: Denis Roegel.

Denis Roegel has ongoing interest in the history of computer science. He published a description of what is now believed the oldest existing key-driven calculating machine in the *IEEE Annals of the History of Computing* [8]. This machine was patented by Schwilgué in 1844 and the patent was disovered by chance in 2003.

Another old calculating instrument is the abacus, which is still in use now in Asia. Denis Roegel worked on the Chinese and Japanese versions of this instrument, analyzed a particular addition algorithm, and wrote macros to draw flexible abaci configurations.

Denis Roegel is also interested in the automatic generation of historical and complex mathematical tables. One such example consisted in a *qibla* table made by Al-Khalīlī in the 14th century. This table gave the direction of Mecca for a number of latitudes and longitudes, but it did so by using Abjad numerals (numerals used in the Arab world before the Hindu-Arabic numerals). Denis Roegel's work has extended Al-Khalīlī's table for the entire world, but it strived to preserve the original spirit of the table [26]. Finally, Denis Roegel is also interested in the graphical representation of equations and the historical means to solve them. Classical representations of equations are those provided by "nomograms", a field developped by Maurice d'Ocagne at the end of the 19th century, and related to slide rules. Denis Roegel worked on one such nomogram, aimed at computing the date of Easter, and he showed how this nomogram can be analyzed and reconstructed using the MetaPost graphical language.

6.9. Computer graphics and typesetting

Keywords: *Metapost*, *graphics*.

Participant: Denis Roegel.

Denis Roegel has continued to develop extensions to MetaPost addressing various abstract representations of objects. Some of these extensions are related to historical instruments or methods and are described in another section. In addition to the work on Chinese and Japanese abaci and on nomography, Denis Roegel has also worked on drawing Chinese versions of Sudokus, so-called Kanji-sudokus [9].

7. Contracts and Grants with Industry

7.1. Microsoft ERO PhD Grant

Participants: Loïc Fejoz, Stephan Merz.

The PhD thesis of Loïc Fejoz (see section 6.7) is supported by a Microsoft ERO PhD grant from January 2006 to December 2008. Our main contact at Microsoft Research is Tim Harris who develops algorithms for lock-free data structures that we verify.

8. Other Grants and Activities

8.1. ANR SETIN RIMEL

Keywords: *B patterns*, *distributed algorithms*, *event B*, *refinement*.

Participants: Nazim Benaïssa, Dominique Cansell, Dominique Méry, Joris Rehm.

The project RIMEL, carried out in cooperation with the teams led by Mohamed Mosbah and by Yves Métivier at LABRI Bordeaux and with ClearSy Systems Engineering, focuses on the *refinement* of event-based models and aims to develop new features related to refinement, such as the reusability of refinement-based development, the composition and decomposition of models with respect to the refinement, the definition of proof-based design patterns, the integration of time constraints and probabilistic aspects, and the development of case studies, especially related to distributed systems. Probabilistic and/or timing aspects are of central importance for many distributed algorithms (such as the IEEE 1394 Tree Identification algorithm), and should therefore be integrated into the framework based on refinement. In this project, we are focusing on distributed algorithms and applications able to recover from a bad state, so-called self-healing systems. We also plan to apply the techniques to system engineering. Our proposed work is decomposed into several research directions:

1. Theory of refinement: integration of fairness constraints and liveness properties, probabilistic refinement and extensions of refinement scope.
2. Proof-based design patterns: a system engineering approach to justifying claims for security and trustworthiness.
3. Self-Healing Systems and Distributed Algorithms.
4. Tools and Dissemination.

The RIMEL project coordinates its research activities through case studies and applications, which give us concrete points of departure for our conceptual research. Results are gathered in three delivrables [22], [23], [24].

8.2. Formal system engineering

Keywords: *information*, *modelling*, *refinement*, *system*.

Participants: Dominique Méry, Gérard Morel, Jean-François Pétin.

The project LABIME has started in the GIS 3SGS and gathers three groups of research from CRAN, LORIA and EDF. The aim of this project is to contribute to the definition of a formal language for identifying and modelling operator requirements about plant operations and information. To obtain the required genericity, the language must be clearly defined without taking into account human factors in the definition of the plant control imagery, the human-system distribution as well as the computerized technologies to be used for plant operations. The project is ongoing.

8.3. Formal analysis of programmable logic for reverse engineering and verification of safety critical behaviour in control systems

Keywords: *PLC*, *control*, *event B*, *safety-critical system*, *transport*.

Participants: Dominique Cansell, Dominique Méry, Rosemary Monahan, Adam Winstanley.

The project is supported by the ULYSSES initiative between France and Ireland; it invoves the department of computer science of the University NUI Maynooth and the team MOSEL. It addresses the question of formally verifying safety critical properties of already implemented software control systems, guaranteeing their reliability and safety. In particular, we address the following questions: What is the best methodology for generating a formal system requirements document (written in Event B) for an already existing tram control system? What is the relationship between the Event B and Programmable Logic? How effectively can we support the formal translation of a system specification written in Event B to its implementation written in programmable logic? Can we demonstrate that this formal transformation preserves the safety critical properties as specified for an existing tram control system? A combination of reverse engineering and refinement techniques will be used to prove the safety critical properties of a tram control system generating a suite of proof based patterns that may be used in the verification of safety critical properties of similar systems. Case studies involving subsystems of the tram control system will be used to develop Masters level courses, ensuring technology transfer between industry and the classroom, and vice versa. The visit of Dominique Méry in February led to a series of lectures in the master program; Dominique Cansell and Dominique Méry are completing models for ensuring the quality of produced codes.

8.4. Tools and Methodologies for Formal Specifications and for Proofs

Participant: Stephan Merz.

As part of the Joint Laboratory between INRIA and Microsoft Research, Stephan Merz participates in the project on Tools and Methodologies for Formal Specifications and for Proofs. The objective of the project is to develop an environment for modeling and verifying distributed algorithms in TLA$^+$ (see also section 6.5).

8.5. Exchanges with Tunisia

Participants: Jacques Jaray, Stephan Merz, Mouna Saad.

We have had a very fruitful cooperation with the team led by Professor Samir Ben Ahmed at the Institut Supérieur d'Informatique (ISI) in Tunis for several years. Jacques Jaray was invited several times to teach courses at the Master's level. The cooperation is currently supported by CMCU (*Comité Mixte de Coopération Universitaire*) through the project DEFI coordinated by Jeanine Souquières and Samir Ben Ahmed. Mouna Saad is preparing her Ph.D. thesis in joint supervision between the University of Tunis and the INPL at Nancy. Mouna Saad visited LORIA for two months in 2008. Reciprocally, Jacques Jaray and Stephan Merz visited Tunis for one week each.

9. Dissemination

9.1. Program committees and conference organisation

- Dominique Cansell was a member of the program committee of ABZ 2008.
- Dominique Méry serves on the program committees of FASE 2008, FME 2008, CAL 2008, ABZ 2008,PSI 2009, CAL 2009, IFM 2009.
- Stephan Merz served on the program committees of the conferences FASE 2009, IFM 2009, SofSem 2009, and TCS 2008, and of the workshops AVoCS 2008, SafeCert 2008 (co-located with ETAPS), Verify 2008 (co-located with IJCAR), and WISG 2009. Together with Serge Autexier, Heiko Mantel, and Tobias Nipkow, he is co-editing a special issue of the Journal of Automated Reasoning devoted to advances in system verification, to appear in early 2009. Together with Christoph Weidenbach and Manuel Lamotte of the Max-Planck-Institute for Informatics, he co-organized the Summer School on Verification Technology, Systems and Applications, which took place on September 15–19, 2008, in Saarbrücken.

9.2. Tutorials, invited talks, panels

- Dominique Méry gave lectures on Event B modelling in the master programme of the NUI Maynooth in Ireland in February 2008.
- Together with Nicolas Navet, Stephan Merz edited a book on the modeling and verification of real-time systems [21].

  Together with Christoph Weidenbach, he gave a lecture series at Tsinghua University, Beijing, on the TLA$^+$ specification language and automated reasoning.

9.3. Theses, habilitations, academic duties

- Pascal Fontaine is a member of an international working group designing the proof format for SMT solvers.
- Jacques Jaray is Vice President of Institut National Polytechnique de Lorraine and Director of the INPL Computing Resources Department.
- Dominique Méry
  - is a member of the IFIP Working Group 1.3 on *Foundations of System Specification*.
  - is the Head of the master programme in computer science for the three universities of Nancy.
  - is a member of the scientific council of the LORIA laboratory.
  - is an expert for the French Ministry of Education (DS9).
  - is an expert for the French Agence Nationale de la Recherche (ANR).
  - is director of international affairs at ESIAL Nancy.
  - is the president of the APCB association.
  - is the Head of the Doctoral School IAEM Lorraine.
- Stephan Merz coordinates, together with Dominique Sauter of the CRAN laboratory, the research theme on Systems Safety and Security within the Research Cluster on Digital Modeling in Lorraine. He is the delegate for international relations at LORIA and INRIA Nancy. He is a member of the managing board of INRIA Nancy. He represents INRIA in the supervisory board for the partnership with the Saarbrücken-Kaiserslautern area in Germany and in the Scientific Directorate of the Dagstuhl meeting center. He is an elected member of the evaluation committee of INRIA and a nominated member of the Section 7 of the Comité National de la Recherche Scientifique. He is a member of the IFIP Working Group 2.2 *Formal Description of Programming Concepts*. He is an expert for the French Agence Nationale de la Recherche (ANR) and evaluated grant applications for national research councils in Austria and the Netherlands.

9.4. Teaching

The majority of the members of the MOSEL team are university employees and have significant teaching obligations. We only indicate the graduate courses they have been teaching in 2008.

- Pascal Fontaine gave introductory courses on specification and verification in the Master's program of the Miage section at Nancy.
- Dominique Méry gave courses in the Master's program at Nancy on: formal system engineering, modelling and verification of systems, theoretical computer science, development of software systems, distributed algorithms.
- Stephan Merz, together with Laurent Vigneron, gave a course on algorithmic verification in the Master's program in Nancy.

10. Bibliography

Major publications by the team in recent years

[1] J.-R. ABRIAL, D. CANSELL, D. MÉRY. *A Mechanically Proved and Incremental Development of IEEE 1394 Tree Identify Protocol*, in "Formal Aspects of Computing", vol. 14, n^{o} 3, 2003, p. 215-227.

[2] J.-R. ABRIAL, D. CANSELL, D. MÉRY. *Refinement and Reachability in Event$_B$*, in "ZB", 2005, p. 222-241.

[3] D. CANSELL, D. MÉRY. *Formal and Incremental Construction of Distributed Algorithms: On the Distributed Reference Counting Algorithm*, in "Theoretical Computer Science", vol. 364, n^{o} 3, 2006, p. 318-337, http://hal.inria.fr/inria-00093164/en/.

[4] D. CANSELL, D. MÉRY. *The Event-B Modelling Method: Concepts and Case Studies*, in "Logics of specification languages Monographs in Theoretical Computer Science", D. BJØRNER, M. HENSON (editors), Springer, 2008, p. 47-152.

[5] F. KRÖGER, S. MERZ. *Temporal Logic and State Systems*, Springer, 2008, http://hal.inria.fr/inria-00274806/en/.

[6] S. MERZ. *The Specification Language TLA$^+$*, in "Logics of Specification Languages, Berlin-Heidelberg", D. BJØRNER, M. C. HENSON (editors), Monographs in Theoretical Computer Science, Springer, 2008, p. 401–452.

[7] D. MÉRY, D. CANSELL, C. PROCH, D. ABRAHAM, P. DITSCH. *The challenge of QoS for digital television services*, in "EBU Technical Review", April 2005.

Year Publications

Articles in International Peer-Reviewed Journal

[8] D. ROEGEL. *An Early (1844) Key-Driven Adding Machine*, in "IEEE Annals of the History of Computing", vol. 30, 2008, p. 59-65, http://hal.inria.fr/inria-00336086/en/.

[9] D. ROEGEL. *Kanji-Sudokus: Integrating Chinese and Graphics*, in "Tugboat", vol. 29, 2008, p. 317-319, http://hal.inria.fr/inria-00336088/en/.

International Peer-Reviewed Conference/Proceedings

[10] N. BENAISSA. *Modelling Attacker's Knowledge for Cascade Cryptographic Protocols*, in "First International Conference on Abstract State Machines, B and Z - ABZ 2008 Abstract State Machines, B and Z Lecture Notes in Computer Science, Royaume-Uni London", E. BÖRGER, M. BUTLER, J. P. BOWEN, P. BOCA (editors), vol. 5238, Springer, 2008, p. 251-264, http://hal.inria.fr/inria-00336641/en/.

[11] D. CAMINHA B. DE OLIVEIRA, D. DÉHARBE, P. FONTAINE. *Combining decision procedures by (model-)equality propagation*, in "Brazilian Symposium on Formal Methods (SBMF), Brésil Salvador, Bahia", P. MACHADO, A. ANDRADE, A. DURAN (editors), 2008, http://hal.inria.fr/inria-00337979/en/.

[12] L. FEJOZ, S. MERZ. *Towards automatic proofs of lock-free algorithms*, in "Exploiting Concurrency Efficiently and Correctly, États-Unis d'Amérique Princeton", 2008, http://hal.inria.fr/inria-00285752/en/.

Workshops without Proceedings

[13] K. CHAUDHURI, D. DOLIGEZ, L. LAMPORT, S. MERZ. *A TLA+ Proof System*, in "Knowledge Exchange: Automated Provers and Proof Assistants (KEAPPA), Qatar Doha", 2008, http://hal.inria.fr/inria-00338299/en/.

[14] D. MÉRY. *A simple refinement-based method for constructing algorithms*, in "First International Workshop on Formal Methods Education and Training", J. DAVIES, J. GIBBONS, M. HINCHEY, K. TAGUCHI (editors), Report GRACE-TR-2008-03, GRACE Center, National Institute of Informatics,, 2008.

[15] D. MÉRY. *Teaching programming methodology using Event B*, in "The B method : from Research to Teaching The B method : from Research to Teaching, France Nantes", C. ATTIOGBÉ, H. HABRIAS (editors), H. Habrias, 2008, http://hal.inria.fr/inria-00287231/en/.

[16] J. REHM. *A Duration Pattern for Event-B Method*, in "2nd Junior Researcher Workshop on Real-Time Computing - JRWRTC 2008, France Rennes", 2008, http://hal.archives-ouvertes.fr/hal-00336320/en/.

Scientific Books (or Scientific Book chapters)

[17] D. CANSELL, D. MÉRY. *The Event-B Modelling Method: Concepts and Case Studies*, in "Logics of specification languages Monographs in Theoretical Computer Science", D. BJØRNER, M. HENSON (editors), Springer, 2008, p. 47-152.

[18] F. KRÖGER, S. MERZ. *Temporal Logic and State Systems*, Springer, 2008, http://hal.inria.fr/inria-00274806/en/.

[19] S. MERZ. *An introduction to model checking*, in "Modeling and Verification of Real-Time Systems - Formalisms and Software Tools", S. MERZ, N. NAVET (editors), ISTE Publishing, 2008, p. 81-116, http://hal.inria.fr/inria-00187577/en/.

[20] S. MERZ. *The Specification Language TLA+*, in "Logics of specification languages Monographs in Theoretical Computer Science", D. BJØRNER, M. HENSON (editors), Springer, 2008, p. 401-452, http://hal.inria.fr/inria-00338330/en/.

Books or Proceedings Editing

[21] S. MERZ, N. NAVET (editors). *Modeling and Verification of Real-Time Systems - Formalisms and Software Tools*, ISTE Publishing, 2008, http://hal.inria.fr/inria-00187581/en/.

Research Reports

[22] P. ANR-RIMEL. *Développement d'algorithmes répartis*, RIMEL Deliverable, LORIA, February 2008.

[23] P. ANR-RIMEL. *Intégration de contraintes temps-réel au sein d'un processus de développement incrémental basé sur la preuve.*, RIMEL Deliverable, LORIA, July 2008.

[24] P. ANR-RIMEL. *Proof-based design patterns*, RIMEL Deliverable, LORIA, July 2008.

[25] N. BENAISSA, D. MÉRY. *Développement incrémental prouvé de systèmes répartis : le cas Mondex*, Rapport de recherche, 2008, http://hal.inria.fr/inria-00336655/en/.

[26] D. ROEGEL. *An Extension of Al-Khalīlī's Qibla Table to the Entire World*, Rapport de recherche, 2008, http://hal.inria.fr/inria-00336090/en/.

Other Publications

[27] S. AKHTAR. *Formal Verification of Distributed Algorithms in +CAL 2.0*, Masters thesis, Nancy Université (Université Henri Poincaré), Nancy, France, June 2008.

[28] J. REHM. *From Absolute-Timer to Relative-Countdown: Patterns for Model-Checking*, 2008, http://hal.archives-ouvertes.fr/hal-00319104/en/.

[29] J. REHM. *Proved Development of the Real-Time Properties of the IEEE 1394 Root Contention Protocol with the Event B Method*, 2008, http://hal.inria.fr/inria-00336624/en/.

References in notes

[30] W.-P. DE ROEVER, H. LANGMAACK, A. PNUELI (editors). *Compositionality: The Significant Difference*, Lecture Notes in Computer Science, vol. 1536, Springer-Verlag, 1998.

[31] J.-R. ABRIAL. *Extending B without changing it (for developing distributed systems)*, in "1st Conference on the B method", H. HABRIAS (editor), IRIN Institut de recherche en informatique de Nantes, 1996, p. 169–190.

[32] J.-R. ABRIAL. *The B-Book: Assigning Programs to Meanings*, Cambridge University Press, 1996.

[33] J.-R. ABRIAL, D. CANSELL, D. MÉRY. *A Mechanically Proved and Incremental Development of IEEE 1394 Tree Identify Protocol*, in "Formal Aspects of Computing", vol. 14, n^{o} 3, 2003, p. 215-227.

[34] D. CANSELL, D. MÉRY. *Formal and Incremental Construction of Distributed Algorithms: On the Distributed Reference Counting Algorithm*, in "Theoretical Computer Science", vol. 364, n^{o} 3, 2006, p. 318-337, http://hal.inria.fr/inria-00093164/en/.

[35] E. GAFNI, L. LAMPORT. *Disk Paxos*, in "Distributed Computing", vol. 16, n^{o} 1, 2003, p. 1–20.

[36] M. JASKELIOFF, S. MERZ. *Proving the Correctness of Disk Paxos in Isabelle/HOL*, June 2005, http://afp.sourceforge.net/entries/DiskPaxos.shtml, The Archive of Formal Proofs.

[37] L. LAMPORT. *Specifying Systems*, Addison-Wesley, Boston, Mass., 2002.

[38] L. LAMPORT. *Checking a Multithreaded Algorithm with* +CAL, in "20th Intl. Symp. Distributed Computing (DISC 2006), Stockholm, Sweden", S. DOLEV (editor), Lecture Notes in Computer Science, vol. 4167, 2006, p. 151–163.

[39] D. MÉRY, D. CANSELL, C. PROCH, D. ABRAHAM, P. DITSCH. *The challenge of QoS for digital television services*, in "EBU Technical Review", April 2005.