INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team triskell

Model-Driven Engineering for Component-Based Software

Rennes - Bretagne-Atlantique

Theme : Distributed Systems and Services

Activity Report

2010

Table of contents

1. Team

Research Scientist
Benoit Baudry [Junior Researcher Inria]

Faculty Members
Jean-Marc Jézéquel [Team Leader, Professor Université de Rennes 1, HdR]
Olivier Barais [Associate Professor Université de Rennes 1]
Noël Plouzeau [Associate Professor Université de Rennes 1]
Benoit Combemale [Associate Professor Université de Rennes 1]
Naouel Moha [Associate Professor Université de Rennes 1]
Johann Bourcier [Associate Professor Université de Rennes 1, from September 2010]

Technical Staff
Didier Vojtisek [Research engineer Inria]
Jacques Falcou [Inria (project CHESS)]
Mahmoud Ben Hassine [Inria (project Galaxy), until October 2010]
Hajanirina Johary Rambelontsalama [Inria IJD]
Claire Diehl-Watrin [Inria (project ENTIMID), from July 2010]
Marie Gouyette [Inria (project MOVIDA)]

PhD Students
Romain Delamare [Brittany Council grant, until January 2010]
Brice Morin [INRIA grant, until September 2010]
François Fouquet [MESR grant]
Freddy Muñoz [INRIA grant, until September 2010]
Sagar Sen [INRIA grant, until April 2010]
Tejeddine Mouelhi [ENST grant, until September 2010]
Maha Driss [INRIA grant, Co-tutelle with U. Tunis]
Paul Istoan [Co-tutelle with U. Luxembourg]
Grégory Nain [INRIA grant]
Muhammad-Ali Memon [SFERE grant, until October 2010]
Mickael Clavreul [INRIA grant]
Juan Cadavid [INRIA grant]
Erwan Daubert [INRIA grant]
Olivier-Nathanael Ben David [INRIA grant, from October 2010]
Stephen Creff [CIFRE grant (with THALES)]
Clément Guy [MESR grant, from October 2010]
Aymerick Hervieu [CIFRE grant (with KEREVAL), from October 2010]
Tam Le Nhan [Vietnam Grant, from October 2010]
Viet-Hoa Nguyen [INRIA Grant]
Emmanuelle Rouillé [CIFRE Grant (with SODIFRANCE), from October 2010]
Nicolas Sannier [CIFRE Grant (with EDF), from October 2010]

Post-Doctoral Fellows
Arnaud Blouin [INRIA grant]
Cédric Bouhours [INRIA grant, from September 2010]
Xavier Dolques [INRIA grant, from December 2010]

Visiting Scientists
Régis Fleurquin [Délégation INRIA, on leave from Université de Bretagne Sud, until August 2010]
Gerson Sunyé [Délégation INRIA, on leave from Université de Nantes, from September 2010]

Administrative Assistant
Loic Lesage [TR Inria]

2. Overall Objectives

2.1. Introduction

2.1.1. Research fields

In its broad acceptation, Software Engineering consists in proposing practical solutions, founded on scientific knowledge, in order to produce and maintain software with constraints on costs, quality and deadlines. In this field, it is admitted that the complexity of a software increases exponentially with its size. However on the one hand, the size itself of the software is on average multiplied by ten every ten years, and on the other hand, economic pressures pushed towards reducing the duration of developments, and in increasing the rates of modifications made to the software.

To face these problems, today's mainstream approaches build on the concept of component based software. The assembly of these components makes it possible to build families of products (a.k.a. *product lines*) made of many common parts, while remaining opened to new evolutions. As component based systems grow more complex and mission-critical, there is an increased need to be able to represent and reason on such assemblies of components. This is usually done by building models representing various aspects of a product line, such as functional variations, structural aspects (object paradigm), or dynamic aspects (languages of scenarios), without neglecting of course non-functional aspects like quality of service (performance, reliability, etc.) described in the form of contracts. Model Driven Engineering (MDE) is then a sub-domain of software engineering focusing on reinforcing design, validation and test methodologies based on the automatic processing of multi-dimensional models.

2.1.2. Project-team Presentation Overview

The research domain of the Triskell project is the model driven development of software product lines. Triskell is particularly interested in component based reactive and large scale distributed systems with quality of service constraints, including reliability, performance, timeliness etc.

Triskell's main objective is to develop model-based methods and tools to help the software designer to efficiently obtain a certain degree of confidence in the reliability of component assemblies that may include third-party components. This involves, in particular, investigating modeling languages allowing specification of both functional and non-functional aspects for software engineering activities ranging from requirements to detailed design. It also involves building a continuum of tools which make use of these models, from model validation and verification, automatic application of design patterns, to test environments and on-line monitors supervising the behavior of the components in Dynamically Adaptable Systems. Since these modeling languages and associated tools appear quite open-ended and very domain specific, there is a growing need for *"tools for building tools for building software"*. Triskell is hence developing KerMeta as an original meta modeling approach allowing the user to fully define his modeling languages (including dynamic semantics) and associated environments (including interpreters, compilers, importers/exporters, etc.) within Eclipse.

To avoid the pitfall of developing *"tools for building tools for the sake of it"*, the Triskell project also has the goal of explicitly connecting its research results to industrial problems through collaborations with industry and technology transfer actions. This implies, in particular, taking into account the industrial standards of the field, namely the Eclipse Modeling Framework (EMF), the OMG's Meta-Object Facility (MOF) and Unified Modeling Language (UML), as well as domain specific component models such as OSGi.

Triskell is at the frontier of two fields of software: the field of specification and formal proof, and that of design which, though informal, is organized around best practices (e.g.; separation of concerns with aspects, models, design patterns, or the use of off-the-shelf components). We believe that the use of our techniques will make it possible to improve the transition between these two worlds, and will contribute to the fluidity of the processes of design, implementation and testing of software.

2.2. Highlights of the Year

- Triskell has organized the 9th International Conference on Aspect-Oriented Software Development AOSD 2010, that was held in Rennes and Saint-Malo on March 15-19, 2010: Jean-Marc Jézéquel was General Chair and Benoit Baudry was Conference Chair.
- We received the best paper award for the paper *Business and Information System Alignment: A Formal Solution for Telecom Services* presented at ICSEA 2010.

3. Scientific Foundations

3.1. Overview

The Triskell project studies Model Driven Engineering techniques for the reliable construction of software product lines, especially for distributed and reactive software. The key problems are components modeling and the development of formal manipulation tools to refine the design, code generation and test activities. The validation techniques used are based on complex simulations of models building on the standards in the considered domain.

3.2. Model Driven Engineering for Distributed Software

3.2.1. Software Product Lines

It is seldom the case nowadays that we can any longer deliver software systems with the assumption that one-size-fits-all. We have to handle many variants accounting not only for differences in product functionalities (range of products to be marketed at different prices), but also for differences in hardware (e.g.; graphic cards, display capacities, input devices), operating systems, localization, user preferences for GUI ("skins"). Obviously, we do not want to develop from scratch and independently all of the variants the marketing department wants. Furthermore, all of these variant may have many successive versions, leading to a two-dimensional vision of product-lines.

3.2.2. Object-Oriented Software Engineering

The object-oriented approach is now widespread for the analysis, the design, and the implementation of software systems. Rooted in the idea of modeling (through its origin in Simula), object-oriented analysis, design and implementation takes into account the incremental, iterative and evolutive nature of software development [75], [73]: large software system are seldom developed from scratch, and maintenance activities represent a large share of the overall development effort.

In the object-oriented standard approach, objects are instances of classes. A class encapsulates a single abstraction in a modular way. A class is both *closed*, in the sense that it can be readily instantiated and used by clients objects, and *open*, that is subject to extensions through inheritance [77].

3.2.3. Design Pattern

Since by definition objects are simple to design and understand, complexity in an object-oriented system is well known to be in the *collaboration* between objects, and large systems cannot be understood at the level of classes and objects. Still these complex collaborations are made of recurring patterns, called design patterns. The idea of systematically identifying and documenting design patterns as autonomous entities was born in the late 80's. It was brought into the mainstream by such people as Beck, Ward, Coplien, Booch, Kerth, Johnson, etc. (known as the Hillside Group). However the main event in this emerging field was the publication, in 1995, of the book *Design Patterns: Elements of Reusable Object Oriented Software* by the so-called Gang of Four (GoF), that is E. Gamma, R. Helm, R. Johnson and J. Vlissides [74]. Today, design patterns are widely accepted as useful tools for guiding and documenting the design of object-oriented software systems. Design patterns play many roles in the development process. They provide a common vocabulary for design, they reduce system complexity by naming and defining abstractions, they constitute a base of experience for building reusable software, and they act as building blocks from which more complex designs can be built. Design patterns can be considered reusable micro-architectures that contribute to an overall system architecture. Ideally, they capture the intent behind a design by identifying the component objects, their collaborations, and the distribution of responsibilities. One of the challenges addressed in the Triskell project is to develop concepts and tools to allow their formal description and their automatic application.

3.2.4. Component

The object concept also provides the bases needed to develop *software components*, for which Szyperski's definition [82] is now generally accepted, at least in the industry:

> *A software component is a unit of composition with contractually specified interfaces and explicit context dependencies only. A software component can be deployed independently and is subject to composition by third party.*

Component based software relies on assemblies of components. Such assemblies rely in turn on fundamental mechanisms such as precise definitions of the mutual responsibility of partner components, interaction means between components and their non-component environment and runtime support (e.g. .Net, EJB, Corba Component Model CCM, OSGI or Fractal).

Components help reducing costs by allowing reuse of application frameworks and components instead of redeveloping applications from scratch (product line approach). But more important, components offer the possibility to radically change the behaviors and services offered by an application by substitution or addition of new components, even a long time after deployment. This has a major impact of software lifecycle, which should now handle activities such as the design of component frameworks, the design of reusable components as deployment units, the validation of component compositions coming from various origins and the component life-cycle management.

Empirical methods without real component composition models have appeared during the emergence of a real component industry (at least in the Windows world). These methods are now clearly the cause of untractable validation and of integration problems that can not be transposed to more critical systems (see for example the accidental destruction of Ariane 501 [76]).

Providing solutions for formal component composition models and for verifiable quality (notion of *trusted components*) are especially relevant challenges. Also the methodological impact of component-based development (for example within the maturity model defined by the SEI) is also worth attention.

3.2.5. Contracts

Central to this trusted component notion is the idea of *contract*. A software contract captures mutual requirements and benefits among stakeholder components, for example between the client of a service and its suppliers (including subcomponents). Contracts strengthen and deepen interface specifications. Along the lines of abstract data type theory, a common way of specifying software contracts is to use boolean assertions called pre- and post-conditions for each service offered, as well as class invariants for defining general consistency properties. Then the contract reads as follows: The client should only ask a supplier for a service in a state where the class invariant and the precondition of the service are respected. In return, the supplier promises that the work specified in the post-condition will be done, and the class invariant is still respected. In this way rights and obligations of both client and supplier are clearly delineated, along with their responsibilities. This idea was first implemented in the Eiffel language [78] under the name *Design by Contract*, and is now available with a range of expressive power into several other programming languages (such as Java) and even in the Unified Modeling Language (UML) with the Object Constraint Language (OCL) [83]. However, the classical predicate based contracts are not enough to describe the requirements of modern applications. Those applications are distributed, interactive and they rely on resources with random quality of service. We have shown that classical contracts can be extended to take care of synchronization and extrafunctional properties of services (such as throughput, delays, etc) [72].

3.2.6. Models and Aspects

As in other sciences, we are increasingly resorting to modeling to master the complexity of modern software development. According to Jeff Rothenberg,

> *Modeling, in the broadest sense, is the cost-effective use of something in place of something else for some cognitive purpose. It allows us to use something that is simpler, safer or cheaper than reality instead of reality for some purpose. A model represents reality for the given purpose; the model is an abstraction of reality in the sense that it cannot represent all aspects of reality. This allows us to deal with the world in a simplified manner, avoiding the complexity, danger and irreversibility of reality.*

So modeling is not just about expressing a solution at a higher abstraction level than code. This has been useful in the past (assembly languages abstracting away from machine code, 3GL abstracting over assembly languages, etc.) and it is still useful today to get a holistic view on a large C++ program. But modeling goes well beyond that.

Modeling is indeed one of the touchstone of any scientific activity (along with validating models with respect to experiments carried out in the real world). Note by the way that the specificity of engineering is that engineers build models of artifacts that usually do not exist yet (with the ultimate goal of building them).

In engineering, one wants to break down a complex system into as many models as needed in order to address all the relevant concerns in such a way that they become understandable enough. These models may be expressed with a general purpose modeling language such as the Unified Modeling Language (UML), or with Domain Specific Languages when it is more appropriate.

Each of these models can be seen as the abstraction of an aspect of reality for handling a given concern. The provision of effective means for handling such concerns makes it possible to establish critical trade-offs early on in the software life cycle, and to effectively manage variation points in the case of product-lines.

Note that in the Aspect Oriented Programming community, the notion of aspect is defined in a slightly more restricted way as the modularization of a cross-cutting concern. If we indeed have an already existing “main” decomposition paradigm (such as object orientation), there are many classes of concerns for which clear allocation into modules is not possible (hence the name “cross-cutting”). Examples include both allocating responsibility for providing certain kinds of functionality (such as loggin) in a cohesive, loosely coupled fashion, as well as handling many non-functional requirements that are inherently cross-cutting e.g.; security, mobility, availability, distribution, resource management and real-time constraints.

However now that aspects become also popular outside of the mere programming world [81], there is a growing acceptance for a wider definition where an aspect is a concern that can be modularized. The motivation of these efforts is the systematic identification, modularization, representation, and composition of these concerns, with the ultimate goal of improving our ability to reason about the problem domain and the corresponding solution, reducing the size of software model and application code, development costs and maintenance time.

3.2.7. Design and Aspect Weaving

So really modeling is the activity of separating concerns in the problem domain, an activity also called *analysis*. If solutions to these concerns can be described as aspects, the design process can then be characterized as a weaving of these aspects into a detailed design model (also called the solution space). This is not new: this is actually what designers have been effectively doing forever. Most often however, the various aspects are not *explicit*, or when there are, it is in the form of informal descriptions. So the task of the designer is to do the weaving in her head more or less at once, and then produce the resulting detailed design as a big tangled program (even if one decomposition paradigm, such as functional or object-oriented, is used). While it works pretty well for small problems, it can become a major headache for bigger ones.

Note that the real challenge here is not on how to design the system to take a particular aspect into account: there is a huge design know-how in industry for that, often captured in the form of Design Patterns (see above). Taking into account more than one aspect as the same time is a little bit more tricky, but many large scale successful projects in industry are there to show us that engineers do ultimately manage to sort it out.

The real challenge in a product-line context is that the engineer wants to be able to change her mind on which version of which variant of any particular aspect she wants in the system. And she wants to do it cheaply, quickly and safely. For that, redoing by hand the tedious weaving of every aspect is not an option.

3.2.8. Model Driven Engineering

Usually in science, a model has a different nature that the thing it models ("do not take the map for the reality" as Sun Tse put it many centuries ago). Only in software and in linguistics a model has the same nature as the thing it models. In software at least, this opens the possibility to automatically derive software from its model. This property is well known from any compiler writer (and others), but it was recently be made quite popular with an OMG initiative called the Model Driven Architecture (MDA). This requires that models are no longer informal, and that the weaving process is itself described as a program (which is as a matter of facts an executable meta-model) manipulating these models to produce a detailed design that can ultimately be transformed to code or at least test suites.

The OMG has built a meta-data management framework to support the MDA. It is mainly based on a unique M3 “meta-meta-model” called the Meta-Object Facility (MOF) and a library of M2 meta-models, such as the UML (or SPEM for software process engineering), in which the user can base his M1 model.

The MDA core idea is that it should be possible to capitalize on platform-independent models (PIM), and more or less automatically derive platform-specific models (PSM) –and ultimately code– from PIM through model transformations. But in some business areas involving fault-tolerant, distributed real-time computations, there is a growing concern that the added value of a company not only lies in its know-how of the business domain (the PIM) but also in the design know-how needed to make these systems work in the field (the transformation to go from PIM to PSM). Reasons making it complex to go from a simple and stable business model to a complex implementation include:

- Various modeling languages used beyond UML,
- As many points of views as stakeholders,
- Deliver software for (many) variants of a platform,
- Heterogeneity is the rule,
- Reuse technical solutions across large product lines (e.g. fault tolerance, security, etc.),
- Customize generic transformations,
- Compose reusable transformations,
- Evolve and maintain transformations for 15+ years.

This wider context is now known as Model Driven Engineering.

4. Application Domains

4.1. Application Domains

From small embedded systems such as home automation products or automotive systems to medium sized systems such as medical equipment, office equipment, household appliances, smart phones; up to large Service Oriented Architectures (SOA), building a new application from scratch is no longer possible. Such applications reside in (group of) machines that are expected to run continuously for years without unrecoverable errors. Special care has then to be taken to design and validate embedded software, making the appropriate trade-off between various extra-functional properties such as reliability, timeliness, safety and security but also development and production cost, including resource usage of processor, memory, bandwidth, power, etc.

Leveraging ongoing advances in hardware, embedded software is playing an evermore crucial role in our society, bound to increase even more when embedded systems get interconnected to deliver ubiquitous SOA. For this reason, embedded software has been growing in size and complexity at an exponential rate for the past 20 years, pleading for a component based approach to embedded software development. There is a real need for flexible solutions allowing to deal at the same time with a wide range of needs (product lines modeling and methodologies for managing them), while preserving quality and reducing the time to market (such as derivation and validation tools).

We believe that building flexible, reliable and efficient embedded software will be achieved by reducing the gap between executable programs, their models, and the platform on which they execute, and by developing new composition mechanisms as well as transformation techniques with a sound formal basis for mapping between the different levels.

Reliability is an essential requirement in a context where a huge number of softwares (and sometimes several versions of the same program) may coexist in a large system. On one hand, software should be able to evolve very fast, as new features or services are frequently added to existing ones, but on the other hand, the occurrence of a fault in a system can be very costly, and time consuming. While we think that formal methods may help solving this kind of problems, we develop approaches where they are kept “behind the scene” in a global process taking into account constraints and objectives coming from user requirements.

Software testing is another aspect of reliable development. Testing activities mostly consist in trying to exhibit cases where a system implementation does not conform to its specifications. Whatever the efforts spent for development, this phase is of real importance to raise the confidence level in the fact that a system behaves properly in a complex environment. We also put a particular emphasis on on-line approaches, in which test and observation are dynamically computed during execution.

5. Software

5.1. Kermeta : Kernel Metamodeling

Participants: Olivier Barais, Jacques Falcou, Cyril Faucher, François Fouquet, Jean-Marc Jézéquel, Hajanirina Johary Rambelontsalama, Didier Vojtisek [correspondant].

Nowadays, object-oriented meta-languages such as MOF (Meta-Object Facility) are increasingly used to specify domain-specific languages in the model-driven engineering community. However, these meta-languages focus on structural specifications and have no built-in support for specifications of operational semantics. Triskell has developed the Kermeta language to explore the idea of using aspect-oriented modeling to add precise action specifications with static type checking and genericity at the meta level, and examine related issues and possible solutions.

Kermeta consists of an extension to the Essential Meta-Object Facilities (EMOF) 2.0 to support behavior definition. It provides an action language to specify the body of operations in metamodels. This action language is imperative and object-oriented.

Kermeta is used in several use cases:

- to give a precise semantic of the behavior of a metamodel which then can be simulated.
- to act as a model transformation language.
- to act as a constraint language.

The development environment built for the Kermeta language currently provides the following tools

- an interpreter and a compiler that allow a metamodel to be executed.
- text and graphical editors, fully integrated within Eclipse, with syntax highlighting, code autocompletion.
- an Eclipse outline view, which allows navigation through the whole model and metamodel.
- various import/export transformations such as ecore2kermeta (kermeta text), kermat2ecore, kermeta2xmi (xmi version of your kermeta metamodel), xmi2kermeta, xmi2ecore.

Developed as an open source software under the terms of the EPL (Eclipse Public License), it has been first deposited to the APP (Agence de Protection des Programmes) in October 2005.

Thanks to Kermeta it is possible to build various frameworks dedicated to domain specific metamodels. Those frameworks are organized into MDKs (Model Development Kits). For example, Triskell proposes MDKs to work with the following metamodels: Java5, UML2, RDL (requirements), Ecore, Traceability, ...Some of these MDKs (UML2, RDL) are advanced enough to constitute a complete application.

In 2010, we've release the version 1.4.1 of Kermeta with a strong focus on the new compiler to improve overall performances.

5.2. Kompose : Generic Model Composition Tool

Participants: Mickael Clavreul, Olivier Barais, Freddy Muñoz, Benoit Baudry [correspondant].

Kompose is a generic framework to support model composition. The core composition mechanism is implemented in Kermeta as a separate metamodel that can be specialized for a specific domain metamodel in order to easily define composition operators for that domain. The framework is made of a generic model element merge algorithms and a directive language. The specialization for a specific metamodel is done by defining appropriate signatures for the classes of this metamodel. As examples, Kompose currently includes specializations for class diagrams, database schemas and feature models as in [80]. Kompose has been developed in collaboration with CSU in the context of the MATT associate team.

5.3. Pramana: automatic synthesis of models

Participants: Sagar Sen, Benoit Baudry, Juan Cadavid, Xavier Dolques, Hajanirina Johary Rambelontsalama, Didier Vojtisek [correspondant].

Pramana is an open-source tool, which automatically generates valid instances of a metamodel. These instances can then be used for analysis, verification, simulation or validation of the metamodel. The core mechanism for model generation relies on the bounded constraint-solver of Alloy, a lightweight model checker developed at the MIT. Alloy is integrated in Kermeta to allow the generation of instances of Ecore or kmt metamodels. Pramana implements this integration through a series of transformations and analysis, all implemented in Kermeta.

5.4. Entimid: a Model-based Middleware for integrating IOT and IOS

Participants: Grégory Nain [correspondant], Olivier Barais, François Fouquet, Claire Diehl-watrin, Jean-Marc Jézéquel [correspondant].

There is a growing interest in leveraging Service Oriented Architectures (SOA) in domains such as home automation, automotive, mobile phones or e-Health. With the basic idea (supported in e.g. OSGi) that components provide services, it makes it possible to smoothly integrate the Internet of Things (IoT) with the Internet of Services (IoS). The paradigm of the IoS indeed offers interesting capabilities in terms of dynamicity and interoperability. However in domains that involve *things* (e.g. appliances), there is still a strong need for loose coupling and a proper separation between types and instances that are well-known in Component-Based approaches but that typical SOA fail to provide. Entimid is a framework to get the best of both worlds by augmenting SOA with a Component-Based approach. Entimid also manages the explosion of possible runtime system configurations (often called modes) and mode transitions of such systems using models and aspects. While Aspect-Oriented Modeling (AOM) was introduced to improve the modularity of software, we present how an AOM approach can be used to tame the combinatorial explosion of Dynamic Adaptive Systems built with Entimid. Using AOM techniques, we derive a wide range of modes by weaving aspects into an explicit model reflecting the runtime system. We use these generated modes to automatically adapt the system. We validate Entimid through a set of systems currently deployed in Rennes metropolis.

5.5. ART2: Models@Runtime for Dynamic Distributed Adaptive System

Participants: Grégory Nain, Olivier Barais [correspondant], François Fouquet, Brice Morin, Claire Diehl-watrin, Jean-Marc Jézéquel.

We are witnessing the emergence of new classes of application that are highly complex, inevitably distributed, and operate in heterogeneous and rapidly changing environments. Examples of such applications include those from pervasive and Grid computing domains. These systems are required to be adaptable, flexible, reconfigurable and, increasingly, self-managing. Such characteristics make systems more prone to failure when executing and thus the development and study of appropriate mechanisms for runtime validation and monitoring is needed. In the model-driven software development area, research effort has focused primarily on using models at design, implementation, and deployment stages of development. The use of model-driven techniques for validating and monitoring run-time behavior can also yield significant benefits. A key benefit is that models can be used to provide a richer semantic base for run-time decision-making related to system adaptation and other run-time concerns. For example, one can use models to help determine when a system should move from a consistent architecture to another consistent architecture. Model-based monitoring and management of executing systems can play a significant role as we move towards implementing the key self-* properties associated with autonomic computing. ART2, is a new models at runtime platform that support the development such systems using models as a first class entities.

6. New Results

6.1. Model Driven and Aspect Oriented Design

6.1.1. Models at Runtime

Participants: Brice Morin, Olivier Barais, Grégory Nain, François Fouquet, Noël Plouzeau, Mahmoud Ben Hassine, Jean-Marc Jézéquel.

Brice Morin's PhD thesis [14] presented a Model-Driven and Aspect-Oriented approach to tame the complexity of Dynamically Adaptive Systems. At design-time, we capture the different facets of a DAS (variability, environment/context, reasoning and architecture) using dedicated metamodels. Each feature of the variability model describing a DAS is refined into an aspect model. We leverage these design models at runtime to drive the dynamic adaptation process. Both the running system and its execution context are abstracted as models. Depending on the current context (model) a reasoner interprets the reasoning model to determine a well fitted selection of features. We then use Aspect-Oriented Modeling techniques to automatically compose the aspect models (associated to the selected features) together in order to automatically derive the corresponding architecture. This way, there is no need to specify the whole set of possible configurations at design-time: each configuration is automatically built when needed. We finally rely on model comparison to fully automate the reconfiguration process in order to actually adapt the running system, with no need to write low-level reconfiguration scripts. An important point is that models embedded at runtime are really mirrors of what really happens in the running system. It is however possible to work on copies of these models, independently of the running system and resynchronize these copies with the reality to actually adapt the running system. In other words, our approach makes it possible to perform offline activities such as continuous design or prediction, while the system is running, but independently from it.

Along the lines of the 2009 results on models at runtime, we have developed a new version of the Art metamodel. This version provides a simple yet powerful component metamodel. The new version, named Art2, extends and improves the existing Art metamodel in the following domains: specification and management of extrafunctional properties; flexible concurrency control features; distributed computation representation and management.

To evaluate the Art2 metamodel we are developing a realistic system. This system is a real time, distributed multi-user application for incident management, targeted at the firefighter command structure.

Triskell is involved in the european IP project named CHESS. This project is centered on an instrumented process that supports performance and reliability for embedded real-time systems. In the context of this project, we have investigated stochastic extensions to extrafunctional properties of software component architectures. In 2010 we designed an extension of the CHESS metamodel to allow for property specifications based on stochastic Petri nets. This extension is being integrated into the Art2 metamodel. This will enable the use of stochastic Petri net evaluators within the model at runtime architectures, allowing for better predictions of quantitative properties of reconfigured models.

6.1.2. Engineering of interactive systems

Participants: Arnaud Blouin, Olivier Barais, Jean-Marc Jézéquel.

- Active Operations on Collections: we propose a formalism and an implementation to dynamically link collections that compose models [35]. This formalism, applicable to different problems, is currently being used in the context of interactive systems to link data and their graphical representations [36].
- Improving modularity and usability of interactive systems with Malai: we propose to merge concepts from the HCI (Human-Computer Interaction) and the MDE domains for the conception of the interactive part of systems. We defined the interactive part as composed of: user interfaces (UI), instruments, interactions and actions. Such fragmentation of the interactive part in several atomic components improves the modularity of the system. This modularity is notably needed to facilitate the adaptation of interactive systems on context changes [38], [37].

6.1.3. Model-Driven Measurement

Participants: Benoit Baudry, Jean-Marc Jézéquel, Naouel Moha.

- Model-driven Generative Development of Measurement Software: we synthesize our contributions about an environment for the definition of metrics over a domain-specific modeling language and the automatic generation of a tool that performs these measures on instance of that language [25].
- DECOR: A Method for the Specification and Detection of Code and Design Smells: this contribution proposes an abstract language for the specification of design smells in object-oriented programs as well as a generative approach to derive tools that detect these smells [23] [79].

6.1.4. Model-Driven Information System Alignment

Participant: Jean-Marc Jézéquel.

- Business and Information System Alignment: A Formal Solution for Telecom Services [59]. The Enterprise Architecture process consists in designing the IS target architecture on several views, according to the company strategy. The business view represents the target organization of the considered company. The functional view focuses on the target functional architecture of the considered IS. We proposed a new formal solution to analyze the consistency between the target functional view and the target business view of telecom services. This solution is based on the definition of a strategic alignment of the target functional view with the target business view. This work has been carried out in cooperation with Telecom Bretagne (J. Simonin) and Univ. Luxembourg (Y. Le Traon).

6.1.5. Verification of dynamic systems

Participants: Benoit Baudry, Freddy Muñoz.

- Validation of reasoning engines and adaptation mechanisms for *self*-adaptive systems: this PhD thesis proposes a test generation strategy for reasoning engines as well as a contract mechanism for aspect-oriented dynamic adaptation [15].

6.1.6. Web services QoS analysis

Participants: Benoit Baudry, Sagar Sen.

- Variability Modeling and QoS Analysis of Web Services Orchestrations: we sample the variability space defined by composite web services in order to estimate global QoS. Sampling is based on pair-wise selection and experiments indicate this is a reasonable strategy [48].

6.1.7. Service-oriented self-adaptation framework

Participant: Johann Bourcier.

- Design, development and open-source contribution of a specific service-oriented component framework for autonomic applications. The design and development of an open and dynamic hierarchical autonomic management infrastructure to enable the creation of autonomic solutions in the pervasive home environment (to be published in 2011 [21]).

6.2. Model-Based Testing

6.2.1. Testing model transformations

Participants: Benoit Baudry, Sagar Sen.

- Barriers for systematic testing of model transformations: We surveyed the literature on that topic and identified important barriers for effective testing of transformations [17]. One of the barriers is about the construction of an oracle for these test cases, for which we made a proposal in [27].
- Automatic discovery of effective models: this PhD thesis proposes a constraint-based approach to automatically discover models that are effective for a specific purpose in a large modeling space [16]. One purpose for automatic model generation was to synthesize input models to test transformations.

6.2.2. Test generation for software product lines

Participants: Benoit Baudry, Sagar Sen.

- Automatic T-wise Test Case Generation for Software Product Lines: this contribution proposes 2 strategies to generate T-wise interaction test suites from a feature model, as well as a series of experiments to evaluate the relevance of the proposal [57].

6.2.3. Test selection in AOP

Participants: Benoit Baudry, Freddy Muñoz.

- Impact Analysis of Aspect Weaving on Test Cases: this contribution introduces an algorithm to statically detect the subset of test cases impacted by the introduction of one or more aspects in a base program [43].

6.2.4. Test emerging paradigms

Participant: Benoit Baudry.

- Introducing the question-learn-test-feedback pattern to investigate evolving software construction paradigms: this habilitation synthesizes a series of work that investigate software testing for new ways of building software. In particular, it raises a number of challenges and proposes a structured method for this type of research on software testing [12].

6.2.5. Model simulation

Participant: Benoit Combemale.

- Generative Technologies for Model Animation in the TopCased Platform: in this work we propose generative technologies that have been designed to ease the development of model animation tools inside the TopCased platform. These tools rely on an architecture for executable metamodel (i.e., a model execution metamodeling pattern) to bind the behavioral semantics of the modeling language [40].

6.3. Meta-Modeling

6.3.1. Modeling Model Transformations

Participants: Benoit Baudry, Benoit Combemale, Jean-Marc Jézéquel, Sagar Sen, Naouel Moha.

- Modeling Modeling Modeling: we propose a formal abstract syntax and a graphical concrete syntax for a language that captures relations between modeled things in order to reason and communicate about modeling activities [28].
- Reusable model transformations: we propose a conjunct use of model typing and metamodel pruning to ease the reuse of model transformations on instances of different metamodels [29].
- Evaluation of Kermeta on Graph Transformation Problems: we model a benchmark of transformations with Kermeta to evaluate different facets of the language (generecity, extensibility and performance) [24]

6.3.2. Model Driven Language Engineering

Participants: Olivier Barais, Benoit Combemale, Jean-Marc Jézéquel, Didier Vojtisek.

Traditional Language Engineering starts with the grammar of a language to produce a variety of tools for processing programs expressed in this language. Recently however, many new languages tend to be first defined through metamodels, i.e. models describing their abstract syntax. Relying on well tooled standards such as E-MOF, this approach makes it possible to readily benefit from a set of tools such as reflexive editors, or XML serialization of models. We have shown how Model Driven Engineering can easily complement these off-the-shelf tools to obtain a complete environment for such a language, including interpreter, compiler, pretty-printer and customizable editors.

6.3.3. Behavioral Semantics of Modeling Languages

Participants: Benoit Combemale, Jean-Marc Jézéquel, Benoit Baudry.

Formally Tracing Executions From an Analysis Tool Back to a Domain Specific Modeling Language's Operational Semantics: in this work, we propose a formal and operational framework for tracing results back (e.g., a program crash log, or a counterexample returned by a model checker) from execution and verification tools to an original DSML's syntax and operational semantics [70].

6.3.4. Model Driven Software Evolution

Participants: Naouel Moha, Régis Fleurquin, Olivier Barais, Jean-Marc Jézéquel.

Code and design smells are recurring design problems in software systems that must be identified to avoid their possible negative consequences on development and maintenance. In collaboration with the University of Montreal and the ADAM project team in Lille, we introduced an approach to automate the generation of detection algorithms from specifications written using a domain-specific language [23]. This language is defined from a thorough domain analysis. It allows the specification of smells using high-level domain-related abstractions. It allows the adaptation of the specifications of smells to the context of the analyzed systems.

We also studied the problem of *God Classes* (large classes which know too much or do too much in an OO design): how they arise, how prevalent they are, and whether they remain or they are removed as the systems evolve over time, through a number of versions. We showed how to detect the degree of godliness of classes automatically, distinguishing between those classes that are so by design (good code) from those that occurred by accident (bad code). This methodology can guide software quality teams in their efforts to implement prevention and correction mechanisms.

However, these kind of model analysis, or other model transformations such as refactorings specified for a given language cannot be readily reused for another language because their related metamodels may be structurally different. To solve this problem, we have discovered [29] an approach allowing the specification of generic model transformations, in particular refactorings, so that they can be applied to different metamodels. Our approach relies on two mechanisms: (1) an adaptation based mainly on the weaving of aspects; (2) the notion of model typing, an extension of object typing in the model-oriented context. We validated our approach by performing some experiments that consisted of specifying three well known refactorings (Encapsulate Field, Move Method, and Pull Up Method) and applying each of them onto three different metamodels (Java, MOF, and UML).

6.3.5. Executable Software Process Modeling

Participants: Benoit Combemale, Jean-Marc Jézéquel.

Describing and managing activities, resources and constraints of software development processes is a challenging goal for many organizations [30]. We proposed a comparison of UML-based Software Process Modeling Languages (SPMLs) [19].

One major advantage of executable software process models is that once defined, they can be simulated, checked and validated in short incremental and iterative cycles. This also makes them a powerful asset for important process improvement decisions such as resource allocation, deadlock identification and process management. We proposed a framework that combines Aspect and Model-Driven Engineering approaches in order to ensure process modeling, simulation and execution [18]. This framework is based upon UML4SPM, a UML2.0-based language for Software Process Modeling and Kermeta, an executable metaprogramming language. This work has been done in collaboration with Reda Bendraou from the UPMC.

7. Contracts and Grants with Industry

7.1. ITEA2 OPEES

Participants: Jean-Marc Jézéquel, Didier Vojtisek, Benoit Baudry, Benoit Combemale, Cédric Bouhours, Xavier Dolques.

OPEES is an ITEA2 project which goal is to build a community able to ensure long-term availability of innovative engineering technologies in the domain of software-intensive embedded systems. Its main benefits should be to perpetuate the methods and tools for software development, minimize ownership costs, ensure independence of development platform, integrate, as soon as possible, methodological changes and advances made in academic world, be able to adapt tools to the process instead of the opposite, take into account qualification constraints. In this purpose, OPEES relies on the Eclipse Modeling Project platform (EMF, GEF, GMF, OCL, UML2, ...) and on many available tools such as Kermeta.

The participation of Triskell into the OPEES project aims at industrializing both Kompose and MDE-test. Kompose is a model composition framework based on Aspect Oriented Software Development research results. Kompose allows a better modularization and separation of concerns when building tools around metamodels. It features powerful model-level pointcut specification and matching mechanisms allowing semantic based weaving of model level aspects. Kompose builds on and is nicely integrated with Kermeta, but it can also be used independently on any kind of EMF model or meta-model. MDE-test is a model transformation testing framework that makes it possible to synthesize input data (i.e. test models) for model transformations and check that the transformation behaves "correctly" on them.

Project duration: 2010-2012

Triskell budget share: 150 keuros

Project Coordinator: OBEO

Participants: AIRBUS, ADACORE, Anyware Technologies, Astrium Satellites, Atos Origin, CEA LIST, CNES, C-S, Dassault, EADS Astrium ST, ENAC, INPT-IRIT, INRIA (AtlanMod/EXPRESSO/TRISKELL), MBDA, OBEO, ONERA, Schneider Electric, Thales, Xipp

7.2. NeSSoS (Network of Excellence)

Participants: Jean-Marc Jézéquel, Benoit Baudry, Olivier-Nathanael Ben David, Olivier Barais.

The Network of Excellence on Engineering Secure Future Internet Software Services and Systems (NESSoS) aims at constituting and integrating a long lasting research community on engineering secure software-based services and systems.

The NESSoS engineering of secure software services is based on the principle of addressing security concerns from the very beginning in system analysis and design, thus contributing to reduce the amount of system and service vulnerabilities and enabling the systematic treatment of security needs through the engineering process. In light of the unique security requirements the Future Internet will expose, new results will be achieved by means of an integrated research, as to improve the necessary assurance level and to address risk and cost during the software development cycle in order to prioritize and manage investments. NESSoS will integrate the research labs involved; NESSoS will re-address, integrate, harmonize and foster the research activities in the necessary areas, and will increase and spread the research excellence. NESSoS will also impact training and education activities in Europe to grow a new generation of skilled researchers and practitioners in the area. NESSoS will collaborate with industrial stakeholders to improve the industry best practices and support a rapid growth of software-based service systems in the Future Internet.

The research excellence of NESSoS will contribute to increase the trustworthiness of the Future Internet by improving the overall security of software services and systems. This will support European competitiveness in this vital area.

Three INRIA EPIs are involved in NeSSoS: ARLES, CASSIS and Triskell. Triskell leads the research workpackage on design and architecture for secured future internet applications. In this context we will survey the state of the art on that area, evaluate architectures for adaptive security concerns and experiment with service-based and cloud applications in order to understand how adaptive security can improve the security of these future internet applications.

Project duration: 2010-2014

Triskell budget share: 100 keuros

Project Coordinator: Dr. Fabio Martinelli (Project Coordinator), Consiglio Nazionale delle Ricerche, Italy

Participants: University of Duisburg-Essen, Technical University Münich, Consiglio Nazionale delle Ricerche, The French National Institute for Research in Computer Science and Control, IMDEA, ETH, University of Malaga, SINTEF, Universita di Trento, KU Leuven, SIEMENS, ATOS

7.3. Artemis CHESS

Participants: Noël Plouzeau, Jean-Marc Jézéquel, Jacques Falcou, Viet-Hoa Nguyen.

CHESS is an Artemis project that seeks industrial-quality research solutions to problems of property-preserving component assembly in real-time and dependable embedded systems, and supports the description, verification, and preservation of non-functional properties of software components at the abstract level of component design as well as at the execution level. CHESS develops model-driven solutions, integrates them in component-based execution frameworks, assesses their applicability from the perspective of multiple domains (such as space, railways, telecommunications and automotive), and verifies their performance through the elaboration of industrial use cases.

In 2010 Triskell contributed to the definition of the component metamodel and to the design of model driven engineering process. Triskell contributes to the development of the CHESS model editor; the team is the main contributor of model transformation tools, by adapting its Kermeta platform to the Chess process and by contributing to the interconnection of external tools from industrial tool provider partners.

Project duration: 2009-2011

Triskell budget share: 400 keuros

Project budget: 6 M euros

Project Coordinator: INTECS

Participants: AICAS, Aonix, Atego ENEA, Ericsonn, Fraunhofer, FZI, GMV, INRIA (Triskell), INTECS, Thales Alenia Space, THALES Communications, UPM, University of Padua, X/Open

7.4. DiVA (IST)

Participants: Jean-Marc Jézéquel, Benoit Baudry, Brice Morin, Freddy Muñoz, Olivier Barais, Didier Vojtisek, Johann Bourcier.

The goal of DiVA is to provide a tool-supported methodology for managing dynamic variability of co-existing, co-dependent configurations in adaptive systems that span system administration and platform boundaries. Examples of such adaptive systems are communication infrastructure in rescue operations and mobile entertainment environments. This is addressed through a combination of aspect-oriented and model-driven techniques. DiVA explores how adaptation policies can be captured in the requirements, how aspects can model the variants used to adapt the system, how models can be kept at runtime to drive the adaptation and which validation techniques have to be developed in this context.

The Triskell team participates mainly in the definition of models that can drive the adaptation at runtime. The benefits of keeping models at runtime is to have an abstract view of the adaptation policies and mechanisms on which it is possible to reason (to check invariants, QoS properties, etc.) before actually adapting the running system. One important challenge tackled by Triskell is a mechanism to synchronize the running system with the model that has been adapted according to the changes in the environment. Triskell is also involved in the different validation tasks that occur when building such systems and when adapting these systems at runtime. An important issue for validation at design time is to select a subset of all possible configurations for testing. At design time, it is necessary to validate interactions between variants and to check that invariants on the system are satisfied.

Project duration: 2007-2010

Triskell budget share: 400 keuros

Project Coordinator: SINTEF

Participants: SINTEF, Uni. Lancaster, INRIA, Pure Systems, Thales IS, CAS.

7.5. S-Cube (Network of Excellence)

Participants: Jean-Marc Jézéquel, Noël Plouzeau, Olivier Barais, Grégory Nain, Sagar Sen, Maha Driss.

S-Cube, the Software Services and Systems Network, will establish an integrated, multidisciplinary, vibrant research community which will enable Europe to lead the software-services revolution, thereby helping shape the software-service based Internet which is the backbone of our future interactive society.

An integration of research expertise and an intense collaboration of researchers in the field of software services and systems are needed to address the following key problems:

- Research fragmentation: Current research activities are fragmented and each research community (e.g., grid computing or software engineering) concentrates mostly on its own specific techniques, mechanisms and methodologies. As a result the proposed solutions are not aligned with or influenced by activities in related research fields.
- Future Challenges: One challenge, as an example, is to build service-based systems in such a way that they can self-adapt while guaranteeing the expected level of service quality. Such an adaptation can be required due to changes in a system's environment or in response to predicted and unpredicted problems.

S-Cube will pursue the following objectives which will have a long-lasting impact on European research:

- Re-aligning, re-shaping and integrating research agendas of key European players from diverse research areas. By synthesizing and integrating diversified knowledge, a long-lasting foundation for steering research and for achieving innovation at the highest level will be achieved.
- Inaugurating a Europe-wide common program of education and training for researchers and industry. This will create a common culture that will have a profound impact on the future of the field.
- Establishing a pro-active mobility plan to enable cross-fertilization, which will foster the integration of research communities and the establishment of a common software services research culture.
- Establishing trust relationships with industry. Via European Technology Platforms (specifically NESSI) a catalytic effect in shaping European research, strengthening industrial competitiveness and addressing main societal challenges will be accomplished.
- Defining a broader research vision and perspective. This will shape the software-service based Internet of the future and will accelerate economic growth and improve the living conditions of European citizens.

Two INRIA project-teams participate to this NoE. Paris and Triskell. The Work in S-Cube clearly distinguishes between principles and methods for engineering and adapting service-based systems and the technology which is used to realize those systems while taking into account cross-cutting issues like Quality of Service (QoS) and SLA compliance. Consequently two joint research activities has been designed. Triskell mainly participates to the first one, which is concerned with engineering and adaptation methodologies for Service-based applications. It combines different research efforts form the requirements engineering discipline, the human computer interaction discipline and the software engineering, adaptation and testing disciplines. This year, Triskell mainly works on the adaptation framework for services in unifying runtime adaptation and design evolution for service-based systems.

Project duration: 2008-2012

Triskell budget share: 150 keuros

Project Coordinator: Prof. Dr. Klaus Pohl (Project Coordinator), University of Duisburg-Essen, Germany – Prof. Dr. Mike Papazoglou (Scientific Director), Tilburg University, The Netherlands

Participants: University of Duisburg-Essen, Tilburg University, City University London, Consiglio Nazionale delle Ricerche, Center for Scientific and Technological Research, The French National Institute for Research in Computer Science and Control, Lero - The Irish Software Engineering Research Centre, Politecnico di Milano, MTA SZTAKI - Computer and Automation Research Institute, Vienna University of Technology, Université Claude Bernard Lyon, University of Crete, Universidad Politécnica de Madrid, University of Stuttgart

7.6. Mopcom Hard (ANR)

Participants: Jean-Marc Jézéquel, Didier Vojtisek, Gilles Perrouin, Cyril Faucher.

Mopcom hard is an ANR project supported by the Competitivity Cluster “Images & réseaux” of Brittany. The project focuses on the use of model driven engineering for the development of embedded system typically based on system-on-chip (SOC). The project will produce a complete methodology and development environment dedicated to the domain.

In 2010, Triskell finishes the development process and the specification of precise metamodels (using Kermeta) for each steps of the process. Triskell also studied and produced tools using Kermeta Java compiler for the MARTE UML profile as it the main metamodel for several of these steps .

Project duration: 2007-2010 years

Triskell budget share: 101 keuros

Project Coordinator: Thales (TSA)

Participants: Thals Systmes Aroports, Thomson, Sodius, ENSIETA, LESTER, Supelec Rennes, INRIA

7.7. Mopcom Ingénierie (Competitivity Cluster I&R)

Participants: Jean-Marc Jézéquel, Didier Vojtisek, Olivier Barais, Mickael Clavreul.

Mopcom Ingénierie is a project of the Competitivity Cluster “Images & réseaux” of Brittany. The project focuses on the use of model driven engineering for the development of Software for Image domain. The project will produce a complete methodology and development environment dedicated to the domain.

In 2010, Triskell evaluates the proposed solution to easily integrate legacy systems with MDE in order to address the Thomson case study.

Project duration: 2008-2011 years

Triskell budget share: 150 keuros

Project Coordinator: Thales (TSA)

Participants: Thals Systmes Aroports, Thomson, Sodifrance, ENSIETA, INRIA, ENST Bretagne, Valoria, Orange Labs

7.8. ANR Movida

Participants: Olivier Barais, Jean-Marc Jézéquel.

Movida is an ANR project which goal is to provide a solution for modeling view in system engineering and to provide decision support for architects. Today, and likely for a long time to come, the complexity of software dominant systems is still growing and the variety of system classes tends to expand. From embedded systems which are required to cope with spare resources, to system of systems for which the evolvability and flexibility is key, requirements classes are expanding. In addition new concerns or more stringent existing concerns bring their extra complexity. They are environmental concerns, maintenance, repair and operation (MRO) concerns, supply management concerns etc. All of them play today an active or even sometime decisive role in the engineering decision process. The difficulty to embrace the whole complexity of the concerns and the difficulty to manage their inter-relations has raised the interest of the engineering community for "concerns driven" engineering. This is addressed today in the model driven engineering research community through the exploration of "viewpoint modelling" technologies. The aim of the MOVIDA project is to provide a support to model-driven viewpoint engineering through:

- Defining and specifying the underlying concepts that must be shared and used when implementing an engineering solution supporting viewpoint management.
- Providing a support to the definition of specific viewpoints, enabling their composition in a consistent whole that fits a specific project needs.
- Managing the consistency of an information bulk made of several views on a system which is accessed, modified and managed by different stakeholders during the system definition process.
- Applying decision-support tools to multi-viewpoint modeling frameworks so as to support architectural trade-offs.

Triskell mainly works this year on the state of the arts and the conceptual meta-model for viewpoint modeling.

Project duration: 2009-2011

Triskell budget share: 184 keuros

Number of person/years: 1,2

Project Coordinator: Thales

Participants: Thales, OBEO, Université Paris 6, INRIA (Triskell)

7.9. IDA

Participants: Olivier Barais, Grégory Nain.

IDA is a project which goal is to study how technologies can help people to stay as long as possible at home. Industrials, associations and public institutions of the metropolis of Rennes, are working together on the IDA project which aims to allow dependent people to stay at home as long as possible. Due to the large scale of the project, and the diversity of disabilities that have to be considered, the deployment context will be different for each equipped house. The technologies used will vary, in order to compensate handicaps or because a technology is already installed, and people do not want it to be removed. Moreover, the system installed in these houses will have to provide a remote access to the devices of the house, and transmit all the necessary information from the sensors of the house to a control center where information will be treated. INRIA helps in providing support for integrating devices and services and providing a solution for managing variability.

Project duration: 2008-2010

Triskell budget share: 10 keuros

Project Coordinator: ASSAD Pays de Rennes

Participants: http://www.ida-autonomie.fr/partenaires.php

7.10. Orange Labs

Participants: Jacques Simonin, Jean-Marc Jézéquel.

Since March 2006, we have a collaboration with Orange Labs (France Télécom R& D), Lannion on applying MDE techniques to telecom operator IS. In this context, Jean-Marc Jézéquel acts as Ph.D advisor for Mariano Belaunde and Slim Ben Hassine, all being senior Orange Labs engineers.

Project duration: 2006-2011

Triskell budget share: 25 keuros

7.11. EDF

Participants: Nicolas Sannier, Benoit Baudry.

Since October 2010, we have a collaboration with EDF R& D, Chatou. This project aims at investigating the application of metamodeling and model-driven engineering for modeling and analyzing requirement documents of control-command systems. The purpose of this modeling activity is to improve the global understanding of dependencies between requirements and their context and to use this knowledge for impact analysis in case of evolution. In this context, Benoit Baudry acts as Ph.D advisor for Nicolas Sannier.

Project duration: 2010-2013

Triskell budget share: 30 keuros

7.12. Kereval

Participants: Aymerick Hervieu, Benoit Baudry.

Since October 2010, we have a collaboration with Kereval, an SME specialized in software testing. In this project we investigate the selection and reuse of test cases for software product lines in the automotive domain. In this context, Benoit Baudry acts as Ph.D advisor for Aymeric Hervieu. Arnaud Gotlieb from the Celtique EPI acts as a co-advisor for the PhD, as well as Olivier Philippot from Kereval.

Project duration: 2010-2013
Triskell budget share: 15 keuros

7.13. Sodifrance

Participants: Emmanuelle Rouillé, Benoit Combemale, Olivier Barais, Jean-Marc Jézéquel.

Since October 2010, we have a collaboration with Sodifrance, Rennes. In this project we investigate the support (capitalization, definition, execution, and adaptation) of software processes in the context of model driven development (MDD). The purpose of this work is twofold:

- automate the tool configuration and the dynamic adaptation of MDD CASE tools.
- support an automated verification of models, according to the requirements for each activity of the process.

In this context, Jean-Marc Jézéquel acts as Ph.D advisor for Emmanuelle Rouillé, also supervised by Benoit Combemale and Olivier Barais.

Project duration: 2010-2013
Triskell budget share: 25 keuros

8. Other Grants and Activities

8.1. Technology Development Actions (ADT)

8.1.1. GALAXY

Participants: Olivier Barais, Mahmoud Ben Hassine, Jean-Marc Jézéquel, Didier Vojtisek.

Galaxy is a Technology Development Action (ADT) by INRIA which goal is to make INRIA a value-added player in the SOA arena, by designing and developing an Open Framework for Agile and Dynamic Software Architecture. This ADT will work for INRIA and INRIA's research project-teams direct benefit, and aims at pre-assembling technological bricks from various teams, projects and preparing them to be transferred through the open source software channel.

galaxy aims at providing an open SOA platform, enabling agility using dynamic architectures. galaxy will provide an integrated environment by assembling and leveraging INRIA's technologies. galaxy covers a large scope of functionalities, from design tooling to adaptable and extensible runtime, offering monitoring and management advanced functions.

Galaxy allows to design, deploy, run, monitor systems, following concepts and paradigms inherited from service-oriented, process and dynamic architectures, and offering a set of management functions for agile and dynamic systems. galaxy technologies are most of them compliant with the Eclipse and the SCA standards.

Galaxy, a platform unifying high standard adaptable, extensible and agile runtime, and agile monitoring and modeling capabilities, built on top of INRIA's technologies:

- component-based models and frameworks Fractal and GCM (Grid Component Model) promoted by the ETSI
- FraSCAti and ProActive Java platforms
- Eclipse STP-IM, SOA Intermediate Model, for enabling interoperability b/w SOA-related standards at design- and run-time, such as BPMN, BPEL, SCA and JBI
- Kermeta MDE technologies
- WildCAT extensible framework for context-aware applications

Triskell EPI mainly work in collaboration with ASCOLA to integrate Wildcat monitoring framework and Kermeta to the Galaxy platform.

Project duration: 2008-2010

Triskell budget share: One associated engineer shared with Ascola EPI

Project Coordinator: Alain Boulze Tuvalu INRIA Project.

Participants: ECOO, ADAM, ASCOLA, TUVALU, SARDES, OASIS, TRISKELL.

8.1.2. KerGekoz

Participants: Didier Vojtisek, Benoit Combemale, Olivier Barais, Clément Guy.

KerGekoz is a A Technology Development Action (ADT) by INRIA which goal is to improve the Gecos platform of Cairn EPI by applying MDE technologies from Triskell EPI.

Gecos platform is a compiler infrastructure for the conception *System on Chip*. Gecos integrates ASIP flow synthesis, automatic parallelisation and hardware synthesis (C to hardware).

This ADT focuses on

- consolidation of existing work,
- improvement of the reusability and maintainability by applying Kermeta MDE technologies.

Triskell EPI mainly works in collaboration with CAIRN to integrate Kermeta to the Gecos platform.

Project duration: 2010-2012

Triskell budget share: One associated engineer shared with CAIRN EPI

Project Coordinator: Steven Derrien, CAIRN INRIA Project.

Participants: CAIRN, TRISKELL.

8.2. National Initiatives

8.2.1. CNRS GDRs

The Triskell project is connected to the national academic community through a lightweight participation to several CNRS GDR (Groupement de Recherche).

- GDR ASR: Action IDM (on Model Driven Engineering) (http://www.actionidm.org)
- GDR GPL: Génie de la Programmation et du Logiciel (http://www-lsr.imag.fr/GPL), where Jean-Marc Jézéquel is a member of the scientific committee.

8.3. European Initiatives

8.3.1. ERCIM Working Group on Software Evolution

Numerous scientific studies of large-scale software systems have shown that the bulk of the total software-development cost is devoted to software maintenance. This is mainly due to the fact that software systems need to evolve continually to cope with ever-changing software requirements. Today, this is more than ever the case. Nevertheless, existing tools that try to provide support for evolution have many limitations. They are (programming) language dependent, not scalable, difficult to integrate with other tools, and they lack formal foundations.

The main goal of the proposed WG (http://w3.umh.ac.be/evol/) is to identify a set of formally-founded techniques and associated tools to support software developers with the common problems they encounter when evolving large and complex software systems. With this initiative, we plan to become a Virtual European Research and Training Centre on Software Evolution.

Triskell contributes to this working group on the following points:

- re-engineering and reverse engineering
- model-driven software engineering and model transformation
- impact analysis, effort estimation, cost prediction, evolution metrics
- traceability analysis and change propagation
- family and product-line engineering

8.4. International Initiatives

8.4.1. Standardization in Eclipse projects

In 2010, Triskell project participates to discussion around the creation of the MXF eclipse project (http://www.eclipse.org/proposals/mxf/). This new Eclipse Modeling Framework Technology (EMFT) sub project proposes ideas similar to those included in Kermeta and can be a good candidate for transferring some Kermeta concepts to an Eclipse project.

8.4.2. Standardization at OMG

In 2010, Triskell project participates to normalization actions at OMG (http://www.omg.org/). It was involved in the CVL Common Variability Language Response to RFP and was interested in the Analysis and Design group which promotes standard modeling techniques including UML and MOF.

8.4.3. Collaboration with foreign research groups

- Colorado State University (CSU), USA. In January 2006 we started a “Equipe associe” (a three year program for an associated team) called MATT between CSU and Triskell. In 2009 the Equipe associe has been renewed under the name MoCAA which stands for Model Composition Aspects and Analysis (see http://www.irisa.fr/triskell/matt for details). In 2010 Mickael Clavreul was at CSU from February to May, working on model composition with Robert France and on an empirical study with Jim Bieman. This visit was also funded by a fellowship from the Bretagne regional council. The CAIRN and the TRISKELL EPIs collaborate to investigate the benefits of MDE and generative techniques to improve the construction of tools for optimizing compilers. On the other hand each EPI is part of an EA with a group in CSU. This has triggered new collaborations between the SAL and Mélange groups in CSU as well as a new research direction that now involves the four research groups.

  Robert France visited Rennes in September 2010 and was a referee for Freddy Muñoz’s PhD thesis. Robert France also led the new proposal for the NSF-INRIA REUSSI program for US visiting students. This proposal has been accepted and the program will be operational in 2011.
- University of Luxembourg. Since 2009 Triskell is involved in a collaborative project called SPLIT: Combine Software Product Line and Aspect-Oriented Software Development (with Nicolas Guelfi and Jacques Klein), that is funded by both the PICS program of CNRS and the FNR of Luxembourg. This project is providing the background and the funding for Paul Istoan’s PhD thesis, done in co-tutelle between University of Rennes and University of Luxembourg. As an initial research result, we showed how aspects can be unwoven, based on a precise traceability metamodel dedicated to aspect model weaving.
- Budapest University of Technology and Economics Since 2008, Benoit Baudry is associate researcher in the IP project SENSORIA (Software Engineering for Service Oriented Overlay Computers). In this context, we collaborate with Daniel Varro from Budapest University of Technology and Economics on testing techniques for model transformations.

9. Dissemination

9.1. Teaching

The Triskell team bears the bulk of the teaching on Software Engineering at the University of Rennes 1, at the levels M1 (Project Management, OO Analysis and Design with UML, Design Patterns, Component Architectures and Frameworks, V&V) and M2 (Model driven Engineering, Aspect-Oriented Software Development, Software Product Lines, Component Based Software Development, etc.).

Each of Jean-Marc Jézéquel, Noël Plouzeau, Olivier Barais, Benoit Combemale, Naouel Moha and Johann Bourcier are teaching about 200 h in these domains, with Benoit Baudry teaching about 50h, for a grand total of about 1000 hours, including several courses at ENSTB and INSA and ENSAI Rennes, as well at the University of Toulouse.

Olivier Barais is the overall responsible for the Master2 Pro in Computer Science at the University of Rennes.

The Triskell team also receives several Master and summer trainees every year.

9.2. Animation of the scientific community

9.2.1. Journals

9.2.1.1. Jean-Marc Jézéquel

is an Associate Editor of the following journals:

- Journal on Software and System Modeling: SoSyM
- Journal of Systems and Software: JSS
- Journal of Object Technology: JOT

9.2.2. Examination Committees

9.2.2.1. Jean-Marc Jézéquel

was in the examination committee of the following PhD thesis and “Habilitation Diriger les Recherches”:

- Marius Bodgza (HDR), February 2010, université de Grenoble (referee);
- Andrew Jackson, February 2010, Trinity College Dublin, Ireland (referee);
- Roy Gronmo, February 2010, University of Oslo, Norway (referee);
- Cyril Ballagny, March 2010, université de pau (member);
- Sagar Sen, June 2010, Université de Rennes 1 (adviser);
- Régis Fleurquin (HDR), June 2010, Univ. Bretagne-Sud (member);
- Freddy Muñoz, September 2010, Université de Rennes 1 (adviser);
- Brice Morin, September 2010, Université de Rennes 1 (adviser);
- An Phung Khac, November 2010, Telecom Bretagne (president);
- Xavier Dolques, November 2010, université de Montpellier 2 (referee);
- Alix Mougeot, November 2010, université de Paris 6 (member);
- Benoit Baudry (HDR), December 2010, Université de Rennes 1 (member);
- Laurent Hubert, December 2010, Université de Rennes 1 (president);

9.2.2.2. Benoit Baudry

was in the examination committee of the following PhD thesis:

- Hakim Belhaouri, March 2010, Université de Paris 6 (member)
- Sagar Sen, June 2010, Université de Rennes 1 (member)
- Tejedinne Mouelhi, September 2010, Telecom Bretagne (member)
- Freddy Muñoz, September 2010, Université de Rennes 1 (member)

9.2.2.3. Olivier Barais

was in the examination committee of the following PhD thesis

- Fady Hamoui, December 2010, Université de Montpellier (member)
- Brice Morin, September 2010, Université de Rennes 1 (co-adviser)

9.2.3. Conferences

9.2.3.1. Jean-Marc Jézéquel

has been a member of the program committee of the following conferences:

- ServiceWave 2010 Ghent, Belgium, 13th - 15th December 2010
- SEAA 2010 The 36th EUROMICRO Conference on Software Engineering and Advanced Applications, Lille, France, 1 - 3 Sept. 2010
- ICSE 2010 The 32nd International Conference on Software Engineering, Cape Town, South Africa, 2 - 8 May 2010 (Industrial Track)
- 5th International Workshop Models at runtime In conjunction with MODELS 2010 - Oslo, Norway, October 3-8, 2010

9.2.3.2. Benoit Baudry

has been a member of the program committee of the following conferences:

- PC chair for SC'2010, the 9th International Conference on Software Composition, Malaga, Spain, June 2010
- MODELS 2010 The 13th International Conference on Model Driven Engineering Languages and Systems Oslo, NO, October 2010
- IEEE ICST'10 The 3rd International Conference on Software Testing Verification and Validation, Paris, France, April 2010
- MoDeVVa 2010, 7th international workshop on Model design and Validation at MODELS'10, Oslo, NO, October 2010
- Mutation'10 workshop at ICST'10, Paris, France, April 2010

9.2.3.3. Olivier Barais

has been a member of the program committee of the following conferences:

- The 10th IEEE International Conference on Computer and Information Technology (CIT'10), Bradford, UK, July, 2010
- The First International Workshop on Composition: Objects, Aspects, Components, Services and Product Lines (Composition& Variability 2010) Rennes & Saint Malo, France, March 2010
- 3eme Confrence Francophone sur les Architectures Logicielles (CAL'10), March 2010
- 3nd Workshop on Context-aware Adaptation Mechanisms for Pervasive and Ubiquitous Services, (CAMPUS'10), June 2010. Amsterdam, Netherlands.

9.2.3.4. Noël Plouzeau

has been a member of the program committee of the following conferences and workshops:

- Component Based Software Engineering (CBSE), march 2010.
- Future Internet Symposium, september 2010.

9.2.3.5. Benoit Combemale

has been a member of the program committee of the following conferences and workshops:

- FML 2010, workshop on Formalization of Modeling Languages at ECOOP, june, 2010.
- SafeModels@IDM 2010, 6èmes journées sur l'Ingénierie Dirigée par les Modèles, 2010.

9.2.4. Workshops, Tutorials and Keynotes

J.-M. Jézéquel gave invited talks at SINTEF (Norway), the University of Luxembourg, and gave a keynote address at the Neptune day in Toulouse. He also gave a tutorial on Model Driven Language Engineering with Kermeta at the College de Polytechnique and at MODELS'10.

Olivier Barais gave an invited talk on EJCP'2010

Benoit Combemale gave a talk on Hyper-Agility at the Agile Tour Rennes 2010, and gave an invited talks on model execution and verification at the Colorado State University (USA).

9.3. Miscellaneous

- J.-M. Jézéquel is Deputy Director of MATISSE Doctoral School. He is head of the Language and Software Engineering Department at Irisa. He is appointed to the board of the Committee of Projects of INRIA Rennes. He is a member of the Steering Committee of the AOSD and the MODELS Conferences series. He is a member of the Scientific Committee of the GDR GPL of CNRS. He belongs to the evaluation committee of the SIO division of DGA (Direction Générale de l'Armement). He is a Member of the Architecture Board of the MDDi Eclipse project. He participated to the creation of IFIP WG 10.2 on Embedded Systems. He is a member of the Advisory Board of the NSF REMODD Project (Repository for Model Driven Development). He is General Chair of the 9th International Conference on Aspect-Oriented Software Development AOSD 2010 to be held in Rennes and Saint-Malo on March 15-19, 2010.
- Benoit Baudry is on the steering committee of the IEEE International Conference on Software Testing Verification and Validation. He has been the local organizing chair for AOSD'10.

10. Bibliography

Major publications by the team in recent years

[1] B. BAUDRY, F. FLEUREY, J.-M. JÉZÉQUEL, Y. LE TRAON. *Automatic Test Cases Optimization: a Bacteriologic Algorithm*, in "IEEE Software", March 2005, vol. 22, n° 2, p. 76–82.

[2] B. BAUDRY, S. GHOSH, F. FLEUREY, R. FRANCE, Y. LE TRAON, J.-M. MOTTU. *Barriers to Systematic Model Transformation Testing*, in "Communications of the ACM journal", Jun 2010, vol. 53, n° 6, 10 pages, http://hal.inria.fr/inria-00542747_v1.

[3] J.-M. JÉZÉQUEL. *Real Time Components and Contracts*, in "Model Driven Engineering for Distributed Real Time Embedded Systems", Hermes Science Publishing Ltd, London, 2005.

[4] J.-M. JÉZÉQUEL. *Model Driven Design and Aspect Weaving*, in "Journal of Software and Systems Modeling (SoSyM)", may 2008, vol. 7, n^{o} 2, p. 209–218.

[5] J. KLEIN, F. FLEUREY, J.-M. JÉZÉQUEL. *Weaving Multiple Aspects in Sequence Diagrams*, in "Transactions on Aspect-Oriented Software Development (TAOSD)", 2007, vol. LNCS 4620, p. 167-199, http://www.irisa.fr/triskell/publis/2007/Klein07a.pdf.

[6] Y. LE TRAON, B. BAUDRY, J.-M. JÉZÉQUEL. *Design by Contract to improve Software Vigilance*, in "IEEE Trans. on Software Engineering", August 2006, vol. 32, n^{o} 8, p. 571–586.

[7] B. MORIN, O. BARAIS, J.-M. JÉZÉQUEL, F. FLEUREY, A. SOLBERG. *Models at Runtime to Support Dynamic Adaptation*, in "IEEE Computer", October 2009, p. 46-53, http://www.irisa.fr/triskell/publis/2009/Morin09f.pdf.

[8] C. NEBUT, F. FLEUREY, Y. LE TRAON, J.-M. JÉZÉQUEL. *Automatic Test Generation: A Use Case Driven Approach*, in "IEEE Trans. on Software Engineering", March 2006, vol. 32, n^{o} 3, p. 140–155.

[9] J. STEEL, J.-M. JÉZÉQUEL. *On Model Typing*, in "Journal of Software and Systems Modeling (SoSyM)", December 2007, vol. 6, n^{o} 4, p. 401–414, http://www.irisa.fr/triskell/publis/2007/Steel07a.pdf.

[10] G. SUNYÉ, A. LE GUENNEC, J.-M. JÉZÉQUEL. *Using UML Action Semantics for Model Execution and Transformation*, in "Information Systems, Elsevier", July 2002, vol. 27, n^{o} 6, p. 445–457.

[11] T. ZIADI, J.-M. JÉZÉQUEL. *Product Line Engineering with the UML: Deriving Products*, in "Software Product Lines - Research Issues in Engineering and Management", T. KÄKÖLÄ, J. C. DUEÑAS (editors), Springer Verlag, 2006, n^{o} ISBN: 978-3-540-33252-7, p. 557–586.

Publications of the year

Doctoral Dissertations and Habilitation Theses

[12] B. BAUDRY. *Question-learn-test-feedback pattern to test emerging software construction paradigms*, Université Européenne de Bretagne, December 2010, Habilitation à Diriger des Recherches, http://hal.inria.fr/tel-00553854/en.

[13] R. FLEURQUIN. *Des langages pour améliorer le développement et la maintenance des logiciels à base de composants*, Université de Bretagne Sud and Université Européenne de Bretagne, July 2010, Habilitation à Diriger des Recherches, http://hal.inria.fr/tel-00511595/en.

[14] B. MORIN. *Modèles à la Conception et à l'Exécution pour Gérer la Variability Dynamique*, Université Rennes 1, September 2010, http://hal.inria.fr/tel-00538548/en.

[15] F. MUNOZ. *Validation des moteurs de raisonnement et mécanismes d'adaptation pour les systèmes auto-adaptatifs*, Université Rennes 1, September 2010, http://hal.inria.fr/tel-00538565/en.

[16] S. SEN. *Découverte automatique de modèles effectifs*, Université Rennes 1, June 2010, http://hal.inria.fr/tel-00507530/en.

Articles in International Peer-Reviewed Journal

[17] B. BAUDRY, S. GHOSH, F. FLEUREY, R. FRANCE, Y. LE TRAON, J.-M. MOTTU. *Barriers to Systematic Model Transformation Testing*, in "Communications of the ACM", 2010, vol. 53, n° 6, http://hal.inria.fr/inria-00542747/en.

[18] R. BENDRAOU, J.-M. JÉZÉQUEL, F. FLEUREY. *Achieving Process modeling and Execution through the Combination of Aspect and Model-Driven Engineering Approaches*, in "Journal of Software Maintenance and Evolution", 2010, http://hal.inria.fr/inria-00538462/en.

[19] R. BENDRAOU, J.-M. JÉZÉQUEL, M.-P. GERVAIS, X. BLANC. *A Comparison of Six UML-Based Languages for Software Process Modeling*, in "IEEE Transactions on Software Engineering", 2010, http://hal.inria.fr/inria-00477520/en.

[20] A. BEUGNARD, J.-M. JÉZÉQUEL, N. PLOUZEAU. *Contract Aware Components, 10 years after*, in "Electronic Proceedings in Theoretical Computer Science", October 2010, n° 37, p. 86-100, http://hal.inria.fr/inria-00554209/en.

[21] J. BOURCIER, A. DIACONESCU, P. LALANDA, M. JULIE. *AutoHome: an Autonomic Management Framework for Pervasive Home Applications*, in "ACM Transactions on Autonomous and Adaptive Systems (TAAS)", 2010, http://hal.inria.fr/inria-00554197/en.

[22] Y. JAMOUSSI, M. DRISS, J.-M. JÉZÉQUEL, H. HAJJAMI BEN GHÉZALA. *QoS Assurance for Service-Based Applications Using Discrete-Event Simulation*, in "IJCSI International Journal of Computer Science Issues", 2010, vol. 7, n° 6, http://hal.inria.fr/inria-00555041/en.

[23] N. MOHA, Y.-G. GUÉHÉNEUC, L. DUCHIEN, A.-F. LE MEUR. *DECOR: A Method for the Specification and Detection of Code and Design Smells*, in "IEEE Transactions on Software Engineering", 2010, vol. 36, n° 1, p. 20–36, http://hal.inria.fr/inria-00538476/en.

[24] N. MOHA, S. SEN, C. FAUCHER, O. BARAIS, J.-M. JÉZÉQUEL. *Evaluation of Kermeta on Graph Transformation Problems*, in "International Journal on Software Tools for Technology Transfer (STTT)", 2010, http://hal.inria.fr/inria-00477521/en.

[25] M. MONPERRUS, J.-M. JÉZÉQUEL, B. BAUDRY, J. CHAMPEAU, B. HOELTZENER. *Model-driven Generative Development of Measurement Software*, in "Software and Systems Modeling (SoSyM)", 2010, vol. tba, http://hal.inria.fr/inria-00504670/en.

[26] M. MONPERRUS, B. LONG, J. CHAMPEAU, B. HOELTZENER, G. MARCHALOT, J.-M. JÉZÉQUEL. *Model-driven Architecture of a Maritime Surveillance System Simulator*, in "Systems Engineering", 2010, vol. 13, n° 3, p. 290-297 [*DOI :* 10.1002/SYS.20149], http://hal.inria.fr/inria-00538455/en.

[27] J.-M. MOTTU, B. BAUDRY, Y. LE TRAON. *Construction de tests qualifiés de transformations de modèles*, in "Technique et Science Informatiques (TSI)", May 2010, vol. 29 (2010), p. pp 537-569, http://hal.inria.fr/hal-00483585/en.

[28] P.-A. MULLER, F. FONDEMENT, B. BAUDRY, B. COMBEMALE. *Modeling Modeling Modeling*, in "SOSYM", 2010, http://hal.inria.fr/inria-00538459/en.

[29] S. SEN, N. MOHA, V. MAHÉ, O. BARAIS, B. BAUDRY, J.-M. JÉZÉQUEL. *Reusable model transformations*, in "Software and Systems Modeling (SoSyM)", 2010, vol. tba, p. 346-379, http://hal.inria.fr/inria-00542766/en.

[30] C. TIBERMACINE, R. FLEURQUIN, S. SADOU. *A family of languages for architecture constraint specification*, in "Journal of Systems and Software", May 2010, vol. 83, n^o 5, p. 815-831, http://hal.inria.fr/hal-00498761/en.

Articles in National Peer-Reviewed Journal

[31] J.-M. JÉZÉQUEL. *Ingénierie Dirigé par les Modèles : du design-time au runtime*, in "Génie Logiciel - Ingéniérie dirigée par les modèles", 2010, n^o 93, http://hal.inria.fr/inria-00504666/en.

[32] J. SIMONIN, P. PICOUET, J.-M. JÉZÉQUEL. *Conception fonctionnelle de services d'entreprise fondée sur l'alignement entre coeur de métier et Système d'Information*, in "Ingénierie des Systèmes d'Information", 2010, http://hal.inria.fr/inria-00477524/en.

International Peer-Reviewed Conference/Proceedings

[33] F. ANDRÉ, E. DAUBERT, G. GAUVRIT. *Towards a Generic Context-Aware Framework for Self-Adaptation of Service-Oriented Architectures*, in "5th International Conference on Internet and Web Applications and Services", Barcelona, Espagne, May 2010, p. 309-314 [*DOI :* 10.1109/ICIW.2010.52], http://hal.inria.fr/inria-00470487/en.

[34] B. BAUDRY, P. BAZEX, J.-C. DALBIN, P. DHAUSSY, H. DUBOIS, C. PERCEBOIS, E. POUPART, L. SABATIER. *Trust in MDE Components: the DOMINO Experiment*, in "SD4RCES workshop in conjunction with SAFECOMP 2010", Vienna, Autriche, 2010, http://hal.inria.fr/inria-00555044/en.

[35] O. BEAUDOUX, A. BLOUIN, O. BARAIS, J.-M. JÉZÉQUEL. *Active Operations on Collections*, in "ACM/IEEE 13th International Conference on Model Driven Engineering Languages and Systems (MODELS'10)", Oslo, Norway, Norvège, 2010, p. 91–105, http://hal.inria.fr/inria-00542763/en.

[36] O. BEAUDOUX, A. BLOUIN. *Linking Data and Presentations: from Mapping to Active Transformations*, in "DocEng'10: Proceedings of the 2010 ACM symposium on Document engineering", Manchester, England, Royaume-Uni, 2010, http://hal.inria.fr/inria-00504671/en.

[37] O. BEAUDOUX, A. BLOUIN, J.-M. JÉZÉQUEL. *Using Model Driven Engineering technologies for building authoring applications*, in "DocEng'10: Proceedings of the 2010 ACM symposium on Document engineering", Manchester, England, Royaume-Uni, 2010, http://hal.inria.fr/inria-00504672/en.

[38] A. BLOUIN, O. BEAUDOUX. *Improving modularity and usability of interactive systems with Malai*, in "EICS'10: Proceedings of the 2nd ACM SIGCHI symposium on Engineering interactive computing systems", Berlin, Germany, Allemagne, 2010, http://hal.inria.fr/inria-00477627/en.

[39] M. CLAVREUL, O. BARAIS, J.-M. JÉZÉQUEL. *Integrating Legacy Systems with MDE*, in "ICSE'10: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering and ICSE Workshops", Cape Town, Afrique Du Sud, 2010, vol. 2, p. 69–78, http://hal.inria.fr/inria-00504669/en.

[40] X. CRÉGUT, B. COMBEMALE, M. PANTEL, R. FAUDOUX, J. PAVEI. *Generative technologies for model animation in the TopCased platform*, in "6th European Conference on Modelling Foundations and Applications

(ECMFA 2010)", Paris, France, Afrique Du Sud, Lecture Notes in Computer Science (LNCS), Springer, 2010, vol. 6138, p. 90–103, http://hal.inria.fr/inria-00504667/en.

[41] E. CUNHA DE ALMEIDA, J. E. MARYNOWSKI, G. SUNYÉ, Y. LE TRAON, P. VALDURIEZ. *Efficient Distributed Test Architectures for Large-Scale Systems*, in "International Conference on Testing Software and Systems", Natal, Brésil, A. PETRENKO, A. DA SILVA SIMÃO, J. C. MALDONADO (editors), LNCS, November 2010 [*DOI :* 10.1007/978-3-642-16573-3_13], http://hal.inria.fr/inria-00541696/en.

[42] E. CUNHA DE ALMEIDA, J. E. MARYNOWSKI, G. SUNYÉ, P. VALDURIEZ. *PeerUnit: a framework for testing peer-to-peer systems*, in "ASE", Antwerp, Belgique, C. PECHEUR, J. ANDREWS, E. D. NITTO (editors), ACM, 2010, p. 169-170 [*DOI :* 10.1145/1858996.1859030], http://hal.inria.fr/inria-00541699/en.

[43] R. DELAMARE, F. MUNOZ, B. BAUDRY, Y. LE TRAON. *Vidock: a Tool for Impact Analysis of Aspect Weaving on Test Cases*, in "International Conference on Testing Software and Systems", Natal, Brazil, Brésil, IFIP, 2010, http://hal.inria.fr/inria-00555069/en.

[44] M. DRISS, N. MOHA, Y. JAMOUSSI, J.-M. JÉZÉQUEL, H. HAJJAMI BEN GHÉZALA. *A Requirement-Centric Approach to Web Service Modeling, Discovery, and Selection*, in "ICSOC International Conference on Service Oriented Computing", San Francisco, California, États-Unis, Springer, 2010, http://hal.inria.fr/inria-00555042/en.

[45] F. FOUQUET, O. BARAIS, J.-M. JÉZÉQUEL. *Building a Kermeta Compiler using Scala: an Experience Report.*, in "Workshop Scala Days 2010", Lausanne, Suisse, EPFL, 2010, http://hal.inria.fr/inria-00542760/en.

[46] G. GAUVRIT, E. DAUBERT, F. ANDRÉ. *SAFDIS: A Framework to Bring Self-Adaptability to Service-Based Distributed Applications*, in "36th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA)", Lille, France, September 2010, p. 211–218 [*DOI :* 10.1109/SEAA.2010.25], http://hal.inria.fr/inria-00480221/en.

[47] M. GOUYETTE, O. BARAIS, J. LE NOIR, J.-M. JÉZÉQUEL. *Managing variability in multi-views engineering : A live demo*, in "Journée Lignes de Produits", Paris, France, France, 2010, http://hal.inria.fr/inria-00538466/en.

[48] A. KATTEPUR, S. SEN, B. BAUDRY, A. BENVENISTE, C. JARD. *Variability Modeling and QoS Analysis of Web Services Orchestrations*, in "International Conference on Web Services", Miami, FL, USA, États-Unis, IEEE, 2010, http://hal.inria.fr/inria-00561164/en.

[49] V. LE GLOAHEC, R. FLEURQUIN, S. SADOU. *Good Architecture = Good (ADL + Practices)*, in "6th International Conference on the Quality of Software Architectures (QoSA'10)", Prague, Tchèque, République, Lecture Notes in Computer Science, Springer, June 2010, vol. 6093, p. 167-182, http://hal.inria.fr/hal-00498782/en.

[50] V. LE GLOAHEC, R. FLEURQUIN, S. SADOU. *Good Practices as a Quality-Oriented Modeling Assistant*, in "QSIC'10: Proceedings of the 10th International Conference on Quality Software", Zhangjiajie, China, Chine, IEEE, 2010, p. 345-348, http://hal.inria.fr/inria-00504665/en.

[51] Y. LE TRAON, T. MOUELHI, F. FLEUREY, B. BAUDRY. *Language-specific vs. language-independent approaches: embedding semantics on a metamodel for testing and verifying access control policies*, in

"Workshop on Quality of Model-Based Testing (QuoMBaT)", Paris, France, 2010, http://hal.inria.fr/hal-00498383/en.

[52] V. MAHÉ, B. COMBEMALE, J. CADAVID. *Crossing Model Driven Engineering and Agility: Preliminary Thought on Benefits and Challenges*, in "3rd Workshop on Model-Driven Tool & Process Integration, in conjunction with ECMFA 2010", Paris, France, France, 2010, http://hal.inria.fr/inria-00538460/en.

[53] B. MORIN, F. FLEUREY, O. BARAIS, J.-M. JÉZÉQUEL. *Aspect-Oriented Modeling to Support Dynamic Adaptation*, in "Forum Demo at AOSD'10", Rennes and St Malo, France, France, 2010, http://hal.inria.fr/inria-00504664/en.

[54] B. MORIN, J. KLEIN, J. KIENZLE, J.-M. JÉZÉQUEL. *Flexible Model Element Introduction Policies for Aspect-Oriented Modeling*, in "13rd ACM/IEEE International Conference on Model-Driven Engineering Languages and Systems (MODELS 2010)", Oslo, Norway, Norvège, 2010, http://hal.inria.fr/inria-00538458/en.

[55] B. MORIN, T. MOUELHI, F. FLEUREY, Y. LE TRAON, O. BARAIS, J.-M. JÉZÉQUEL. *Security-Driven Model-Based Dynamic Adaptation*, in "25nd IEEE/ACM International Conference on Automated Software Engineering (ASE 2010)", Antwerp, Belgium, Belgique, 2010, http://hal.inria.fr/inria-00538500/en.

[56] G. NAIN, F. FOUQUET, B. MORIN, O. BARAIS, J.-M. JÉZÉQUEL. *Integrating IoT and IoS with a Component-Based approach*, in "Procedings of the 36th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA 2010)", Lille, France, France, 2010, http://hal.inria.fr/inria-00538469/en.

[57] G. PERROUIN, S. SEN, J. KLEIN, B. BAUDRY, Y. LE TRAON. *Automatic and Scalable T-wise Test Case Generation Strategies for Software Product Lines*, in "International Conference on Software Testing", Paris, France, France, Springer Lecture Notes in Computer Science (LNCS), 2010, http://hal.inria.fr/inria-00477523/en.

[58] J. SIMONIN, E. BERTIN, Y. LE TRAON, J.-M. JÉZÉQUEL, N. CRESPI. *Business and Information System Alignment: A Formal Solution for Telecom Services*, in "International Conference on Software Engineering Advances", Nice, France, IARA, 2010, http://hal.inria.fr/inria-00542752/en.

[59] J. SIMONIN, E. BERTIN, Y. LE TRAON, J.-M. JÉZÉQUEL, N. CRESPI. *Business and information system alignment: a formal solution for telecom services*, in "ICSEA 2010 : The Fifth International Conference on Software Engineering Advances", Nice, France, CPS, 2010, p. 278-283 [*DOI :* 10.1109/ICSEA.2010.49], http://hal.inria.fr/hal-00523087/en.

[60] C. TIBERMACINE, S. SAKHRAOUI, V. LE GLOAHEC, R. FLEURQUIN, S. SADOU. *Towards an Automation of Software Evolution Good Practices*, in "SEKE'10: Proceedings of the 22nd International Conference on Software Engineering and Knowledge Engineering", Redwood City, USA, États-Unis, Knowledge Systems Institute, 2010, p. 1-6, http://hal.inria.fr/inria-00504663/en.

National Peer-Reviewed Conference/Proceedings

[61] P. ALBERT, M. BLAY-FORNARINO, P. COLLET, B. COMBEMALE, S. DUPUY-CHESSA, A. FRONT, A. GROST, P. LAHIRE, X. LE PALLEC, L. LEDRICH, T. NODENOT, A.-M. PINNA-DERY, S. RUSINEK. *End-User Modelling*, in "Défis du Génie de la Programmation et du Logiciel, journées nationales du GDR GPL", Pau, France, France, 2010, http://hal.inria.fr/inria-00504668/en.

Workshops without Proceedings

[62] F. André, E. Daubert, G. Nain, B. Morin, O. Barais. *F4Plan: An Approach to build Efficient Adaptation Plans*, in "7th International ICST Conference on Mobile and Ubiquitous Systems (MobiQuitous)", Sydney, Australie, December 2010, short paper, http://hal.inria.fr/hal-00548736/en.

Scientific Books (or Scientific Book chapters)

[63] *Proceedings of the 9th Int. Conference on Aspect-Oriented Software Development*, ACM, March 2010, 230, http://hal.inria.fr/hal-00467398/en.

[64] F. André, V. Andrikopoulos, O. Barais, L. Baresi, G. Baryannis, S. Benbernou, I. Brandic, A. Bucchiarone, C. Cappiello, C. Di Napoli, E. Di Nitto, M. Driss, J. Carlos Dueñas, S. Dustdar, G. Gauvrit, M. Carro, N. Carroll, M. Comuzzi, F. Cuadrado, O. Danylevych, E. Daubert, M. Giordano, Q. Gu, M.-S. Hacid, F. Hantry, R. Haque, W.-J. Van Den Heuvel, D. Karastoyanova, R. Kazhamiakin, A. Kertész, K. Kritikos, A. Mocci, Z. Németh, C. Nikolau, M. Papazoglou, M. Parkin, J.-L. Pazat, S. Lane, W. Lamersdorf, F. Leymann, P. Leitner, V. Mazza, A. Metzger, B. Pernici, P. Plebani, K. Pohl, H. Psaier, W. Renz, I. Richardson, F. Rosenberg, F. Silvestri, J. Sudeikat, M. Uhlig, B. Wetzstein, E. Whelan. , M. Papazoglou, K. Pohl, M. Parkin, A. Metzger (editors)*Service Research Challenges and Solutions for the Future Internet*, Lecture Notes in Computer Science, Springer, 2010, vol. 6500 [*DOI :* 10.1007/978-3-642-17599-2], http://hal.inria.fr/inria-00559100/en.

[65] C. Bouhours. *Les Patrons Abimes - Détection, Explications et Restructuration*, Editions Universitaires Europeennes, 2010, http://hal.inria.fr/inria-00555362/en.

[66] B. Combemale. *Simulation et vérification de modèle par métamodélisation executable*, éditions Universitaires Européennes, 2010, http://hal.inria.fr/inria-00542740/en.

[67] J.-M. Jézéquel, O. Barais, F. Fleurey. *Model Driven Language Engineering with Kermeta*, in "3rd Summer School on Generative and Transformational Techniques in Software Engineering", J. M. Fernandes, R. Lammel, J. Saraiva, J. Visser (editors), LNCS 6491, Springer, 2010, http://hal.inria.fr/inria-00538461/en.

Books or Proceedings Editing

[68] A. Capiluppi, A. Cleve, N. Moha (editors). *Proceedings of the International Workshop on Principles of Software Evolution and ERCIM Workshop on Software Evolution (IWPSE-EVOL'2010)*, International Conference Proceedings Series, ACM, September 2010, 98, http://hal.inria.fr/inria-00514773/en.

Research Reports

[69] B. Combemale, X. Crégut, M. Pantel. *A Design Pattern for Executable DSML*, INRIA, November 2010, http://hal.inria.fr/inria-00540648/en.

[70] V. Rusu, L. Gonnord, B. Combemale. *Formally Tracing Executions From an Analysis Tool Back to a Domain Specific Modeling Language's Operational Semantics*, INRIA, October 2010, n^{o} RR-7423, http://hal.inria.fr/inria-00526561/en.

Other Publications

[71] P. LAHIRE, G. GEORG, M. OUSSALAH, J. WHITTLE, N. MOHA, S. VAN BAELEN. *Composition&Variability'2010. First International Workshop on Composition: Objects, Aspects, Components, Services and Product Lines*, March 2010, http://hal.inria.fr/hal-00484507/en.

References in notes

[72] A. BEUGNARD, J.-M. JÉZÉQUEL, N. PLOUZEAU, D. WATKINS. *Making Components Contract Aware*, in "IEEE Computer", July 1999, vol. 13, n^o 7.

[73] G. BOOCH. *Object-Oriented Analysis and Design with Applications*, 2nd, Benjamin Cummings, 1994.

[74] E. GAMMA, R. HELM, R. JOHNSON, J. VLISSIDES. *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison Wesley, 1995.

[75] M. JACKSON. *System Development*, Prentice-Hall International, Series in Computer Science, 1985.

[76] J.-M. JÉZÉQUEL, B. MEYER. *Design by Contract: The Lessons of Ariane*, in "Computer", January 1997, vol. 30, n^o 1, p. 129–130.

[77] B. MEYER. *Reusability: The Case for Object-Oriented Design*, in "IEEE SOFTWARE", March 1987, n^o 3, p. 50–64.

[78] B. MEYER. *Applying "Design by Contract"*, in "IEEE Computer (Special Issue on Inheritance & Classification)", October 1992, vol. 25, n^o 10, p. 40–52.

[79] N. MOHA, Y.-G. GUÉHÉNEUC, A.-F. LE MEUR, L. DUCHIEN, A. TIBERGHIEN. *From a Domain Analysis to the Specification and Detection of Code and Design Smells*, in "Formal Aspects of Computing", 2009, http://hal.inria.fr/inria-00512135/en.

[80] G. PERROUIN, J. KLEIN, N. GUELFI, J.-M. JÉZÉQUEL. *Reconciling Automation and Flexibility in Product Derivation*, in "12th International Software Product Line Conference (SPLC 2008)", Limerick, Ireland, IEEE Computer Society, September 2008, p. 339–348, http://www.irisa.fr/triskell/publis/2008/Perrouin08a.pdf.

[81] A. RASHID, J. ARAÚJO. *Modularisation and composition of aspectual requirements*, in "Proceedings of the 2nd international conference on Aspect-oriented software development", 2003, p. 11–20.

[82] C. SZYPERSKI. *Component Software: Beyond Object-Oriented Programming*, ACM Press and Addison-Wesley, New York, N.Y., 1998.

[83] J. WARMER, A. KLEPPE. *The Object Constraint Language*, Addison-Wesley, 1998.