IN PARTNERSHIP WITH:
Université Nantes

Ecole des Mines de Nantes

Activity Report 2014

Project-Team ATLANMOD

Modeling Technologies for Software Production, Operation, and Evolution

IN COLLABORATION WITH: Laboratoire d'Informatique de Nantes Atlantique (LINA)

RESEARCH CENTER
Rennes - Bretagne-Atlantique

THEME
Distributed Systems and middleware

Table of contents

Project-Team ATLANMOD

Keywords: Model-driven Engineering, Software Engineering, Reverse Engineering, Formal Methods

Creation of the Team: 2008 July 01*, updated into Project-Team:* 2013 January 01.

1. Members

Faculty Members

Jordi Cabot [Team leader, EMN, Associate Professor, HdR]
Salvador Martinez Perez [EMN, Associate Professor]
Gerson Sunye [Univ. Nantes, Associate Professor]
Massimo Tisi [EMN, Associate Professor]

Engineers

Matthieu Allon [EMN, until May 2014]
Camilo Alvarez Duran [EMN, until Feb 2014]
Robin Boncorps [EMN, from Nov 2014]
Hugo Bruneliere [EMN]
Javier Canovas Izquierdo [Inria, granted by FP7 ARTIST project]
Hassene Choura [Inria, until Jun 2014, granted by OSEO Innovation]
Valerio Cosentino [Inria, granted by Région des Pays de la Loire]
Duc-Hanh Dang [Inria, until Feb 2014, granted by Région des Pays de la Loire]
Hamza Ed-Douibi [Armines, from Apr 2014]
Jokin Garcia Perez [ARMINES, from Oct 2014]
Zied Saidi [Inria, from Sep 2014]
Eric Umuhoza [EMN, from Apr 2014]
Juan David Villa Calle [Inria, until Sep 2014, granted by OSEO Innovation]

PhD Students

Amine Benelallam [ARMINES]
Gwendal Daniel [Inria, from Mar 2014]
Carlos Alberto Gonzalez Perez [EMN]

Post-Doctoral Fellow

Abel Gomez Llana [Inria, granted by R. des Pays de la Loire]

Visiting Scientist

Javier Criado [EMN, from May 2014 until Jul 2014]

Administrative Assistants

Anne Claire Binetruy [Inria]
Cecile Derouet [Inria, until Sep 2014]

Others

Maria Rolandi [Inria, from May 2014 until Oct 2014]
Michel Albonico [EMN, from Apr 2014]

2. Overall Objectives

2.1. Presentation

Model Driven Engineering (MDE) is a software engineering paradigm that advocates for the rigorous use of (software) models and model manipulation operations (known as model transformations) as the main artifacts in all software engineering activities. This comes from an industrial need to have a regular and homogeneous organization where different facets of a software system may be easily separated or combined when appropriate. The basic assumption of MDE is that models provide a better abstraction level than the classical programming code to manage the complexity of software development (and, in general, any other software-related task). When needed, executable code can be semi-automatically generated from (low-level) models of the system.

AtlanMod focuses on developing pioneering solutions to solve core research challenges in MDE and to ensure its successful application on relevant industrial problems. In particular, AtlanMod is focusing on three key challenges: evaluating the correctness of models and model transformations, ensuring the scalability of modeling techniques to deal with very large models and developing software comprehension and modernization methods by means of the application of model-driven reverse engineering techniques on running software.

2.2. Previous Achievements

AtlanMod has significantly contributed to the evolution of MDE and to the progressive emergence of a scientific community in this field. The team developed a complete modeling framework [41] [50] providing core MDE components for (meta)model definition and manipulation.

The iterative definition of this conceptual framework has been validated by the construction of an MDE toolbox based on the conclusion that MDE is in fact a branch of language engineering. Models can be formally defined by means of metamodels, considered as the definition of the abstract syntax of a language in the same way grammars are used to define programming languages. Modeling languages are then the combination of a metamodel (abstract syntax), a notation (concrete syntax) plus a definition of the language semantics. In this sense, our toolbox can be regarded as a kind of language workbench offering the building blocks to define and manipulate models and metamodels. All these mutually interrelated tools are available under *Eclipse.org* (projects or components: M2M, ATL, AM3, AMW). They are currently in use in research, teaching, and industry and they have a broad user community.

Beyond the development of core MDE components, AtlanMod has also expressed a strong interest in the application and adaptation of these building blocks for specially relevant and challenging industrial problems. As an example, in this context, AtlanMod has been leading the MoDisco project [1] to build reverse engineering solutions for legacy systems.

3. Research Program

3.1. MDE Foundations

Traditionally, models were often used as initial design sketches mainly aimed for communicating ideas among developers. On the contrary, MDE promotes models as the primary artifacts that drive all software engineering activities (i.e. not only software development but also evolution, reverse engineering, interoperability and so on) and are considered as the unifying concept [41]. Therefore, rigorous techniques for model definition and manipulation are the basis of any MDE framework.

[1] http://eclipse.org/MoDisco/

The MDE community distinguishes three levels of models: (terminal) model, metamodel, and metametamodel. A terminal model is a (partial) representation of a system/domain that captures some of its characteristics (different models can provide different knowledge views on the domain and be combined later on to provide a global view). In MDE we are interested in terminal models expressed in precise modeling languages. The abstract syntax of a language, when expressed itself as a model, is called a metamodel. A complete language definition is given by an abstract syntax (a metamodel), one or more concrete syntaxes (the graphical or textual syntaxes that designers use to express models in that language) plus one or more definition of its semantics. The relation between a model expressed in a language and the metamodel of that language is called *conformsTo*. Metamodels are in turn expressed in a modeling language called metamodeling language. Similar to the model/metamodel relationship, the abstract syntax of a metamodeling language is called a metametamodel and metamodels defined using a given metamodeling language must conform to its metametamodel. Terminal models, metamodels, and metametamodel form a three-level architecture with levels respectively named M1, M2, and M3. A formal definition of these concepts is provided in [49] and [42]. MDE promotes *unification by models*, like object technology proposed in the eighties *unification by objects* [39]. These MDE principles may be implemented in several standards. For example, OMG proposes a standard metametamodel called Meta Object Facility (MOF) while the most popular example of metamodel in the context of OMG standards is the UML metamodel.

In our view the main way to automate MDE is by providing model manipulation facilities in the form of model transformation operations that taking one or more models as input generate one or more models as output (where input and output models are not necessarily conforming to the same metamodel). More specifically, a model transformation Mt defines the production of a model Mb from a model Ma. When the source and target metamodels (MMs) are identical ($MMa = MMb$), we say that the transformation is endogenous. When this is not the case ($MMa \neq MMb$) we say the transformation is exogenous. An example of an endogenous transformation is a UML refactoring that transforms public class attributes into private attributes while adding accessor methods for each transformed attribute. Many other operations may be considered as transformations as well. For example verifications or measurements on a model can be expressed as transformations [44]. One can see then why large libraries of reusable modeling artifacts (mainly metamodels and transformations) will be needed.

Another important idea is the fact that a model transformation is itself a model [40]. This means that the transformation program Mt can be expressed as a model and as such conforms to a metamodel MMt. This allows an homogeneous treatment of all kinds of terminal models, including transformations. Mt can be manipulated using the same existing MDE techniques already developed for other kinds of models. For instance, it is possible to apply a model transformation Mt' to manipulate Mt models. In that case, we say that Mt' is a higher order transformation (HOT), i.e. a transformation taking other transformations (expressed as transformation models) as input or/and producing other transformations as output.

As MDE developed, it became apparent that this was a branch of language engineering [43]. In particular, MDE offers an improved way to develop DSLs (Domain-Specific Languages). DSLs are programming or modeling languages that are tailored to solve specific kinds of problems in contrast with General Purpose Languages (GPLs) that aim to handle any kind of problem. Java is an example of a programming GPL and UML an example of a modeling GPL. DSLs are already widely used for certain kinds of programming; probably the best-known example is SQL, a language specifically designed for the manipulation of relational data in databases. The main benefit of DSLs is that they allow everybody to write programs/models using the concepts that actually make sense to their domain or to the problem they are trying to solve (for instance Matlab has matrices and lets the user express operations on them, Excel has cells, relations between cells, and formulas and allows the expression of simple computations in a visual declarative style, etc.). As well as making domain code programmers more productive, DSLs also tend to offer greater optimization opportunities. Programs written with these DSLs may be independent of the specific hardware they will eventually run on. Similar benefits are obtained when using modeling DSLs. In MDE, new DSLs can be easily specified by using the metamodel concept to define their abstract syntax. Models specified with those DSLs can then be manipulated by means of model transformations (with ATL for example [48]).

When following the previously described principles, one may take advantage of the uniformity of the MDE organization. As an example, considering similarly models of the static architecture and models of the dynamic behavior of a system allows at the same time economy of concepts and economy of implementation.

The following sections describe the main MDE research challenges the team is addressing. They go beyond the development of core MDE techniques (topic on which the team, as mentioned above, has largely contributed in the past, and that we believe is quite well-covered already) and focus on new aspects that are critical for the successful application of MDE in industrial contexts.

3.2. Reverse Engineering

One important domain that is being investigated by the AtlanMod team is the reverse engineering of existing IT systems. We do believe that efficiently dealing with such legacy systems is one of the main challenges in Software Engineering and related industry today. Having a better understanding of these systems in order to document, maintain, improve or migrate them is thus a key requirement for both academic and industrial actors in this area. However, it is not an easy task and it still raises interesting challenging issues to be explored [46].

We have shown how reverse engineering practices may be advantageously revisited with the help of the MDE approach and techniques, applying (as base principle) the systematic representation as models of the required information discovered from the legacy software artifacts (e.g. source code, configuration files, documentation, metadata, etc). The rise in abstraction allowed by MDE can bring new hopes that reverse engineering is now able to move beyond more traditional ad-hoc practices. For instance, a industrial PhD in partnership with IBM France aimed to investigate the possibilities of conceptualizing a generic framework enabling the extraction of business rules from a legacy application, as much as possible, independently of the language used to code it. Moreover, different pragmatic solutions for improving the overall scalability when dealing with large-scale legacy systems (handling huge data volumes) are intensively studied by the team.

In this context, AtlandMod has set up within the past years and is still developing the open source Eclipse MoDisco project (see 5.2). MoDisco is notably being referenced by the OMG ADM (Architecture Driven Modernization) normalization task force as the reference implementation for several of its standard metamodels. It is also used practically and improved in various collaborative projects the team is currently involved in (e.g. FP7 ARTIST). Complementary to the work based on MoDisco, we have also been experimenting (still in an industrial context, cf. TEAP FUI project) on the related problem of data federation from heterogeneous sources in the domain of Enterprise Architecture. This has notably resulted in a prototype called EMF Views that can be practically used in such reverse engineering scenarios.

Reverse engineering techniques have also been used in the context of the Web. In the last years the development of Web APIs has become a discipline that companies have to master to succeed in the Web. The so-called API economy requires, on the one hand, companies to provide access to their data by means of Web APIs and, on the other hand, web developers to study and integrate such APIs into their applications. The exchange of data with these APIs is usually performed by using JSON, a schemaless data format easy for computers to parse and use. While JSON data is easy to read, its structure is implicit, thus entailing serious problems when integrating APIs coming from different vendors. Web developers have therefore to understand the domain behind each API and study how they can be composed. We tackle this problem by developing a MDE-based process able to reverse engineer the domain of Web APIs and to identify composition links among them. The approach therefore allows developers to easily visualize what is behind the API and the connections points that may be used in their applications.

We have recently opened a new research line in the context software analysis, in particular, in the Open-Source Software (OSS) field. The development of OSS follows a collaborative model where any developer can contribute to the advance of the project. To enable this collaboration, OSS projects use a plethora of tools such as forums, issue-trackers and Q&A websites, that developers can adopt to coordinate each other in the development process. Such a collaboration environment includes adapted solutions and provides effective communication means, but also causes scattering of the collaboration data, which hamper the understanding of the whole development process (e.g., who is leading the development or making the decisions). In this context, we propose to use reverse engineering techniques to better understand how OSS projects are developed in a broad sense, thus taking into account the different collaboration tools used and how they influence in the development of OSS projects.

3.3. Security Engineering

Several components are required to build up a system security architecture, such as firewalls, database user access control, intrusion detection systems, and VPN (Virtual Private Network) routers. These components must be properly configured to provide an appropriate degree of security to the system. The configuration process is highly complex and error-prone. In most organizations, security components are either manually configured based on security administrators expertise and flair; or simply recycled from existing configurations already deployed in other systems (even if they may not be appropriated for the current one). These practices put at risk the security of the whole organization.

As a first step we intend to apply model-driven techniques for the extraction of high level model representations of security policies enforced by system components like networks of firewalls, RDBMS and CMSs. Firewalls, core components in network security systems, are generally configured by using very low level vendor specific rule-based languages, difficult to understand and to maintain. As a consequence, as the configuration files grow, understanding which security policy is being actually enforced or checking if inconsistencies has been introduced becomes a very complex and time consuming task. Similarly, in RDBMSs and CMSs policies are configured and stored by using different, often low-level, mechanisms.

We propose to raise the level of abstraction so that the user can deal directly with the high level policies. Once a model representation of the enforced policy is available, model-driven techniques will ease some of the tasks we need to perform, like consistency checking, validation, querying and visualization. Easy migration between different vendors will be also enabled.

As a further step we intend to apply model-driven techniques for the integration of the diverse security policies extracted from concrete system components. In the case of complex systems composed of a number of interacting heterogeneous subsystems, access-control is pervasive with respect to their architecture. As mentioned above, we can find access-control enforcement rules in different components placed at different architectural levels where rules in a component may impact the execution of the security rules of another component. In addition, the access-control techniques implemented in each component may follow different AC models in order to best suit the needs of the component. Thus, ideally, a global representation of the access-control policy of the whole system should be available, as analysing a component policy in isolation does not provide enough information. Unfortunately, most times this global policy is not explicit or is outdated. This step requires to unveil the implicit dependencies between the set of policies working in an encompassing system, so that a model representing the global AC policy can be built and the global analysis of the AC security is enabled

3.4. Software Quality

As with any type of production, an essential part of software production is determining the quality of the software. The level of quality associated to a software product is inevitably tied to properties such as how well it was developed and how useful it is to its users. AtlanMod team focus on researching techniques for the formal verification and testing of software models and model transformations.

These techniques must be applied at the model level (to evaluate the quality of specific software designs) and at the metamodel level (to evaluate the quality of modeling languages). In both cases, the Object Constraint Language (OCL) of the OMG is widely accepted as a standard textual language to complement (meta)model specifications with all those rules/constraints that cannot be easily defined using graphical modeling constructs.

Among all possible properties to verify, we take as the basic property the *satisfiability* property, from which many others may be derived (as liveliness, redundancy, subsumption,...). Satisfiability checks whether it is possible to create a valid instantiation (i.e. one that respects all modeling constraints) of a give (meta)model. Satisfiability is an undedicable problem when general OCL constraints are used as part of the model definition.

To deal with this problem, the team maintains the tool EMFtoCSP which translates the model verification challenge into the domain of constraint logic programming (CLP) for which sophisticated decision procedures exist. The tool integrates the described functionality in the Eclipse Modeling Framework (EMF) and the Eclipse Modeling Tools (MDT), making the functionality available for MDE in practice.

To complement these formal verification techniques we are also working on testing techniques, specially to optimize the testing of model transformations. White-box testing for model transformations is a technique that involves the extraction of knowledge embedded in the transformation code to generate test models. In our work, we apply static analysis techniques to model transformation specifications and represent the extracted knowledge as partial models that can drive the generation of highly effective test models (specially in terms of coverage).

3.5. Collaborative Development

Software development processes are collaborative in nature. The active participation of end-users in the early phases of the software development life-cycle is key when developing software. Among other benefits, the collaboration promotes a continual validation of the software to be build, thus guaranteeing that the final software will satisfy the users' needs. In this context, we have opened two novel research lines focused on the collaborative development *in* MDE and the collaborative development *with* MDE. The former is aimed at promoting the collaboration in the context of MDE while the latter uses MDE techniques to promote the participation in software development processes.

Collaboration is important in the context of MDE, in particular, when creating Domain-Specific Modeling Languages (DSMLs) which are (modeling) languages specifically designed to carry out the tasks of a particular domain. While end-users are actually the experts of the domain for which a DSML is developed, their participation in the DSML specification process is still rather limited nowadays (they are normally only involved in providing domain knowledge or testing the resulting language). This means that the MDE technical experts and not end-users are the ones in control of the DSML construction and evolution. This is a problem because errors in understanding the domain may hamper the development process and the quality of the resulting DSML. Thus, it would be beneficial to promote a more active participation of end-users in the DSML development process.

We have been working on the required support to make effective this participation, in particular, we have developed Collaboro, an approach which enables the involvement of the community (i.e., end-users and developers) in the DSML creation process. Collaboro allows modeling the collaborations between community members taking place during the definition of a new DSML and supports both the collaborative definition of the abstract (i.e., metamodel) and concrete (i.e., notation) syntaxes for DSMLs by providing specific constructs to enable the discussion. Thus, each community member will have the chance to request changes, propose solutions and give an opinion (and vote) about those from others. We believe this discussion will enrich the language definition significantly and ensure that the end result satisfies as much as possible the expectations of the end-users. Collaboro has also been extended to support the example-driven development of DSMLs, thus promoting the engagement of end-users in the process.

The lessons learnt from this MDE-focused collaboration research are now being applied to the more general context of software development. In particular, our interest is to study how software development processes are governed (i.e. how the collaboration among developers and user takes place). Any software development project has to cope with a huge number of tasks consisting of either implementing new issues or fixing bugs. Thus, effective and precise prioritization of these tasks is key for the success of the project. Governance rules enable the coordination of developers in order to advance the project. Despite their importance, in practice governance rules are hardly ever explicitly defined, specially in the context of Open Source Systems (OSS), where it is hard to find a explicit system-level design, a project plan, schedule or list of deliverables. To alleviate this situation, mechanisms to facilitate the communication and the assignment of work are considered crucial for the success of the development. Tracking and issue-tracking systems, mailing lists and forums are broadly used to manage the tasks to be performed. While these tools provide a convenient compartmentalization of work and effective means of communication, they fall short in providing adequate support for specifying and enforcing governance rules (e.g. supporting the voting of tasks, easy tracking of decisions made in the project, etc.).

Thus, we believe the explicit definition of governance rules along with the corresponding infrastructure to help developers follow them would have several benefits, including improvements in the transparency of the decision-making process, traceability (being able to track why a decision was made and who decided it) and the automation of the governance process (e.g. liberating developers from having to be aware and follow the rules manually, minimizing the risk of inconsistent behaviour in the evolution of the project). We resort on MDE techniques to tackle this problem and provide a DSL specially adapted to the domain of governance in software projects to let project managers easily define the governance rules of their projects.

3.6. Scalability

As MDE is increasingly applied to larger and more complex industrial applications, the current generation of modelling and model management technologies are being stressed to their limits in terms of their capacity to accommodate collaborative development, efficient management and persistence of models larger than a few hundreds of megabytes in size. Additional research and development is imperative in order to enable MDE to remain relevant with industrial practice and to continue delivering its widely recognised productivity, quality, and maintainability benefits. Achieving scalability in modelling and MDE involves being able to construct large models and domain-specific languages in a systematic manner, enabling teams of modellers to construct and refine large models in a collaborative manner, advancing the state-of-the-art in model querying and transformations tools so that they can cope with large models (of the scale of millions of model elements), and providing an infrastructure for efficient storage, indexing and retrieval of large models. AtlanMod wants to provide a solution for these aspects of scalability in MDE by extending the Eclipse modeling framework, to create an open-source solution to scalable modeling in industry.

3.7. Industrialization of open source tools

Research labs, as a source of innovation, are potential key actors of the Software Engineering market. However, an important collaborative effort with the other players in the software industry is still needed in order to actually transfer the corresponding techniques or technologies from the research lab to a company. Based on the AtlanMod concrete experience with the previously mentioned open source tools/projects, we have extracted a pragmatic approach [4] for transforming the results of scientific experimentation into practical industrial solutions.

While dealing with innovation, this approach is also innovation-driven itself, as the action is actually conducted by the research lab via a technology transfer. Three different partners are directly involved in this process, using open source as the medium for maintaining a constant interaction between all of them:

- **Use Case Provider.** Usually a company big enough to have to face real complex industrial scenarios which need to be solved (at least partially) by applying new innovative principles and techniques;
- **Research Lab.** Usually a group from a research institute (public or private) or university evaluating the scientific relevance of the problems, identifying the research challenges and prototyping possible solutions;
- **Technology Provider.** Usually a small or medium company, with a particular technical expertise on the given domain or Software Engineering field, building and delivering the industrial version of the designed solutions;

From our past and current experience, three main characteristics of this industrialization *business model* can be highlighted:

- **Win-win situation.** Each partner can actually focus on its core activity while also directly benefiting from the results obtained by the others (notably the research lab can continue to do research);
- **Application-driven context.** The end-user need is at the origin of the process, which finally makes the developed solution actually relevant;
- **Iterative process.** The fact of having three distinct partners requires different regular and consecutive exchanges between all of them.

4. Application Domains

4.1. Application domains

By definition, MDE can be applied to any software domain. Core MDE techniques developed by the team have been successfully applied to a large variety of industrial domains from information systems to embedded systems. MDE is not even restricted to software engineering, but also applies to data engineering [47] and to system engineering [38]. There are a lot of problems in these application domains that may be addressed by means of modeling and model transformation techniques.

As a result, AtlanMod has collaborated with a great variety of different companies ranging from the Automotive to the Insurances domains and from SMEs to large enterprises through the projects described later on in this same report. AtlanMod hopes to continue this trend in the future.

5. New Software and Platforms

5.1. The ATL Model Transformation Language

URL: http://www.eclipse.org/atl/

With an eye on the normative work of the OMG (MOF, OCL, QVT, etc.), a new conceptual framework has been developed based on a second generation model transformation language called ATL. Although ATL influenced the OMG standard, the approach is more general as discussed in [48]. In 2004 IBM gave an Eclipse innovation award to the ATL project. In 2007 Eclipse recognized ATL as one central solution for model transformation and promoted it to the M2M project (see *Eclipse.org/m2m*). There are more than 200 industrial and academic sites using ATL today, and several Ph.D. thesis in the world are based on this work.

In 2011 we started a new evolution phase for ATL. Our mid-term plan is making of ATL the leading solution for building autonomous reactive transformation systems, i.e. transformation networks that can autonomously manage a set of dataflows among the application models.

Following this line, we first implemented a new refinement mode for ATL, to support in-place transformations. This extension allows the dynamic manipulation of models while keeping them connected to runtime applications. Next, we presented a lazy execution algorithm for ATL. With it, the elements of the target model are generated only when and if they are accessed. This extension allows to build reactive transformation systems that react to requests of model elements, by triggering the necessary computation. Our lazy version of ATL enables also transformations that generate infinite target models, extending the application space of the model-transformation paradigm.

The latest (still ongoing) work in this direction is the development of a full reactive ATL engine, able to activate the minimal computation for responding to updates or request on the involved models. This engine is studied to scale up with large ATL networks. In this line we also introduced an algorithm for simplifying ATL transformation chains.

Performing just the required work on model transformation improves scalability, an open issue the previous described works contribute to solve.n efficient execution, as in the the lazy and reactive scenarios, may help with scalability problems by focusing the tasks in the required part of a very large transformation. However, this is not always the case and we might have to perform operations in the whole model. In this scenario, a solution for the scalability problem would be to take advantage of multi-core architectures that are very popular today, to improve computation times in the transformation of very large models. In this sense, a first step explores the strong parallelization properties rule-based languages like ATL have. A new prototype implementation of a parallel ATL engine has been developed showing how transformations can be developed without taking into account concurrency concerns, and such a transformation engine can automatically parallelize operations improving execution times.

Aligned with this research line we propose in recent works an approach to automatically parallelize the computation of model transformation using Cloud infrastructures. For this, we take advantage of a well-known distributed programming model: *MapReduce*. In this sense, we introduce an algorithm aligning both execution semantics of ATL and MapReduce. Based on this, a new prototype tool has been developed [2] showing in several experiments the scalability of the solution.

5.2. MoDisco (Model Discovery)

URL: http://www.eclipse.org/MoDisco/

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

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

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

The initiative continues to be developed within the context of the European FP7-ICT project named ARTIST [3], and also to a lower extent within the context of the French FUI 13 project named TEAP.

5.3. Community-driven language development

URL: http://atlanmod.github.io/collaboro

Software development processes are collaborative in nature. Neglecting the key role of end-users leads to software that does not satisfy their needs. This collaboration becomes specially important when creating Domain-Specific Languages (DSLs), which are (modeling) languages specifically designed to carry out the tasks of a particular domain. While end-users are actually the experts of the domain for which a DSL is developed, their participation in the DSL specification process is still rather limited nowadays.

[2] https://github.com/atlanmod/ATL_MR

[3] http://www.artist-project.eu/

Thus, Collaboro is an approach to make language development processes more participative, meaning that both developers and users of the language can collaborate together to design it and make it evolve. Since the very first implementation of the Collaboro toolset was released, it has evolved to provide support to both Eclipse-based and web-based clients.

The Eclipse-based client has been developed as a plugin in the platform while the web-based client includes two components: (1) the server-side part, which offers a set of services to access to the main functionalities of Collaboro; and the client-side part, which allows both end-users and developers to take part of the DSML development process from their browsers. The server-side component has been developed as a Java web application which uses a set of Servlets providing the required services. On the other hand, the client-side component has been developed as an AngularJS-enabled website and provides.

The Collaboro clients provide access to the following features:

- Version view to navigate through the Proposals of a version of a language. For each Proposal, the solutions and comments are shown.
- Collaboration view to show the data related to a Collaboration selected in the version view. This view also shows the changes to apply if the selected element is a Solution.
- The user can login to the Collaboro system and create proposals, solutions and comments by right-clicking in the version view. The user can also vote for/against the collaborations.
- Decision engine based on a total agreement (i.e., all the community users must vote for the collaboration). The decision engine can be launch by using the menu bar.
- Notation engine and Notation view to render SVG snapshots of the DSL concrete syntax.
- Support for example-driven development of DSMLs, thus incoporating a graphical editor which allows end-users to draw examples of the DSML they are developing.

5.4. JSON Discoverer

URL: http://atlanmod.github.io/json-discoverer/

Given a set of JSON documents, the tool (distributed as an open source Eclipse plugin contributed to MoDisco) returns a model describing their implicit schema. We follow an iterative process where new JSON documents (from the same or different services within the API) contribute to enrich the generated model. The model helps to both understand single services and to infer possible relationships between them, thus suggesting possible compositions and providing an overall view of the application domain. The tool has also been released as a web site, thus allowing any web developer to use our approach without the need of installing Eclipse.

5.5. EMF-REST

URL: http://emf-rest.com/

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

5.6. EMF Views (Model Views)

URL: https://github.com/atlanmod/emfviews

The Eclipse Modeling Framework (EMF) is widely used in the Eclipse community: defining domain models and generating corresponding source code, modeling software architectures, specifying DSL concepts or simply representing software/user data in different contexts. This implies that any software project involves a large number of heterogeneous but interrelated EMF models. To make matters worse, not all participants in the project should have the same kind of access/views on the models. Some users only need to see some parts of one model, others have to get the full model extended with data from another model, or simply access to a combination of information coming from different interconnected models. Up to now, creating such perspectives transparently in EMF was almost impossible. Based on the unquestionable success/usefulness of database views to solve similar problems in databases, EMF Views aims to bring the same concept to the modeling world. Thanks to the three main constructs (inspired from SQL) offered by the tool, designers can create new model views: SELECTing a subset of elements from a model, PROJECTing only some of the properties of those elements and/or JOINing them with elements from other models. A model view is a special type of model whose instances are directly computed at runtime based on the model view definition and concerned actual model(s). EMF Views has been initially developed in the context of the TEAP industrial project http://www.teap-project.org/ that ended in November 2014, by showing different possible applications of model views including:

- Software architect/developer views relating UML design models and Java code models (cf. Eclipse MoDisco).
- Enterprise architect views linking (BPMN) business process models, (ReqIF) requirements models and (TOGAF) architecture models.
- View transformation using dedicated technologies (e.g. Eclipse ATL).
- Report generation from views, etc.

The EMF Views prototype is currently being re-used and further developed, in a (meta)model extension context this time, within the ongoing MoNoGe industrial project. The objective of this present work is to propose a simple base generic (meta)model extension mechanism, relying on EMF Views capabilities, that could be deployed in different scenarios where (meta)model extension is required (e.g. metamodel evolution, model integration, etc.)

A presentation of EMF Views tooks place at EclipseCon 2014 [4], held in San Francisco, California, U.S.A

5.7. EMFtoCSP

URL: http://code.google.com/a/eclipselabs.org/p/emftocsp/

EMFtoCSP is a tool for the verification of precisely defined conceptual models and metamodels. For these models, the definition of the general model structure (using UML or EMF) is supplemented by OCL constraints. The Eclipse Modeling Development Tools (MDT [5]) provides mature tool support for such OCL-annotated models with respect to model definition, transformation, and validation.

However, an additional important task that is not supported by Eclipse MDT is the assurance of model quality. A systematical assessment of the correctness of such models is a key issue to ensure the quality of the final application. EMFtoCSP fills this gap by provided support for automated model verification in Eclipse.

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

1. Satisfiability – is the model able to express our domain? For this check, the minimal number of instances and links can be specified to ensure non-trivial instances.
2. Unsatisfiability – is the model unable to express undesirable states? To verify this, we add further constraints to the model that state undesired conditions. Then we can check if is it impossible to instantiate the amended model.

[4] https://www.eclipsecon.org/na2014/session/modeling-symposium
[5] http://www.eclipse.org/modeling/mdt/?project=ocl

3. Constraint subsumption – is one constraint already implied by others (and could therefore be removed)?
4. Constraint redundancy – do different constraints express the same fact (and could therefore be removed)?

To solve these search problems, EMFtoCSP translates the EMF/OCL (resp. UML/OCL) model into a constraint satisfaction problem and employs the Eclipse CLP solver [6]to solve it. This way, constraint propagation is exploited to tackle the (generally NP-hard) search.

The tool is a continuation of the UMLtoCSP approach [45] developed previously by Jordi Cabot, Robert Clarisó and Daniel Riera. It provides a generic plugin framework for Eclipse to solve OCL-annotated models using constraint logic programming. Apart from already supported Ecore and UML metamodels, further metamodels can be added easily in the future. Similarly, other constraint solving back-ends can be integrated. It is provided under the Eclipse Public License.

5.8. NeoEMF

URL: http://www.neo4emf.com

NeoEMF (a relaunch of the tool formerly known as Neo4EMF) is an open source software distributed under the terms of the Eclipse Public License that provides a backend-agnostic persistence solution for big, complex and highly interconnected EMF models. NeoEMF is a model repository and persistence framework allowing on-demand loading, storage, and unloading of large-scale EMF models.

NeoEMF is designed to allow the easy integration of custom backends depending on user needs. By default, NeoEMF is bundled with out-of-the-box support for graph databases (based on the blueprints API [7] and key-value stores (based on MapDB [8]). Blueprints is an abstraction layer for graph storages that allows changing the actual database used without affecting the application code. The blueprints-based back-end allows the integration of NeoEMF and Neo4j—among other databases—providing in NeoEMF the full set of features already implemented in Neo4EMF. MapDB is an efficient key-value store that provides concurrent Maps, Sets and Queues backed by disk storage or off-heap memory.

In terms of performance, NeoEMF eases data access and storage not only in a manner to reduce time and memory usage but also to allow big models to fit into small memory. This is provided through the following features:

- Lazy-loading mechanism. Model objects are loaded on demand while needed. In its basic configuration, model objects act as a proxy that ocuppy little memory, and fields are only retrieved when accessed.
- Caching. NeoEMF relies on database caches to retrieve EObjects, but in some situation this is not enough. For this reason, the architecture on NeoEMF allows the easy implementation of domain-specific cache strategies based on the decorator pattern.
- Auto-commit. In back-ends in which transaction data is stored on the heap, it is possible to use the auto-commit feature to split large transaction into several small ones.
- Dirty saving. The dirty saving feature is an step forward on the auto-commit strategy. It allows to safely handle big transactions by spliting them into small ones by saving partial changes made on models to disk. In case of transaction failure or cancellation, the partial model changes can be reverted and the model is restored to its original state.

A session about NeoEMF took place at eclipseCon France 2014 [9], held in Toulouse, France.

[6] http://eclipseclp.org/

[7] https://github.com/tinkerpop/blueprints/

[8] https://github.com/jankotek/MapDB

[9] https://www.eclipsecon.org/france2014/session/neo4emf-when-big-models-are-no-longer-issue

Works are still going over NeoEMF (within the context of the project ITM Factory - FUI14) to provide more utilities such as backend-aware query languages (which allows improving performance by taking advantage of the backend built-in query languages), concurrent access, model distribution, and other Ecore utilities.

5.9. GitHub Label Analyzer

URL: http://atlanmod.github.io/gila/

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

We believe that visualization techniques could be applied here to overcome this challenge. In particular, we have created GiLA, a tool to better understand how labels are being used in GitHub projects, with the aim of providing more insights into how such projects are being managed. GiLA provides three visualizations addressing three different viewpoints, specifically:

V1 Label usage, which helps to identify the most used labels and which ones are commonly used together.

V2 User involvement, which allows discovering the most active and knowledgeable users around each label.

V3 Typical Label timeline, which provides some insights about how issues under that label evolve over time (e.g., time to be treated).

The tool can be used to explore these viewpoints on all the original projects (i.e., projects that are not a fork of a previous project) in GitHub. We believe that the results favour not only a better comprehension of the project but also help in its advancement, e.g., by helping to quickly identify experts on a particular topic/label.

6. New Results

6.1. Model Quality

Our work aims to enhance the quality of the modeling activity in the context of software engineering and language engineering. This year, this has translated in the following results:

- A systematic review [16] of all formal verification approaches targeting the quality evaluation of software models to be used as the basis for future research on the topic and as a kind of reference comparison to compare new tools with existing ones.
- A complete description of our CSP-based approach for the verification of UML/OCL models (where both the uml constructs and OCL expressions are translated into a constraint satisfaction problem) [12]
- A new test data generation approach for Model Transformations that combines partitions and constraint analysis to try to mazimize the coverage of the generates tests [29]

6.2. Model Driven approach to mobile applications development

Cross-platform and multi-device design, implementation and deployment is a barrier for today's IT solution providers, especially SME providers, due to the high cost and technical complexity of targeting development to a wide spectrum of devices, which differ in format, interaction paradigm, and software architecture. Our work aims at exploiting the modern paradigm of Model-Driven Engineering and code generation to simplify multi-device development, reducing cost and development times, so as to increase the profit of SME solution providers and at the same time reduce the price and total cost of ownership for end-customers. In [22] we defined a Platform Independent Modeling language for mobile applications. The language has been defined as a mobile extension of an OMG standard called Interaction Flow Modeling Language (IFML). The research included also the development of an Eclipse-based modeling tool for mobile apps and the first prototypes of automatic code generators.

6.3. Security

Most companies information systems are composed by heterogeneous components responsible of hosting, creating or manipulating critical information for the day-to-day operation of the company. Securing this information is therefore one of their main concerns, more particularly specifying Access Control (AC) policies. However, the task of implementing an AC security policy (sometimes relying on several mechanisms) remains complex and error prone as it requires knowing low level and vendor-specific facilities. In this context, discovering and understanding which security policies are actually being enforced by the Information System (IS) becomes critical. Thus, the main challenge consists in bridging the gap between the vendor-dependent security features and a higher-level representation. This representation has to express the policies by abstracting from the specificities of the system components, allowing security experts to better understand the policy and to implement all related evolution, refactoring and manipulation operations in a reusable way.

In 2014, we have presented a Ph.D. thesis tackling the aforementioned problems. It proposes a model-driven automatic reverse engineering mechanism capable of analyzing deployed security aspects of components (e.g. concrete firewall configurations) to derive the abstract model (e.g. network security global policy) that is actually enforced. Once the model is obtained, it can be reconciled with the expected security directives, to check its compliance, can be queried to test consistency or used in a process of forward engineering to generate validated security configurations. This work also provides the first steps towards the integration of the diverse security policies extracted from the subsystems composing a complex Information System in a global security representation.

6.4. Model-Driven Document Engineering

As a result of a long-term collaboration of one of the AtlanMod team members with the ISSI research group at the Universitat Politècnica de València, we have participated in the publication of several works on the area of the Document Engineering. In this research line, we have applied the MDE methods and tools to the product-line-based generation of customized documents resulting in the so-called DPL methodology [10]. The Document Product Lines (DPL) approach, which we throughly describe in a journal publication [17], provides a framework for variable content document generation that follows an alternate path to the traditional variable document generation. DPL has been created with a twofold goal: first, to make creating variable content documents available to non-experts by including a domain engineering process previous to the document generation itself; and secondly, to enforce content reuse at domain level.

DPLFW is the main tool supporting the DPL methodology, and in the demonstrations track of the MODELS conference we showed all its capabilities. In addition to these contributions, we have published several works demonstrating the applicability of the DPL–DPLFW tandem in different domains, such as the development of executable emergency plans in crisis managent contexts [25], the development of learning objects in the e-learning field [32] and the generation of customized documents in e-Government solutions [33].

[10] http://dpl.dsic.upv.es, only in Spanish

6.5. Reverse Engineering and Evolution

Model Driven Reverse Engineering (MDRE), and its applications such as software modernization, is a discipline in which model-driven development (MDD) techniques are used to treat legacy systems. During this year, Atlanmod has continued working actively on this research area. The main contributions are the following:

- In the context of the ARTIST FP7 project, the work has been continued on reusing (and extending accordingly) MoDisco and several of its components to provide the Reverse Engineering support required within the project. At conceptual-level, the MoDisco Model Discovery + Model Understanding overall two-step approach [11] has been published and promoted as an important part of the ARTIST migration methodology and process [18]. At tooling-level, several (MoDisco-based) model discovery components from Java and SQL have been developed and made available as part of the official ARTIST OS Release [11]. Directly related to some of these components, a promising work has been initiated on studying deeper the discovery of behavioral aspects of software and dealing with their further understanding based on the OMG FUML standard combined with different modeling techniques (transformation, slicing, etc.). Complementary work has also been performed in the context of the TEAP FUI project finishing by the end of this year. It concerns the related problem of data federation from heterogeneous sources in the domain of Enterprise Architecture. This has notably resulted in a prototype called EMF Views that can be practically used in such reverse engineering scenarios [36] and also in other cases to be further explored (cf. the MoNoGe FUI project dealing with (meta)model extension).
- In a web context, in a previous work we shown how to discover the schema which is implicit in JSON data. This year we built on that contribution to study how schemas coming from different JSON-based web APIs can be composed [24]. Thus, we presented an approach able to identify composition links between schemas of different APIs. This composition information plus the API schemas are used to render a graph where paths represent API compositions and are used to easily identify how to compose the APIs. For instance, we illustrated one application based on generating sequence diagrams from graph paths, where the diagram includes the API calls (and their corresponding parameters) that web developers have to perform in order to compose one or more APIs.
- In the context of our work around DSLs, we have been working on facilitating the definition of DSLs from existing APIs. Sometimes library developers prefer to provide their users with a DSL, instead of (or in addition to) an API. APIs and DSLs can be seen as alternative methods to access the library functionalities, and are characterized by specific advantages. We therefore proposed a method to automatically analyze an existing object-oriented API and generate a DSL out of it. Our approach leverages on model-driven techniques to analyze and represent APIs at high-level of abstraction (i.e., as metamodels) which are later used to automatically generate the DSL components and the corresponding tooling, including parser, compiler and development environment. Developers can influence the DSL generation by editing the model-based API representation and by specifying design choices about the structure of the DSL to generate. A proof-of-concept implementation of the method has been developed, called *DSLit*, that is able to analyze Java APIs and generate textual DSLs.
- On the evolution side, we have been working on an approach to automatically resynchronize code-generation artefacts (in particular, model-to-text transformations) after changes on the target platform [28]

6.6. Scalability

The increasing number of companies embracing MDE methods and tools have exceeded the limits of the current model-based technologies, presenting scalability issues while facing the growing complexity of their data. Since further research and development is imperative in order to maintain MDE techniques as relevant as

[11] http://www.artist-project.eu/tools-of-toolbox/193

they are in less complex contexts, we have focused our research in three axes, (i) scalable persistence solutions, (ii) scalable model transformation engines, and (iii) testing of large scale distributed systems.

In [21], we lead the first open-set benchmark gathered from real-world cases to stress scalability issues in model transformation and query engines. This benchmark suite has been made public with a twofold goal: (i) to provide a reference benchmark suite to both the industry and the research community that can be used to compare and evaluate different technologies that may fulfill their needs; and (ii) to motivate the MDE community to be part of its development by allowing them to extend and contribute with additional cases not covered by the initial set.

On the other hand, we introduce Neo4EMF [20], a NoSQL database persistence framework based on Neo4j [12]. Neo4EMF provides light-weight on-demand loading and storage facilities for handling very large models. Additionally, we also show that Neo4EMF can handle the creation of very-large models without performing periodical saves manually.

In this paper [31], we argue that fUML may be leveraged to address the well-known interoperability issue between tools from different modeling platforms. This is done by providing a common execution language and by abstracting modeling frameworks into generic actions that perform elementary operations on models. User models can not only benefit from a unified execution semantics, but also modeling tools can benefit too. As a proof of concept, we show [37] how it can be applied to model transformation engines, in particular ATL. To this end, an prototype compiler from ATL to fUML has been built.

In [19], we present a model-based approach to define a dynamic oracle for checking global properties on distributed software. Our objective is to abstract relevant aspects of such systems into models by gathering data from different nodes and building a global view of the system, where properties are validated. These models are updated at runtime, by monitoring the corresponding distributed system. This process requires a distributed test architecture and tools for representing and validating global properties. To evaluate the ability of our approach, a real-scale experimental validation has been conducted.

7. Partnerships and Cooperations

7.1. Regional Initiatives

Program: **Pole Images et Reseaux - Appel Projets PME 2011**

Project title: StreamMaster

Duration: 2012 - 2014

Coordinator: Data Syscom

Other partners: Research and University (University of Nantes, Ecole de Design Nantes Atlantique, ESC Rennes) and Vendors and service providers (IMINFO)

Abstract: The purpose of the StreamMaster project is creating a universal software solution for the smart management of document streams, providing an added value over all the chain. StreamMaster will provide: an hybrid (local and remote) technological platform to allow user access, the possibility of connection to every information system and every input and output stream, the management of all the parameters of the document stream (cost, speed, delay, quality, environmental impact), security and reinforced document authentication mechanisms, non-falsifiable documents by means of invisible document tatooing, an innovative and multimodal HMI.

Program: **Pays de la Loire regional funding. Call: Creation of new teams**

Project title: AtlanMod New Team Creation

Duration: 2011 - 2014

Coordinator: AtlanMod

[12] http://www.neo4j.org

Other partners: None

Abstract: AtlanMod has been funded by the Pays de la Loire Regional Council new research teams program. This funding will mainly cover a PhD Student and two years of a postdoc to work on the quality of models research line.

7.2. National Initiatives

7.2.1. FUI

Program: FUI - AAP 15

Project acronym: MoNoGe

Project title: Atelier de Modélisation de Nouvelle Génération

Duration: 2013 - 2016

Coordinator: Softeam

Other partners: Industry (DCNS), Research and University (ARMINES AtlanMod, LIP6) and Vendors and service providers (Softeam, Soft-Maint, Mia-Software)

Abstract: There is currently in companies a wide diversity of models and modeling tools according to the application domains, services or contexts which are concerned. This implies different problems forbidding their plain exploitation: traceability, global coherence, continuity between works, knowledge management, etc. All are largely penalized by this situation that harms the mastering of the complexity of the related systems and software. The MoNoGe project has for objective to bring innovative solutions allowing to ensure the agility of the models and modeling tools. The term agility is here referring to the properties of interoperability, extensibility and evolution of models. The dynamic extension mechanism to be developed in MoNoGe, potentially inspiring from the OMG MEF standard currently under definition, is intended to preserve the original metamodel which can be conserved, partially hidden or extended. Thus, the legacy data and models can stay operational with the extended metamodel. The user does not have to deal with heavy migration or conversion operations, and can this way focus on its modeling activities while continuously exploiting past models. Our focus within the project is on defining conceptually such a (meta)model extension solution and proposing an implementing prototype based on Eclipse/EMF. To this intent, we are already studying the potential reuse (and improvement) of our EMF Views prototype in this given context.

Program: FUI - AAP 13

Project acronym: TEAP

Project title: TOGAF Entreprise Architecture Platform

Duration: 2012 - 2014

Coordinator: Obeo

Other partners: Industry (DCNS), Research and University (Inria AtlanMod) and Vendors and service providers (Obeo, Capgemini)

Abstract: The fast evolution of technologies (SOA, Cloud, mobile environments), the systems complexity and the growing need for agility require to be able to represent information systems as a whole. The high-level approach promoted by Enterprise Architecture (EA) is a key element in this context and intends to address all the systems dimensions: software components, associated physical resources, relationships with the companies requirements and business processes, implied actors/roles/structures, etc. The objective of the TEAP project was to specify and implement an EA platform based on the Open Group international standard named TOGAF and on the SmartEA technical solution. In addition to its base modeling capabilities, this platform now allows data federation from different existing sources (e.g. for reverse engineering purposes such as retro-cartography) as well as the definition of possible transformation chains (for governance and modernization). As part of this project, we have been notably using in practice (and improving accordingly) some of our works and corresponding prototypes such as EMF Views, ATL or some MoDisco components.

Program: FUI - AAP 13

Project acronym: ITM Factory

Project title: Information Technology Modernisation Factory

Duration: 04/2012 - 10/2014

Coordinator: Soft-Maint (Groupe SODIFRANCE)

Other partners: Mia-Software (Groupe SODIFRANCE), ACAPNOS, MMA and Inria AtlanMod.

Abstract: Application maintenance represents about 80 per cent of the computer market (at the French and global level). The challenge of software maintenance is to keep running applications with technologies that are no longer required to be maintained and with changing development teams and whose skills are not always validated on ancient languages. The main goal of the ITM Factory is to propose a software modernization framework, based on the ModDisco project and including: (i) an integrated workbench for software modernization engineers and (ii) a set of ready to use modernization cartirdges, i.e., a solution brick that meets a business challenge level, as opposed to a technical bricks that provides technical solutions that are integrated into a business solution.

7.3. European Initiatives

7.3.1. FP7 & H2020 Projects

7.3.1.1. ARTIST

Type: COOPERATION

Defi: Cloud Computing, Internet of Services and Advanced Software engineering

Instrument: Integrated Project

Duration: October 2012 - September 2015

Coordinator: Clara Pezuela (ATOS Spain)

Partner: ATOS and TECNALIA (Spain), Inria AtlanMod (France), Fraunhofer (Germany), TU Wien and Sparks (Austria), ENGINEERING (Italy), Spikes (Belgium), ATC and ICCS (Greece)

Inria contact: Hugo Bruneliere

Abstract: Nowadays Cloud Computing is considered as the ideal environment for engineering, hosting and provisioning applications. A continuously increasing set of cloud-based solutions is available to application owners and developers to tailor their applications exploiting the advanced features of this paradigm for elasticity, high availability and performance. Even though these offerings provide many benefits to new applications, they often incorporate constrains to the modernization and migration of legacy applications by obliging the use of specific development technologies and explicit architectural design approaches. The modernization and adaptation of legacy applications to cloud environments is a great challenge for all involved stakeholders, not only from the technical perspective, but also in business level with the need to adapt the business processes and models of the modernized application that will be offered from now on, as a service. The purpose of the ARTIST project is to propose and develop a novel model-driven approach for the migration of legacy applications in modern cloud environments which covers all aspects and phases of the migration process, as well as an integrated framework that supports all migration process.

7.3.1.2. MONDO

Title: Scalable Modelling and Model Management on the Cloud

Type: COOPERATION (ICT)

Defi: Cloud Computing, Internet of Services and Advanced Software engineering

Instrument: Small or medium-scale focused research project (STREP)

Duration: November 2013 - May 2016

Coordinator: The Open Group - X/Open Company

Partners: The Open Group - X/Open Company (United Kingdom), University of York (United Kingdom), Universidad Autonoma de Madrid (Spain), Budapest University of Technology and Economics (Hungary), IKERLAN (Spain), MIA Software (France), Cassidian (Germany)

Inria contact: Massimo Tisi

Abstract: As Model Driven Engineering (MDE) is increasingly applied to larger and more complex systems, the current generation of modelling and model management technologies are being pushed to their limits in terms of capacity and efficiency, and as such, additional research is imperative in order to enable MDE to remain relevant with industrial practice and continue delivering its widely recognised productivity, quality, and maintainability benefits. The aim of MONDO is to tackle the increasingly important challenge of scalability in MDE in a comprehensive manner. Achieving scalability in modelling and MDE involves being able to construct large models and domain specific languages in a systematic manner, enabling teams of modellers to construct and refine large models in a collaborative manner, advancing the state-of-the-art in model querying and transformations tools so that they can cope with large models (of the scale of millions of model elements), and providing an infrastructure for efficient storage, indexing and retrieval of large models. To address these challenges, MONDO brings together partners with a long track record in performing internationally-leading research on software modelling and MDE, and delivering research results in the form of robust, widely-used and sustainable open-source software, with industrial partners active in the fields of reverse engineering and systems integration, and a global consortium including more than 400 organisations from all sectors of IT.

7.3.1.3. Automobile

Title: Automated Mobile App Development

Type: Research For SMEs

Duration: November 2013 - October 2015

Coordinator: WebRatio s.r.l.

Partners: WebRatio, Politecnico di Milano (ITaly), AtlanMod-Armines, Moon Submarine (UK), ForwardSoftware (Rumania).

Inria contact: Jordi Cabot

Abstract:The AutoMobile project aims at designing and bringing to the market innovative methodologies, software tools, and vertical applications for the cost-effective implementation of cross-platform, multi-device mobile applications, i.e. business applications that can be accessed by users on a variety of devices and operating systems, including PC, cellular / smart phones and tablets.

Cross-platform and multi-device design, implementation and deployment is a barrier for today’s IT solution providers, especially SME providers, due to the high cost and technical complexity of targeting development to a wide spectrum of devices, which differ in format, interaction paradigm, and software architecture.

AutoMobile will exploit the modern paradigm of Model-Driven Engineering and code generation to dramatically simplify multi-device development, reducing substantially cost and development times, so as to increase the profit of SME solution providers and at the same time reduce the price and total cost of ownership for end-customers.

AutoMobile will rely on modeling languages such as IFML (Interaction Flow Modeling Languages) and on tools like WebRatio.

7.3.2. Collaborations in European Programs, except FP7 & H2020

Program: CORE Multi-annual thematic research programme. Fonds National de la Recherche Luxembourg.

Project acronym: TOOM

Project title: Testing Orders of Magnitude

Duration: September 2013 - August 2015

Coordinator: SnT/University of Luxembourg

Other partners: the iTrust company, EBRC, Inria Rennes/University of Nantes and the UFPR (Brazil).

Abstract: Over the last decade, large-scale systems drew much attention due to scalability and resiliency features. Many popular large-scale data-oriented systems (i.e., BigData), including, Peer-to-peer (P2P) and MapReduce, reached millions of users and processed petabytes of data, such as: Hadoop, Skype, Bittorrent, and Gnutella. The main reason is due to a decentralized manner to remove potential performance bottlenecks and centralized points of failure. Recently, cloud computing is gathering all these BigData systems underneath its layers (e.g. Paas, Saas, Iaas) to free developers from large-scale issues, such as: deployment, distribution, resiliency, security, and performance. Several companies around the globe rely on cloud computing to build robust and reliable services for their business operations (e.g. eBay, Amazon, Skype) mainly to handle heavy load conditions (e.g. seasonal sales, Internet-scale malicious attacks). Testing robustness and reliability of cloud computing services is a hard activity, the state of the art shows that the existing testing techniques suffer to handle aspects, such as: the scale of the cloud, the dynamism of the nodes, and the amount of data and load. In general, these testing techniques rely on a combination of unit tests with some mocking approach that may hide the cloud aspects and may not be suited for large-scale testing. The TOOM project is planned to present a solution for testing robustness of cloud computing services built on top of P2P technology to address scalability and dynamism aspects. The main contributions lie on two main steps. The first one is to validate the overall resilience and reliability of cloud services. The second one is to reproduce large-scale stress loads, such as Distributed Denial of Service (DDoS) and peak loads, either gathered from the real load traces or synthetically generated. We plan to leverage data warehouse technology to house real load traces and use them during testing. To generate synthetic loads, we plan to use known load patterns or adapt them to new load trends. To assess the effectiveness of the TOOM outcomes, we will reproduce stress loads submitted by P2P technology across the cloud infrastructure on top of step-stress testing methodologies. In this manner, we can progressively increase the load in orders of magnitude up to a peak load. Then, we will measure the effectiveness either by code coverage whether the SUT is open-source, by the quality of service (QoS) of the SUT, or by the coverage of network and computing components used by the cloud computing services.

7.4. International Initiatives

7.4.1. Inria International Partners

7.4.1.1. Informal International Partners

The four main research partners of the team are:

- Politecnico di Milano (Italy) - DB Group, specially with Marco Brambilla
- TU Wien (Austria) - BiG Group, specially Manuel Wimmer
- Politecnica de Catalunya (Spain) - GESSI Group, specially Xavier Franch
- Universitat Poliècnica de València (Spain) - ISSI Group, specially José H. Canós

7.5. International Research Visitors

7.5.1. Visits of International Scientists

Javier Criado (University of Almeria, Spain), June-July

7.5.1.1. Internships

Rolandi, María Belén

Subject: Democracy in Open Source projects

Date: from May 2014 until Oct 2014

Institution: Universidad Nacional del Centro de la Provincia de Buenos Aires (Argentina)

7.5.2. Visits to International Teams

7.5.2.1. Research stays abroad

In March, M. Tisi visited the National Institute of Informatics (NII) of Tokyo, Japan, for one month, in the frame of a collaboration on bidirectionalization of model-transformation languages.

8. Dissemination

8.1. Promoting Scientific Activities

8.1.1. Scientific events organisation

8.1.1.1. Member of the organizing committee

The AtlanMod team has coorganized the following events:

- A. Gómez was publicity co-chair for the 10th European Conference on Modelling Foundations and Applications (ECMFA 2014).
- M. Tisi has coorganized the Scalable Model Driven Engineering Workshop (BigMDE'14) within the STAF federated conference.
- J. Cabot coorganized the CloudMDE workshop within the MoDELS Conference and was the demo co-chair for the International Conference on Web Engineering (ICWE)

8.1.2. Scientific events selection

8.1.2.1. Chair of conference program committee

J. Cabot was the PC Cochair for the 10th European Conference on Modelling Foundations and Applications (ECMFA 2014)

8.1.2.2. Member of the conference program committee

Participation to conference program committees/boards:

- Jordi Cabot:
  - International: International Conference on Model-Driven Engineering Language and Systems, European Conference on modeling foundations and applications, International Conference on Web Engineering, International Conference on Advanced Information Systems Engineering, International Conference on Model Transformations, Automated Software Engineering Conference, OCL workshop.
  - National: French national conference on MDE, Spanish national conference on MDE.
- Massimo Tisi:
  - International: International Conference on Model Transformation (ICMT2014), Software Language Engineering conference (SLE2014 - Industrial Track), Transformation Tool Contest (TTC2014), International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM2014)
- Gerson Sunye:

  - International: International Conference on Model Transformation (ICMT2014), International Conference on Software Engineering and Knowledge Engineering, International Workshop on Large-Scale Testing, International Workshop on Testing The Cloud.
- Salvador Martínez:
  - International: Metamodelling for Healthcare Systems (MMHS) Workshop
- Javier Canovas:
  - International: European Conference on Modelling Foundations and Applications, International Conference on Web Engineering, International Conference on Model Driven Engineering Languages and Systems, International Conference on Automated Software Engineering, Workshop on Modeling Social Media
- Abel Gómez:
  - International: 23rd International Conference on Information Systems Development.
- Hugo Bruneliere:
  - International: 8th Symposium on the Maintenance and Evolution of Service-Oriented Systems and Cloud-Based Environments

8.1.2.3. Reviewer

- Abel Gómez:
  - International: 40th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM 2015) (external reviewer).

8.1.3. Journal

8.1.3.1. Member of the editorial board

Jordi Cabot is on the editorial board of the Journal of Object Technology and Software and Systems Modeling journals

8.1.3.2. Reviewer

AtlanMod members have collaborated in the reviewing process for the following journals this year

- Javier Canovas: Software and Systems Modeling Journal.
- Jordi Cabot: IEEE Software, IEEE Transactions on Software Engineering Journal, Journal of Systems and Software, Information and Software Technology Journal, Software and Systems Modeling Journal, Science of Computer Programming journal, Journal of Software: Evolution and Process
- Massimo Tisi: Transactions on Software Engineering Journal (TSE), Journal of Logic and Algebraic Programming (JLAP), Journal of Software: Evolution and Process (JSEP), NeuroComputing, Journal on Web Engineering (JWE)
- Gerson Sunyé: Transactions on Software Engineering Journal (TSE), IBM Journal of Research and Development, Journal of System and Software, Distributed Computing Journal, Software: Practice and Experience Journal.

8.2. Teaching - Supervision - Juries

8.2.1. Teaching

License: M. Tisi, Projet Integrateur PRIME, 12h, L3, Ecole des Mines de Nantes

License: M. Tisi, Projet IPIPIP, 8h, L3, Ecole des Mines de Nantes

Master: M. Tisi, Bases de Données, 30h, M1, Ecole des Mines de Nantes

Master: M. Tisi, Interaction Homme-Machine, 14h, M1, Ecole des Mines de Nantes

Master: M. Tisi, Model-Driven Engineering II, 45h, M2, Ecole des Mines de Nantes, Formation par Apprentissage

Master: M. Tisi, Sensibilisation Recherche, 12h, M2, Ecole des Mines de Nantes, Formation par Apprentissage

Master: J. Cabot, Model-Driven Engineering I, 45h, M2, Ecole des Mines de Nantes, Formation par Apprentissage

Master: G. Sunye, Model-Driven Engineering, 30h, M2, Université de Nantes

Master: G. Sunye, Software Engineering, 48h, M1, Université de Nantes

Master: G. Sunye, Software Testing, 48h, M1, Université de Nantes

Master: G. Sunye, Software Development, 24h, M1, Université de Nantes

License: S. Martínez, Bases de Données, 28h, L3, Ecole des Mines de Nantes

License: S. Martínez, Interaction Homme-Machine, 14h, L3, Ecole des Mines de Nantes

Master: S. Martínez, Master MIAGE - Model Transformations, 6h, Université de Nantes

License: S. Martínez, Domain Specific Languages, 3h, L3, Ecole des Mines de Nantes

8.2.2. Supervision

MSc: Pau Martí: Generación de Familias de Documentos en DPL: Soporte a componentes parcialmente instanciados, Universitat Politècnica de València, September 2014, M. Carmen Penadés and Abel Gómez.

PhD in progress: Ahmed Ezzat Labib, Automatic reconstruction and analysis of semantic network security policies from deployed security components, Universitat Politècnica de València, José H. Canós, Abel Gómez

PhD in progress: Amine Benellanem, Scalability of model transformations, Jordi Cabot, Massimo Tisi and Gerson Sunye.

PhD in progress: Gwendal Daniel, Efficient storage of large models, Jordi Cabot, Massimo Tisi and Gerson Sunye.

PhD in progress: Michel Albonico, Model-driven testing of cloud environments, Jordi Cabot and Gerson Sunye.

PhD in progress: Mohamed Boussaa, An Architecture for Testing Large-Scale Dynamic Distributed Systems, Université de Rennes 1, Benoit Baudry, Olivier Barais, Gerson Sunyé.

PhD in progress: Kevin Corre, Model-based Trust, Université de Rennes 1–Orange, Olivier Barais, Gerson Sunye, Jean-Michel Crom, Vincent Frey

PhD : Carlos A. González, Pragmatic Model Verification, Jordi Cabot. Thesis defended on October 2014

PhD : Salvador Martínez Pérez, Automatic reconstruction and analysis of semantic network security policies from deployed security components, Jordi cabot and Frédéric Cuppens. Thesis defended on July 2014

8.2.3. Juries

Jordi Cabot was a reviewer for the thesis of Jokin Garcia (Basque Country University), Oscar Sanchez (University of Murcia) and Eugene Doma (University of Sydney) and member of the jury for the thesis of David Aguilera (Technical University of Catalonia).

Massimo Tisi was a member of the ph.d. jury of Mehdi IRAQI HOUSSAINI (Arts et métiers ParisTech).

Gerson Sunyé was a member of the Ph.D jury of Jorge Augusto Meira (University of Luxembourg).

8.3. Popularization

In January 2014 the team participated to the Rencontres Inria - Industrie on the theme Répondre aux Défis de L'Ingénierie Logicielle, in which we notably presented our MoDisco project and collaboration [13]. The seminar had many participants from both industry and academia. Moreover, since several years, the AtlandMod team is already very involved in the open source community, notably via its constant activity within the context of the Eclipse Foundation. This activity actually takes different forms: creation and leading/development of Eclipse projects (under *Eclipse.org* or *Eclipse Labs*), participation to the major worldwide community events (i.e.; EclipseCon North America, EclipseCon Europe and EclipseCon France), organization of events targeting the local community (i.e. Eclipse DemoCamps), etc. This year again, the team has been active and visible in terms of concrete contributions to the community. The main remarkable items are the following ones:

- Leading of the MDT MoDisco project (Hugo Bruneliere), the Eclipse reference project concerning model-driven reverse engineering;
- Commitment to other projects directly under *Eclipse.org*: MMT ATL and EMFT EMF Facet (Hugo Bruneliere);
- Commitment to other projects under *EclipseLabs*: EMFToCSP (Jordi Cabot, Carlos Gonzalez) and EMF Views (Juan David Villa Calle, Jokin Garcia Perez, Hugo Bruneliere);
- Creation and development of a new incubation project under *Eclipse Labs*: Collaboro (Javier Canovas);
- Publication of a paper about our experience and feedback on developing and promoting open source projects in Eclipse [23], presented during the Open Source Software for MDE workshop at MODELS 2014 Conference (Hugo Bruneliere and Jordi Cabot);
- Presentation of talks during two main Eclipse events (Hugo Bruneliere): at EclipseCon North America 2014 about EMF Views [36], at EclipseCon France 2014 about Neo4EMF [35].
- In the context of the ARTIST EU collaborative project, we are strongly involved in the official ARTIST Open Source Release [14], notably by developing and making available several (MoDisco-based) model discovery components from Java and SQL.

Generally, the team visible presence under Eclipse is also an efficient way to continue active collaborations with industrial partners, such as Mia-Software (Sodifrance) on MoDisco - EMF Facet, and Obeo on ATL.

9. Bibliography

Major publications by the team in recent years

[1] M. ALBERT, J. CABOT, C. GÓMEZ, V. PELECHANO. *Automatic generation of basic behavior schemas from UML class diagrams*, in "Software and System Modeling", 2010, vol. 9, n^{o} 1, pp. 47-67

[2] D. AMELLER, C. AYALA, J. CABOT, X. FRANCH. *Non-functional Requirements in Architectural Decision Making*, in "IEEE Software", March 2013, vol. 30, n^{o} 2, pp. 61-67, http://hal.inria.fr/hal-00822203

[3] M. BRAMBILLA, J. CABOT, M. WIMMER. *Model-Driven Software Engineering in Practice*, Morgan & Claypool, September 2012, 182 p. , http://hal.inria.fr/hal-00755006

[4] H. BRUNELIÈRE. *An Open Source-Based Approach for Industrializing Research Tools*, in "OW2Con 2011, Session "Open Source Innovation Catalyst" - ICT Labs, IRILL", Paris, Issy Les Moulineaux, France, November 2011, http://hal.inria.fr/hal-00639559/en

[13] http://videos.rennes.inria.fr/InriaMeito/index-HugoBruneliere.html

[14] http://www.artist-project.eu/open-source-package

[5] J. Cabot, R. Clarisó, E. Guerra, J. de Lara. *Verification and validation of declarative model-to-model transformations through invariants*, in "Journal of Systems and Software", 2010, vol. 83, n^{o} 2, pp. 283-302

[6] J. Cabot, R. Clarisó, D. Riera. *On the verification of UML/OCL class diagrams using constraint programming*, in "Journal of Systems and Software", 2014, vol. 93, pp. 1–23, http://dx.doi.org/10.1016/j.jss.2014.03.023

[7] J. Cabot, R. Pau, R. Raventós. *From UML/OCL to SBVR specifications: A challenging transformation*, in "Inf. Syst.", 2010, vol. 35, n^{o} 4, pp. 417-440

[8] J. Canovas, F. Jouault, J. Cabot, J. Garcia Molina. *API2MoL: Automating the building of bridges between APIs and Model-Driven Engineering*, in "Information and Software Technology", October 2011 [*DOI :* 10.1016/j.infsof.2011.09.006], http://hal.inria.fr/hal-00642154/en/

[9] A. Vignaga, F. Jouault, M. C. Bastarrica, H. Brunelière. *Typing artifacts in megamodeling*, in "Springer Journal on Software and Systems Modeling", February 2011, vol. 10, n^{o} 1 [*DOI :* 10.1007/s10270-011-0191-2], http://www.springerlink.com/content/j212100qq405n5q1/

Publications of the year

Doctoral Dissertations and Habilitation Theses

[10] S. Martinez. *Automatic reconstruction and analysis of security policies from deployed security components*, Ecole des Mines de Nantes, June 2014, https://tel.archives-ouvertes.fr/tel-01065944

Articles in International Peer-Reviewed Journals

[11] H. Brunelière, J. Cabot, G. Dupé, F. Madiot. *MoDisco: a Model Driven Reverse Engineering Framework*, in "Information and Software Technology", August 2014, vol. 56, n^{o} 8, pp. 1012-1032 [*DOI :* 10.1016/j.infsof.2014.04.007], https://hal.inria.fr/hal-00972632

[12] J. Cabot, R. Clarisó, D. Riera. *On the Verification of UML/OCL Class Diagrams using Constraint Programming*, in "Journal of Systems and Software", March 2014 [*DOI :* 10.1016/j.jss.2014.03.023], https://hal.inria.fr/hal-00974321

[13] J. Canovas, J. Garcia Molina. *Extracting Models from Source Code in Software Modernization*, in "Software & Systems Modeling", May 2014, vol. 13, n^{o} 2, 21 p. [*DOI :* 10.1007/s10270-012-0270-z], https://hal.inria.fr/hal-00752688

[14] E. A. G. Carnelossi, E. Lerat, H. Henri, S. Martinez, C. M. A. Carareto, C. Vieira. *Specific activation of an I-like element in Drosophila interspecific hybrids*, in "Genome Biology and Evolution", June 2014, pp. 1806-17, https://hal.inria.fr/hal-01079996

[15] J. S. Cuadrado, J. Canovas, J. Garcia Molina. *Applying model-driven engineering in small software enterprises*, in "Science of Computer Programming", September 2014, vol. 89, n^{o} B, 22 p. , https://hal.inria.fr/hal-00869356

[16] C. A. González Pérez, J. Cabot. *Formal Verification of Static Software Models in MDE: A Systematic Review*, in "Information and Software Technology", March 2014 [*DOI :* 10.1016/j.infsof.2014.03.003], https://hal.inria.fr/hal-00972004

[17] A. GÓMEZ, M. C. PENADÉS, J. H. CANÓS, M. R. S. BORGES, M. LLAVADOR. *A Framework for Variable Content Document Generation with Multiple Actors*, in "Information and Software Technology", January 2014, vol. 56, n^{o} 9, pp. 1101-1121 [*DOI :* 10.1016/J.INFSOF.2013.12.006], https://hal.inria.fr/hal-00965542

[18] A. MENYCHTAS, K. KONSTANTELI, J. ALONSO, L. ORUE-ECHEVARRIA, J. GORRONOGOITIA, G. KOUSIOURIS, C. SANTZARIDOU, H. BRUNELIÈRE, B. PELLENS, P. STUER, O. STRAUSS, T. SENKOVA, T. VARVARIGOU. *Software modernization and cloudification using the ARTIST migration methodology and framework*, in "Scalable Computing: Practice and Experience (Scientific International Journal for Parallel and Distributed Computing)", July 2014, vol. 15, n^{o} 2, pp. 131-152 [*DOI :* 10.12694/SCPE.V15I2.938], https://hal.inria.fr/hal-01021002

[19] G. SUNYÉ, E. CUNHA DE ALMEIDA, Y. LE TRAON, B. BAUDRY, J.-M. JÉZÉQUEL. *Model-Based Testing of Global Properties on Large-Scale Distributed Systems*, in "Information and Software Technology", 2014, https://hal.inria.fr/hal-00942576

International Conferences with Proceedings

[20] A. BENELALLAM, A. GÓMEZ, G. SUNYÉ, M. TISI, D. LAUNAY. *Neo4EMF, a Scalable Persistence Layer for EMF Models*, in "ECMFA- European conference on Modeling Foundations and applications", York, UK, United Kingdom, J. CABOT, J. RUBIN (editors), Springer International Publishing, April 2014, vol. 8569, pp. 230-241 [*DOI :* 10.1007/978-3-319-09195-2_15], https://hal.inria.fr/hal-00968516

[21] A. BENELALLAM, M. TISI, I. RATH, B. IZSO, D. S. KOLOVOS. *Towards an Open Set of Real-World Benchmarks for Model Queries and Transformations*, in "BigMDE", York, UK, United Kingdom, CEUR Workshop Proceedings, July 2014, https://hal.inria.fr/hal-01035450

[22] M. BRAMBILLA, A. MAURI, E. UMUHOZA. *Extending the Interaction Flow Modeling Language (IFML) for Model Driven Development of Mobile Applications Front End*, in "Mobile Web Information Systems", Barcelona, Spain, August 2014, https://hal.inria.fr/hal-01026316

[23] H. BRUNELIÈRE, J. CABOT. *On Developing Open Source MDE Tools: Our Eclipse Stories and Lessons Learned*, in "Open Source Software for Model Driven Engineering (OSS4MDE'14) workshop - a MODELS 2014 Satellite Event", Valencia, Spain, September 2014, https://hal.inria.fr/hal-01068885

[24] J. CANOVAS, J. CABOT. *Composing JSON-based Web APIs*, in "International Conference on Web Engineering", Toulouse, France, July 2014, https://hal.inria.fr/hal-00974938

[25] J. H. CANÓS-CERDÁ, J. SÁNCHEZ-DÍAZ, V. ORTS, M. C. PENADÉS, A. GÓMEZ, M. R. S. BORGES. *Turning Emergency Plans into Executable Artifacts*, in "The 11th International Conference on Information Systems for Crisis Response and Management", State College, Pennsylvania, United States, S. HILTZ, M. PFAFF, L. PLOTNICK, P. SHIH (editors), Proceedings of the 11th International ISCRAM Conference, May 2014, pp. 496-500, ISBN: 978-0-692-21194-6 Available under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0) License, https://hal.inria.fr/hal-00965587

[26] V. COSENTINO, M. TISI, J. L. CANOVAS IZQUIERDO. *A Model-Driven Approach to Generate External DSLs from Object-Oriented APIs*, in "41st International Conference on Current Trends in Theory and Practice of Computer Science", Pec pod Sněžkou, Czech Republic, January 2015, https://hal.inria.fr/hal-01094214

[27] G. Daniel, G. Sunyé, A. Benelallam, M. Tisi. *Improving memory efficiency for processing large-scale models*, in "BigMDE", York, UK, United Kingdom, University of York, July 2014, https://hal.inria.fr/hal-01033188

[28] J. Garcia, O. Díaz, J. Cabot. *An Adapter-Based Approach to Co-evolve Generated SQL in Model-to-Text Transformations*, in "26th International Conference on Advanced Information Systems Engineering", Thessaloniki, Greece, June 2014, https://hal.inria.fr/hal-00974322

[29] C. A. González Pérez, J. Cabot. *Test Data Generation for Model Transformations Combining Partition and Constraint Analysis*, in "7th International Conference on Model Transformation (ICMT) 2014", York, United Kingdom, July 2014, https://hal.inria.fr/hal-00972008

[30] A. Gómez, P. Martí, M. C. Penadés, J. H. Canós. *DPLFW: a Framework for the Product-Line-Based Generation of Variable Content Documents*, in "Proceedings of the Demonstrations Track of the ACM/IEEE 17th International Conference on Model Driven Engineering Languages and Systems (MODELS 2014)", Valencia, Spain, October 2014, https://hal.inria.fr/hal-01096362

[31] F. Jouault, M. Tisi, D. Jerome. *fUML as an Assembly Language for MDA*, in "6th International Workshop on Modeling in Software Engineering (MiSE 2014)", Hyderabad, India, June 2014, https://hal.inria.fr/hal-00966441

[32] A. E. Labib, M. C. Penadés, J. H. Canós, A. Gómez. *Enforcing reuse and customization in the development of learning objects: a product line approach*, in "30th ACM/SIGAPP Symposium On Applied Computing (SAC 2015)", Salamanca, Spain, April 2015 [*DOI :* 10.1145/2695664.2695991], https://hal.inria.fr/hal-01095594

[33] M. C. Penadés, P. Martí, J. H. Canós, A. Gómez. *Product Line-based customization of e-Government documents*, in "PEGOV : Personalization in e-Government Services, Data and Applications", Aalborg, Denmark, N. Loutas, F. Narducci, A. Ojo, M. Palmonari, C. Paris, G. Semeraro (editors), UMAP 2014 Extended Proceedings, CEUR-WS, July 2014, vol. 1181, https://hal.inria.fr/hal-01030791

[34] E. Robles, J. M. Rivero, M. Urbieta, J. Cabot. *Improving the scalability of Model driven Web engineering approaches with runtime transformations*, in "14th International Conference on Web Engineering", Toulouse, France, July 2014, https://hal.inria.fr/hal-00974323

Conferences without Proceedings

[35] A. Benelallam, H. Brunelière. *Neo4EMF : when big models are no longer an issue*, in "EclipseCon France 2014", Toulouse, France, June 2014, https://hal.inria.fr/hal-01026219

[36] H. Brunelière, J. D. Villa Calle. *EMF Views: Dealing with several interrelated EMF models*, in "EclipseCon 2014 - Modeling Symposium", San Francisco, United States, March 2014, https://hal.inria.fr/hal-00974711

Scientific Books (or Scientific Book chapters)

[37] B. Combemale, D. J. Pearce, O. Barais, J. J. Vinju (editors). *fUML as an Assembly Language for Model Transformation*, Springer International PublishingVästerås, Sweden, September 2014, vol. 8706, 20 p. [*DOI :* 10.1007/978-3-319-11245-9_10], https://hal.inria.fr/hal-01096634

References in notes

[38] M. BARBERO, F. JOUAULT, J. BÉZIVIN. *Model Driven Management of Complex Systems: Implementing the Macroscope's vision*, in "Proceedings of the 15th Annual IEEE International Conference and Workshop on Engineering of Computer Based Systems (ECBS 2008), 31 March - 4 April 2008, Belfast, Northern Ireland", IEEE Computer Society, 2008, pp. 277–286

[39] J. BÉZIVIN. *From Object Composition to Model Transformation with the MDA*, in "TOOLS '01: Proceedings of the 39th International Conference and Exhibition on Technology of Object-Oriented Languages and Systems (TOOLS39)", Washington, DC, USA, IEEE Computer Society, 2001, pp. 350–354

[40] J. BÉZIVIN, F. BÜTTNER, M. GOGOLLA, F. JOUAULT, I. KURTEV, A. LINDOW. *Model Transformations? Transformation Models!*, in "Model Driven Engineering Languages and Systems, 9th International Conference, MoDELS", Genova, Italy, LNCS, Springer Berlin / Heidelberg, October 1-6 2006, n^{o} 4199, pp. 440–453

[41] J. BÉZIVIN. *On the Unification Power of Models*, in "Software and System Modeling", 2005, vol. 4, n^{o} 2, pp. 171–188

[42] J. BÉZIVIN, O. GERBÉ. *Towards a Precise Definition of the OMG/MDA Framework*, in "ASE'01, Automated Software Engineering", San Diego, USA, IEEE Computer Society Press, 2001, pp. 273-280

[43] J. BÉZIVIN, R. HECKEL. *04101 Summary – Language Engineering for Model-driven Software Development*, in "Language Engineering for Model-Driven Software Development", Dagstuhl, Germany, J. BÉZIVIN, R. HECKEL (editors), Dagstuhl Seminar Proceedings, Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2005, n^{o} 04101

[44] J. BÉZIVIN, F. JOUAULT. *Using ATL for Checking Models*, in "Proceedings of the International Workshop on Graph and Model Transformation (GraMoT)", Tallinn, Estonia, 2005

[45] J. CABOT, R. CLARISÓ, D. RIERA. *On the verification of UML/OCL class diagrams using constraint programming*, in "Journal of Systems and Software", 2014, vol. 93, pp. 1–23, http://dx.doi.org/10.1016/j.jss.2014.03.023

[46] G. CANFORA, M. DI PENTA, L. CERULO. *Achievements and challenges in software reverse engineering*, in "Commun. ACM", April 2011, vol. 54, pp. 142–151, http://doi.acm.org/10.1145/1924421.1924451

[47] M. DIDONET DEL FABRO, J. BÉZIVIN, F. JOUAULT, P. VALDURIEZ. *Applying Generic Model Management to Data Mapping*, in "Proceedings of the Journées Bases de Données Avancées (BDA05)", 2005

[48] F. JOUAULT, F. ALLILAIRE, J. BÉZIVIN, I. KURTEV. *ATL: a Model Transformation Tool*, in "Science of Computer Programming", 2008, vol. 72, n^{o} 3, Special Issue on Second issue of experimental software and toolkits (EST), pp. 31–39

[49] F. JOUAULT, J. BÉZIVIN. *KM3: a DSL for Metamodel Specification*, in "Proceedings of 8th IFIP International Conference on Formal Methods for Open Object-Based Distributed Systems, LNCS 4037", Bologna, Italy, 2006, pp. 171–185

[50] I. KURTEV, J. BÉZIVIN, F. JOUAULT, P. VALDURIEZ. *Model-based DSL Frameworks*, in "Companion to the 21st Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications, OOPSLA 2006, October 22-26, 2006, Portland, OR, USA", ACM, 2006, pp. 602–616