IN PARTNERSHIP WITH:
Ecole des Mines de Nantes

Activity Report 2011

Team ATLANMOD

Modeling Technologies for Software Production, Operation, and Evolution

RESEARCH CENTER
Rennes - Bretagne-Atlantique

THEME
Distributed Systems and Services

Table of contents

Team ATLANMOD

Keywords: Model-Driven Engineering, Software Engineering

1. Members

Research Scientists
Javier Cánovas [Postdoc, since October 2011]
Sagar Sen [Postdoc- École des mines de Nantes, since September 2011]
Fabian Büttner [Postdoc, CESAR, since October 2011]
Robert Tairas [Postdoc]

Faculty Members
Jordi Cabot [Team Leader, Associate Professor, École des mines de Nantes]
Frédéric Jouault [Associate Professor, École des mines de Nantes]
Massimo Tisi [Associate Professor, École des mines de Nantes]

Technical Staff
Hugo Brunelière [Engineer, École des mines de Nantes]
Guillaume Doux [Engineer, IDM++ - Université de Nantes until August 31st, INRIA - Mia-Software Technology transfer action after]
Cauê Avila Clasen [Engineer, CESAR]
Salvador Martínez Pérez [Engineer-OPEES, PhD Student from November 2011]
Wolfgang Kling [Engineer ARMINES, GALAXY]
Carlos González Pérez [Engineer-CESAR, PhD Student from October 2011]
Valerio Cosentino [PhD Student, CIFRE IBM]
Vincent Mahé [Engineer, IDM++ until February 2011, Université de Nantes]

Visiting Scientists
Dennis Wagelaar [Postdoc, January-February 2011]
Víctor Garía Pau [Master Thesis, April-July 2011]

Administrative Assistants
Cecile Derouet [Part-time, since September 2011]
Hanane Maaroufi [Part-time, until August 2011]

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. It has been shown that adoption of MDE increases productivity and quality of the software engineering process.

In this sense, AtlanMod is developing pioneering solutions to solve core research challenges in MDE and to ensure its successful application on relevant industrial problems.

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 has progressively developed a complete modeling framework [48] [58] that is now well accepted in the scientific community. This framework provides core MDE components (described in several research papers e.g., [56] [57] [51] [55] [7]) for (meta)model definition and manipulation. The iterative definition of this conceptual framework has been validated by the construction of the MDE toolbox AmmA (for AtlanMod Model Management Architecture) available in Eclipse. This is at the same time a research platform and an industrial toolbox.

The AmmA platform is 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, the AmmA 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, TMF, 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.

2.3. Highlights

- Team consolidation after the change of the team leader. The level of activity of the team has not been affected by the process
- New projects and technology transfer actions that guarantee the funding for the immediate future and offer new insights on interesting industrial problems.
- Opening of new research lines with the integration of new PhD Students. In particular, we would like to mention a new line on the use of models to improve the security of information systems in collaboration with F. Cuppens (Télécom Bretagne)
- Very good ratio of number of published papers per faculty member of the team.
- International team. AtlanMod has members from eight different nationalities. All these different research backgrounds are definitely a plus for the team.

3. Scientific Foundations

3.1. MDE Foundations

MDE can be seen as a generalization and abstraction of object technology allowing to map more abstract organizations on class-based implementations. In MDE, (software) models are considered as the unifying concept [48].

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. Therefore, rigorous techniques for model definition and manipulation are the basis of any MDE framework.

[1] urlhttp://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 [56] and [49]. MDE promotes *unification by models*, like object technology proposed in the eighties *unification by objects* [47]. 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 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 [51]. 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 [2]. 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 [50]. 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 [7]).

When following the previously described principles, one may take advantage of the uniformity of the MDE organization. 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. Considering models of products (e.g., software artifacts like UML) and models of processes (e.g., software processes like SPEM) may lead to a dual process/product organization. Considering transformation models, weaving models, and traceability models as special cases of correspondence models may also lead to simplicity and efficiency of implementations. These are some of the use cases that are being explored in the team.

4. Application Domains

4.1. Introduction

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 [54] and to system engineering [44]. There are a lot of problems in these application domains that may be addressed by means of modeling and model transformation techniques. For the core techniques, we are now focusing on solving the scalability problem to make sure they can be successfully adopted by our industrial partners in the context of large and complex software systems.

Nevertheless, the team has indeed selected a set of vertical and horizontal domains that AtlanMod finds specially interesting. In what follows we describe four of them.

4.2. Reverse Engineering

One important and original 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 [53].

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, an industrial PhD in partnership with IBM France has recently started 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 and is developing accordingly the open source Eclipse-MDT 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.

4.3. System Interoperability

Recently, MDE has also been used as a solution for system interoperability. In this new context, transformations are performed while systems are in execution. For example, a set of transformations may keep two different tools synchronized, by exchange of structured data or even by interpretation of complex events. This approach revisits tool interoperability by explicitly representing the associated metamodels (if needed, deduced from the tool API or storage format), defining mappings between them and using those mappings to automatically generate transformations between these *virtual* tool representation. The establishment of correspondences between technical spaces (e.g., Eclipse Modeling and Microsoft DSL tools or OSLO [45]) follows a similar schema.

For system interoperability it is necessary to provide mechanisms to interoperate with the system components at hand, normally through the APIs they provide. When using MDE for system interoperability, these APIs must be represented and manipulated as models. This model-based view of APIs allow designers to work at a much better abstraction level when bridging between systems or components thanks to the homogeneous treatment of all APIs involved in the software system. As an example, models obtained from API objects could be used in scenarios such as web interoperability and Graphical User Interface (GUI) manipulation at runtime.

Automating the building of these bridges would promote system interoperability. In this context, we have developed API2MoL [14], which is an approach aimed at automating the implementation of API-MDE bridges. API2MoL is based on a rule-based declarative language to specify mappings between the artefacts of a given API (e.g., API classes in object-oriented APIs) and the elements of a metamodel that represents this API in the MDE technical space. Thus, a mapping definition provides the information which is necessary to build a bridge for a concrete API specification and metamodel.

Another example of bridge between technical spaces is the one developed during the IDM++ project between the MDE and the Constraint Programming (CP) technical spaces to solve configuration problems [22]. The use case supporting this example was focused on the generation of build configurations for Eclipse distributions. In this approach, the information concerning the different components available for the configuration is stored in a configuration model. The model was then converted into a format usable by a configuration CP engine using model transformations techniques. The CP solver was then in charge of finding a configuration solution respecting all dependencies in the model. Finally, an executable configuration description was generated based on the CP feedback to drive the final distribution build.

The main advantage of this kind of approach bridging several technical spaces is the use of the right tooling for the right problem. In particular, for this example of build configuration generation, the dependency resolution between components was clearly a configuration problem. The main motivation for this choice is that configuration is a widely studied problem in the CP community and as a consequence the solutions proposed in this technical space are efficient for this class of problem. On the other side, the concepts used in the CP community are often in a relatively low level of abstraction and MDE can help raising the level of abstraction for describing the problem and so help the final users. This concrete example shows clearly how technical spaces (here MDE and CP) can benefits from each others.

4.4. 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.

We have started a Phd thesis in this domain intended to investigate the construction of a model-driven automatic reverse engineering mechanism (implemented as an extension of the MoDisco project) 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 over the system. 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.

As a first step we intend to apply model-driven techniques for the extraction of high level model representations of security policies enforced by firewalls. 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. 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 firewall vendors will be also enabled.

4.5. 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. The AtlanMod team is considering software quality aspects in two areas that are described in the following paragraphs: formal verification of models and corpus-based domain-specific language (DSL) analysis.

In the context of MDE, models are expressions of a conceptualization (metamodel) of their respective domain. Therefore precise definitions of our metamodels are important to get the subsequent MDE activities right, such as testing, concrete syntax definition, model interchange, code generation, and any more. In this sense, metamodels establish a single point of truth. The Object Constraint Language (OCL) of the OMG is widely accepted as a standard language to provide such precise definitions.

However, precise metamodels are complex artifacts. To get them right, systematic approaches to quality assurance are required. As a solution to this problem, we propose formal checking of model satisfiability (can we express the desired things), and formal checking of unsatisfiability (does the metamodel not have models with undesirable properties). As both activities constitute NP-hard problems, they have to be put carefully into algorithms. The team maintains the tool EMFtoCSP which translates the problem into the domain of constrain 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.

In terms of DSLs, an equally important aspect of software language engineering other than the initial development process is the identification of quality characteristics of the language. We are currently investigating the use of corpus-based analysis to identify language characteristics. A corpus in this case consists of artifacts or models of the DSL that have been generated by its users. We aim to extract information based on the corpus of a DSL to identify various characteristics that can potentially prove useful for the language engineer in his or her efforts to improve the language. Such information can complement other quality measurements including the formal verification method described in the previous paragraph and user feedback-based evaluations. In addition to an initial investigation of cloning in DSLs, our corpus-based analysis is also investigating metamodel element instance and relationship analyses

5. Software

5.1. The ATL Model Transformation Language

URL: http://www.eclipse.org/m2m/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 [8]. 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 implemented in [43] 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.

In [31] 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 enable also transformations that generate infinite target models, extending the application space of the model-transformation paradigm.

The latest evolution of the ATL engine is a full reactive version, able to activate the minimal computation for responding to updates or request on the involved models. This engine is studied to scalably support large ATL networks. In this line we also introduced an algorithm for simplifying ATL transformation chains.

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).

5.3. 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 [2]) 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.
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 problem of constraint logic programming and employs the Eclipse CLP solver [3] 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 [52] 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.

5.4. AMW (AtlanMod Model Weaver)

URL: http://wiki.eclipse.org/AMW

AMW is an open source Eclipse project, under the Eclipse Public License (EPL), that provides a generic and extensible tooling dedicated to model weaving. It can be used to establish, represent and manage any kind of links (correspondences) between elements coming from different models (or metamodels). These links are stored in separated weaving models, allowing them to be exchanged and reused in various contexts using different modeling techniques. Thus, weaving models are actually relevant in many concrete application scenarios where several models are involved: (model driven) tool interoperability, mapping definition, transformation specification, traceability, model annotation, model merging, model comparison, etc.

The AMW tooling is composed of several Eclipse plugins, and relies on the de-facto standard Eclipse Modeling Framework (EMF) for model handling. The provided model weaving workbench comes notably with a base weaving editor, reusable matching algorithms and weaving model serialization capabilities. These components are fully generic, so that they can be extended and adapted to any specific weaving metamodel (defining any kind of weaving links). AMW also offers a connector allowing to both use and produce weaving models as part of model-to-model transformations written in ATL.

[2] http://www.eclipse.org/modeling/mdt/?project=ocl
[3] http://eclipseclp.org/

AMW is being used by more than 40 user sites, including research labs and major companies (NASA, BAE, Versata, Mia-Software, Obeo, etc.). Currently part of the Eclipse-GMT project (to be terminated in the coming months), AMW is now in the process of being migrated into the Eclipse-M2M ATL project.

5.5. AM3 (AtlanMod MegaModel Management)

URL: http://wiki.eclipse.org/index.php/AM3

The AtlanMod Megamodel Management tool offers several functionalities for modeling in the large [3], i.e. for handling several related models (either terminal models, metamodels or transformation models) used as part of a complex modeling project.

The main component in AM3 is a generic megamodel manager that allows the user to browse and manipulate a set of related models. This manager knows the semantic relations between all these models. These relations are often associated to a given weaving model allowing not only navigating the traces between models, but also the traces between model elements. Since the links are stored externally as weaving models, the participating models do not get polluted and may be used as they are. Furthermore it is possible to handle multiple traceability chains going through similar models.

AM3 provides also a textual domain-specific language for model management called MoScript (URL: http://wiki.eclipse.org/MoScript). With MoScript, users can automate model management tasks by means of textual scripts written in an extension of the OCL language. For instance, user may write queries (based on model content, structure, relationships, and behaviour derived through on-the-fly simulation) to retrieve models from model repositories, manipulate them (e.g., by running transformations on sets of models), and store them back in the repository. MoScript also allows to populate and update the megamodel automatically by doing reverse engineer of simple modeling artifact repositories.

The generic tool for megamodel management has been used by different partners for several use cases like operationalization of chains of transformations.

5.6. Virtual EMF (Model Virtualization)

URL: http://code.google.com/a/eclipselabs.org/p/virtual-emf/

Virtual EMF is an Eclipse plugin built on top of EMF that enables the creation and manipulation of *virtual models*, i.e., models whose elements do not contain concrete data, but are rather proxies to elements contained in other models. The idea is related to that of model composition, as it aims capturing the (often overlapping) concepts a set of models as one single global model. This is a frequently faced problem as, in complex scenarios, modelers often have to deal with a large number of heterogeneous and interrelated models, and most times the view a specific kind of user requires does not correspond to any of these models, but is a combination of cross-domain information scattered among several ones.

Current composition techniques rely on the materialization of the composed model, an approach that poses some important limitations in terms of (i) *efficiency*, as they do not scale (the data duplication mechanism they use implies in extra memory usage and time-consuming generation of the composed model), (ii) *synchronicity*, as updates in the composed model are not propagated to the original ones (or vice-versa), thus losing consistency, or even (iii) *interoperability*, as in some cases the composed model requires a specific API/tool to be handled.

Virtual EMF allows overcoming the limitatins above. A virtual model provides to tools/user the *illusion* of working with a regular model whereas, in fact, all model access and manipulation requests are transparently redirected to its set of *virtualized* models. It serves as a centralized and transparent access point to a set of interconnected models, allowing users to easily compose, weave and link them. thus providing the following beneficial properties:

- *Interoperability:* it behaves as a normal model. Therefore, compatibility with existing EMF-based solutions/tools (e.g. models transformations, model editors, ...) is guaranteed;
- *Synchronization:* changes are automatically and transparently propagated between virtual and original models;
- *Scalability:* support for very big models;
  - low memory usage: no data duplication, direct access to original model elements;
  - faster generation time: no need for (time-consuming) information cloning operations (e.g. executing a model transformation);
- *Genericity:* support for several types of inter-model relationships (e.g. merge, association, filter) and extension capabilities for their semantics.

Virtual EMF is available as an open-source project on Eclipse Labs. It has been presented in a talk in EclipseCon Europe 2011 and contributed by the AtlanMod team to the CESAR project.

5.7. EMF Facet

URL: http://www.eclipse.org/modeling/emft/facet/

EMF Facet is an open source Eclipse project, under the Eclipse Public License (EPL), that provides a generic and extensible framework dedicated to the dynamic and non-intrusive extension of models. It can be used to extend already existing metamodels with additional concepts and properties, the corresponding models being then transparently augmented, reduced or modified accordingly at runtime. Such a metamodel extension is called a facet, and can be specified on top of any metamodel in EMF Ecore. The underlying mechanism is based on the runtime execution of queries on the models corresponding to the *faceted* metamodels. Facets are notably particularly relevant for obtaining different views on existing models without having to actually alter them with any extra data.

The EMF Facet framework is composed of several Eclipse plugins, and relies on the de-facto standard Eclipse Modeling Framework (EMF) for model handling. The facet definitions are stored as facet models, allowing them to be exchanged and reused in various contexts. The queries can be implemented using any suitable query language (e.g.; ATL, OCL, Java, XPath), as far as the corresponding adaptors exist and are correctly registered within the framework. The proposed tooling includes dedicated editors for creating, editing and saving both facet and query definitions, the implemented support for Java, OCL and ATL queries, a Table Editor for visualizing query results. An advanced support for the model display customization (e.g.; icons, colors, fonts) is also provided as part of the framework.

EMF Facet is currently intensively used in MoDisco for extracting and displaying different specific views from large models of legacy systems. Its extension and customization capabilities are actually integrated into several MoDisco components, such as notably the MoDisco Model Browser. However, different other integration possibilities will be also explored in the future.

5.8. Portolan (Model-Driven Cartography)

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

Processing large amounts of data to extract useful information is an essential task within companies. To help in this task, visualization techniques have been commonly used due to their capacity to present data in synthesized views, easier to understand and manage. However, achieving the right visualization display for a data set is a complex cartography process that involves several transformation steps to adapt the (domain) data to the (visualization) data format expected by visualization tools. With its simple core principles and set of base generic techniques (metamodeling, model transformation, model weaving, etc), Model-Driven Engineering (MDE) provides the relevant support for bridging the gap between data sets and visualization tools and thus for designing and implementing Cartography solutions. The proposed Portolan prototype is a concrete illustration of both a model-based and model-driven Cartography platform. Thus, the objective of Portolan is to facilitate the identification of interoperability solutions between tools by:

1. discovering (at least semi-automatically) maps of given situations in terms of deployed tools and relationships between them;
2. easily navigating and editing these maps;
3. augmenting or specializing them with both manually-entered and computed information;
4. visualizing them, using different customizable ways, in order to facilitate their understanding.

To this aim, the Portolan platform integrates:

- a set of default DSLs like GraphML, KML, Excel;
- visual displays based on Prefuse, Google Maps;
- modeling tools such as ATL language, Ecore modeler, etc.

It includes also an extension mechanism allowing the tool customization for advanced users. This recently developed generic tooling for cartography has already been used during the first action of our collaboration with BNP Paribas, as well as in the context of the IDM++ project.

5.9. The AmmA ToolBox

ATL, AMW, TCS, MoDisco, and AM3 are among the most important *Eclipse.org* components produced by the AtlanMod team. However there are also other components and a lot of functionalities, examples, and use cases made available and necessary to express solutions to many problems. The whole set of contributions composes the AmmA platform.

5.10. Industrialization strategy for research prototypes

Research labs, as priority innovation providers, are also indirectly key actors of the Software Engineering market. However, even if they already initiate the promotion of many new innovations to the industry, an important collaborative effort is still needed in order to actually transfer the corresponding techniques or technologies from the research lab to the company. Based on the AtlanMod concrete experience with the previously mentioned open source tools/projects, we have extracted a pragmatic approach [37] 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.

6. New Results

6.1. Model Transformation

Model transformation and in particular our ATL model transformation language continues playing a key role in our MDE strategy. During 2011 the new results in this area have been:

- The development of an execution algorithm for the lazy execution of ATL transformations [31]. The increasing adoption of Model-Driven Engineering in industrial contexts highlights scalability as a critical limitation of several MDE tools. When these tools are built around model-to-model (M2M) transformations, the efficiency of the transformation engine risks to become a performance bottleneck for the whole MDE environment. This new execution mode solves this problem by providing on-demand execution of model transformations. The computation required to generate a data element of the target model is only triggered once the user requests to read that data element.
- Towards a general semantics for transformation languages. In the mid-term, we would like to be able to achieve interoperability among existing transformation languages (e.g. to create mixed transformations or simply to compare alternative transformation solutions for the same escenario). In this direction, we have worked on a general composition semantics for rule-based transformation languages [32] and an extensive survey of inheritance support in existing languages [33]. Moreover, to make ATL closer to other transformation languages, in special graph transformations, we have developed a new ATL refining mode [43] that allows executing in-place transformations
- Also relevant, this year AtlanMod has coorganized the main international conference on model transformations [41]

6.2. Model Representation

As part of our work on core techniques for the specification and internal representation of models we would like to highlight:

- Improved model typing. In order to represent metadata more accurately, we have worked on a functional typing system for megamodeling [18]. The basic idea is to consider transformations as functions, and to give them functional types that can facilitate the reasoning on them.
- Virtual EMF [21] for the transparent Composition, Weaving and Linking of Models. Virtual EMF facilitates the global manipulation of an heterogeneous but related set of models by providing the illusion of having a single and unified view of the modeled system.
- EMF Profiles [25], a mechanism to enable users annotate existing models without *polluting* them (i.e. without directly changing their content).
- Batch scripting support for the retrieval and manipulation of models from model repositories thanks to our new language MoScript [24]

6.3. Model Quality

Our work on model quality defends the idea that there is not a silver bullet technique to verify the quality of models and that a combined approach offers the best trade-off. Lightweight techniques can provide a quick feedback even if it is partial and, when necessary, more complete ones can be utilized to complement the results. In this sense, this year we have developed:

- Lightweight techniques for the verification of ATL transformations [27] and UML Executable models [26]
- Proposed a method for the automatic generation of correct dynamic models for a given domain [11]. Basic operations to modify the information the software needs to store and manipulate about the domain are generated. The generation process ensures that all generated operations are strongly executable.
- Created a brand new Eclipse Lab project aimed at facilitating the verification of any kind of (EMF-based) model with the EMFtoCSP tool (see the tool description in the tools section of this report)
- And applied some of these ideas to the verification and testing of web applications [15]

6.4. Domain Specific Language

There is a growing interest in the research community on the definition of a new generation of language workbenches to facilitate the definition of non-trivial domain specific languages. In this area, we are working on the definition of a set of quality properties that will help language designers to validate if their language is well-adapted to the needs of the language users. This is done by mining repositories containing a corpus of examples of the language.

To begin with we have been reusing the notion of software clones [16] [30], a well-known property in the programming domain and see if it is also meaningful for languages at the model level [29].

6.5. Reverse Engineering

We have continued our work on MoDisco, specially wrt the dissemination activities around the platform [19] [40] and the extension of the tooling support mainly through the spin-off of the EMFFacet Eclipse project, explained in the tools section of this report

6.6. Systems interoperability

MDE can be used as an intermediate representation between two different technical spaces / platforms / tools to facilitate their interoperability. During this year, we have followed this approach in the following results:

- Automation of the interactions with APIs by automatically discovering and expressing them as models thanks to our API2MOL approach [14]
- Bridging the business and the technical domains, developing among others the Portolan tool [42] for the cartography of Information Systems.
- MDE itself can benefit from the work and techniques available in a different technical space. In this context we have combined MDE and constraint programming to see how the combination improves the solution of classical problems like the configuration of a set of components/plug-ins [22]. We have even organized a workshop on the topic of merging MDE and logic programming [20] to better understand how they can benefit each other.

7. Contracts and Grants with Industry

7.1. BNP Paribas collaboration, Continuity between the process modeling and software design (2010-2011)

This direct collaborative action with the BNP Paribas company focused on the general study of business process modeling, and more specifically its continuity between the domain (business) and the technological (IT) spaces. Our work during this technological collaboration action was composed of two main parts:

1. A detailed study of the state of the art (in both industry and research worlds), also presenting initial proposals of concrete solutions;
2. An implementation of an extensible prototype, following the comments and recommendations of the previously mentioned study.

This work has been realized in several iterations allowing regular exchanges with BNP Paribas and useful feedback integration.

As a final result, the delivered items have been the following ones:

- The *State-of-the-art* document generally introducing MDE, and presenting existing works in business and IT modeling from both the research and industrial worlds. It also includes the overall specification of a prototype to be implemented as a potential solution for cartography;
- A complementary document presenting the principles and techniques for a model-based bridge between two tools. It also provides an architectural overview of such a bridge considering two real tools from both the business (BPMN) and IT (UML2) sides together with the concrete prototype implementation of such a bridge from BPMN to UML2 Activity Diagrams;
- A model-driven cartography framework called Portolan (http://code.google.com/a/eclipselabs.org/p/portolan/), based on the specification from the first document. It can be used as a general solution for building global views on business and technical spaces of a company.

7.2. MIA-Software research collaboration, Scalability of MDE techniques (2011-2012)

Since several years, AtlandMod and Mia-Software are actively collaborating around the topic of Model Driven Reverse Engineering (MDRE), i.e.; the combined use of different model-based techniques to solve real reverse engineering problems. This has resulted in the successful creation and development of two open source Eclipse projects, namely Eclipse-MDT MoDisco (providing a generic and extensible MDRE framework) and Eclipse-EMFT EMF Facet (providing a dynamic model extension framework), both reaching today an industrial maturity level.

However, for these technologies to be definitely adopted and deployed in the context of very large systems handling huge data volumes, some remaining scalability issues still have to be addressed. Thus, scalability of model-driven techniques is one of the main challenges MDE is facing right now. In this context, AtlanMod has joined forces with MIA-Software as part of an INRIA technology transfer action. This initiative is devoted to the development of new generation MDE techniques, for model creation and general handling, that effectively scale up.

Several of our projects are going to be positively impacted by the results produced during this collaboration, of course the MoDisco and EMF Facet frameworks as mentioned before, but also others such as notably the related ATL model transformation technology. Among the different research challenges behind the MDE scalability and performance improvement, the following ones are being or will be explored in the context of this collaborative action:

- **Model random access.** Advanced use of on-demand lazy loading techniques;

- **Model clustering and slicing.** Advanced use of semantic grouping and partial loading techniques;
- **Model virtualization.** Transparent and on-demand access to different views on a same model;
- **Lazy evaluation of model transformation.** On-demand lazy execution of transformations;
- **Incremental model transformation.** Partial model access and transformation execution;
- **Multi-threaded model transformation.** Parallelization of both model accesses and rule executions.

Officially started recently, this initiative has already opened very promising perspectives for the future in terms of both research and industrial opportunities (e.g.; regarding new national and international project proposals on innovative research topics of common interest).

7.3. WebRatio/University of Trento - User interaction modeling language (2011-2012)

AtlanMod will help WebRatio and the University of Trento in the definition (to be provided as an answer to the corresponding OMG RFP) of a modeling language for designing user interaction flows (not limited to the Web). Such a language should be: Extremely compact (no useless overhead), Effective (allows to model exactly what users want), Efficient (grants high reusability of model fragments), Easy to learn (very low learning curve), Comprehensive (covers most of the user interaction needs), Open and extensible (for covering any ad-hoc logic) and Platform independent (addressing any type of user interface device).

This action is just starting at the moment.

8. Partnerships and Cooperations

8.1. Regional Initiatives

Program: Pays de la Loire regional funding. Call: Soutenir et accompagner la constitution et/ou l'implantation de nouvelles équipes sur des thématiques émergentes

Project title: AtlanMod New Team Creation

Duration: 2011 - 2014

Coordinator: AtlanMod

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.

8.2. National Initiatives

Program: ANR - ARPEGE program

Project acronym: Galaxy

Project title: Galaxy

Duration: 2010 - 2013

Coordinator: Airbus

Other partners: Industry (Airbus), Research and University (Armines -AtlanMod-, IRIT, LIP6) and Vendors and service providers (AKKA, Softeam)

Abstract: GALAXY (http://galaxy.lip6.fr) proposes to deal with the model driven collaborative development of complex systems. Galaxy aims at defining an open and flexible architecture particularly designed to be scalable. One of the key points is related to the fragmentation and distributiveness of huge models, their synchronization and relationship with communication means classically used by development teams. The work is being driven by use cases provided by a company (Airbus), which describe scalability issues they face during systems developments. Our work in this project is composed of two main parts: 1) the conception of efficient mechanisms for multiple views of complex (large) models; 2) the definition of a solution for the automation of modeling tasks on large model repositories, like the execution of large amounts of transformations, the orchestration of their execution, and the effective browsing of repositories for finding specific models. In this context we have developed MoScript, a scripting language (and corresponding execution engine) to write batch processing modeling tasks.

Program: ANR

Project acronym: IDM++

Project title: Ingénierie dirigée par les modèles ++

Duration: 2008 - 2011

Coordinator: IBM (ILog)

Other partners: CEA, Mia-Software, Prima Solution

Abstract: IdM++ (http://www.emn.fr/x-info/idmpp/index.php/Accueil) main goal is to investigate advanced issues in model engineering. The IDM++ consortium proposes the combination of Global Model Management and Model Configuration techniques. The goal is to bring together two different communities: the Model-Driven Engineering (MDE) community and the logic programming community, to explore how each community can benefit from the techniques of the other. We refer to the logic programming community in a broad sense (i.e. including Constraint Logic Programming, Answer-Set Programming but also ontology and semantic web aspects).

This approach is promoted according to the partners background in Model Driven Engineering, Constraint based programming and optimization techniques. The team is particularly in charge of WP 2, on global model management.

8.3. European Initiatives

Program: Artemis

Project acronym: CESAR

Project title:Cost-Efficient methods and processes for SAfety Relevant embedded systems

Duration: 2009 - 2012

Other partners: More than 50 partners

Abstract: The three transportation domains, automotive, aerospace, and rail, as well as the automation domain share the need to develop ultra-reliable embedded systems to meet social demands for increased mobility and safety in a highly competitive global market.To maintain the European leading edge position in the transportation as well as automation market, CESAR aims to boost cost efficiency of embedded systems development and safety and certification processes by an order of magnitude. CESAR pursuits a multi-domain approach integrating large enterprises, suppliers, SME's and vendors of cross sectoral domains and cooperating with leading research organizations and innovative SME's. In particular, we work on the Reference Technology Platform, which aims at tool integration. We propose to achieve tool integration by means of metamodeling and model transformations [46]. In the context of this project we are developing VirtualEMF (http://code.google.com/a/eclipselabs.org/p/virtual-emf/) , an approach and tool for the transparent composition, weaving and linking of heterogeneous models.

Program: ITEA2

Project acronym: OPEES

Project title:Open Platform for the Engineering of Embedded Systems

Duration: 2009 - 2012

Coordinator: Obeo

Other partners: Many other research labs and companies

Abstract:OPEES (http://www.opees.org) mission statement is to settle a community and build the necessary means and enablers to ensure long-term availability of innovative engineering technologies in the domain of dependable or critical software-intensive embedded systems. In particular, within OPEES, our schema of open source industrial collaboration [37] (e.g. around ATL) will be tested and developed as a team contribution to this project. AtlanMod is also responsible for providing a model-driven interoperability solution for the integration of the ecosystem of OPEES components, based on metamodeling the domain data of each component and bridging, by model transformation, the specific data representations.

8.3.1. Collaborations in European Programs, except FP7

Program: Leonardo da Vinci (LifeLong learning programme)

Project acronym: MDEExpertise

Project title:Exchanging knowledge, techniques and experiences around Model Driven Engineering education

Duration: 2010 - 2012

Coordinator: Lublin University of Technology

Other partners: Potecnico di Milano, Universidad de Alicante

Abstract: MDE Expertise (http://www.learnMDE.org) is an European Leonardo da Vinci project focused on the development of common educational materials for the Model Driven Engineering (MDE) area. The main aim of the project is to transfer and adapt the education in Model Driven Engineering concepts to the local IT education societies of the partner's countries, thus improving the partners' knowledge about up to date current software development methods. This results in the best preparation for professionals competing on the IT market. Direct results include: development of common MDE teaching methods, suited for the partners' local needs and market requirements; creation of teaching materials (with online version) localized for the partners' languages and definition of tools for e-learning and knowledge exchange. Indirect effects include improving the capability of local SMEs in solving complex software design problems through modeling, and evolving the software development job market.

8.4. International Initiatives

8.4.1. INRIA International Partners

AtlanMod keeps a close and continuous collaboration with a number of foreign research group. To mention the top 5:

- Vienna University of Technology. Collaboration on model transformations (refactoring, refinement, evolution,...)
- University of Bremen. Collaboration on model validation and verification.
- Technical University of Barcelona. Collaboration on conceptual modeling, semantics of modeling primitives, code-generation and the like.
- Politecnico di Milano. Collaboration on modeling in a web engineering context and on model search problems in repositories.
- University of Toronto. Collaboration on intentional models, requirements engineering and social modeling topics.

8.4.2. Visits of International Scientists

In 2011 three visitors did a research stay with AtlanMod:

- Dennis Wagelaar (Vrije Universiteit Brussel, Belgium), January-February
- Jesus Gallardo (Universidad de Castilla la Mancha, Spain), July
- Jokin Garcia (University of the Basque Country, Spain), May

8.4.2.1. Internship

Víctor García from the technical university of Valencia did his master thesis in AtlanMod, in the context of our MoDisco project.

9. Dissemination

9.1. Animation of the scientific community

9.1.1. Animation of the local scene

In 2009, the AtlanMod team initiated a regular series of events called *Les Jeudis des Modèles* [4]. This event has successfully continued during 2011. Two/three times per year approximately, an international expert from the model engineering community is invited to come and present a subject of research or innovation of interest to the scientific and industrial community. These events typically attract between 30 and 60 researchers, students and industrials. In fact, *Les Jeudis des Modèles* event has become a regional rendez-vous of the model engineering community, attracting people from Rennes, Vannes, La Rochelle, and many other places beyond Nantes. Among others, we have invited Patrick Albert (IBM), Nicolas Rouquette (NASA/JPL), Ed Merks (Macro Modeling), Ivar Jacobson (IJ company), Sridhar Iyengar (IBM) and Jon Whittle (Professor at Lancaster University). It should be noted that these visits allow us to organize different meetings between the researchers of the AtlanMod team and the various industrials attending the main presentation.

Besides this, as part of our commitment to the Eclipse community, we co-organized (together with the Obeo company) an Eclipse DemoCamp event to celebrate the new Eclipse Indigo version release. The Eclipse DemoCamps are an opportunity to showcase all of the technology being built by the Eclipse community and, for the team, an excellent opportunity to meet other Eclipse enthusiasts in the region and show them the Eclipse tools developed by the team.

9.1.2. Organization

In 2011, the AtlanMod team has coorganized the following events:

- J. Cabot has been PC Chair (together with Eelco Visser) of the Int. Conf. on Model Transformation 2011 [5]
- M. Tisi has coorganized the 3rd International Workshop on Model Transformation with ATL [6] (MtATL 2011) co-located with the TOOLS conferences in Zurich.
- J. Cabot has coorganized the OCL and Textual Modeling Workshop (OCL'11) [7] co-located with the TOOLS conference in Zurich.
- J. Cabot has coorganized the MELO 2011 - Model-Driven Engineering, Logic and Optimization: friends or foes? Workshop. Co-located with ECMFA 2011 in Birmingham UK.

[4] http://www.emn.fr/z-info/jmodeles/
[5] http://www.model-transformation.org/ICMT2011/
[6] http://www.emn.fr/z-info/atlanmod/index.php/MtATL2011
[7] http://gres.uoc.edu/OCL2011/

9.1.3. Editorial Bords

Participation to editorial boards of scientific journals:

- Jordi Cabot: Journal of Object Technology

9.1.4. Reviewing of International Journals

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

- Jordi Cabot: IEEE Software, IET Software, IEEE Transactions on Software Engineering Journal, Transactions of the Society for Modeling and Simulation International, Journal of Systems and Software, Information and Software Technology Journal, Software and Systems Modeling Journalm Data and Knowledge Engineering Journal, Science of Computer Programming journal, Journal of Web Engineering journal, International Journal of Software and Informatics, Transactions on Aspect-Oriented Software Development journal, Transactions on Systems, Man, and Cybernetics Journal, Computers and Security Journal, Journal of Computer Science Education
- Robert Tairas: Automated Software Engineering, International Journal of Computers and Applications
- Massimo Tisi: Science of Computer Programming, Software and Systems Modeling, Journal of Web Engineering

9.1.5. Program Committee members

Participation to conference program committees:

- Jordi Cabot:
  - International: World Wide Web Conference, International Conference on Model-Driven Engineering Language and Systems, Software Language Engineering conference, European Conference on modeling foundations and applications, International Conference on Engineering of Complex Computer Systems, International Conference on Web Engineering, Model Driven Web Engineering Workshop, Workshop on Algebraic Methods in Model-Based Software Engineering, Applications of Graph Transformation With Industrial Relevance, Tool Transformation Contest, International Symposium on Rules, RuleML Challenge,Industry Track of Software Language Engineering, Model-Driven Requirements Engineering, International Workshop on the Web and Requirements Engineering.
  - National: French national conference on MDE, Spanish national conference on MDE.
- Robert Tairas:
  - International: ACM Symposium on Applied Computing - Software Engineering Track.
- Massimo Tisi:
  - International: Transformation Tool Contest, Workshop on Domain-Specific Modeling, International Conference on Web Information Systems and Technologies 2012, International Conference on Current Trends in Theory and Practice of Computer Science 2012

9.2. PhD Juries

- Jordi Cabot: Islam Abdelhalim. PhD confirmation report. University of Surrey (UK)

9.3. Research Evaluation Committees

- Jordi Cabot:Reviewer for Spanish CICYT Research Projects Call, Reviewer for the National UK Projects Call, Reviewer for National Argentinian Projects Call

9.4. Involvement in the Open Source Community via Eclipse

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 and EclipseCon Europe), organization of events targeting the local community, 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 M2M ATL (Frédéric Jouault) and MDT MoDisco (Hugo Bruneliere, also including AM3) projects, the Eclipse reference projects concerning respectively model-to-model transformation and model-driven reverse engineering;
- Commitment to other projects directly under *Eclipse.org*: EMFT EMF Facet (Hugo Bruneliere), GMT AMW (Frédéric Jouault);
- Creation and development of new incubation projects under *Eclipse Labs*: EMFToCSP (Jordi Cabot, Carlos Gonzalez) and Virtual EMF (Caue Clasen, Jordi Cabot);
- Organization of an official Eclipse DemoCamp in Nantes, on the 30th of June 2011, for locally promoting the Eclipse Indigo release including the MoDisco, ATL and EMF Facet team's projects (Hugo Bruneliere, in collaboration with the Obeo and Mia-Software companies);
- Publication of a technical paper on MoDisco in the international JavaTech Journal issue dedicated to this same Eclipse Indigo release (Hugo Bruneliere);
- Presentation of talks during the two main Eclipse events (Hugo Bruneliere): at EclipseCon North America 2011 about MoDisco, at EclipseCon Europe 2011 about Virtual EMF.

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. This collaborative approach, allowing the industrialization of our research prototypes using open source and Eclipse as the medium, has also been presented this year to the OW2 open source community (*The open source community for infrastructure software*) during their main annual event OW2Con 2011, as part of the Session *Open Source Innovation Catalyst* (Hugo Bruneliere).

9.4.1. Teaching

During the period 2010/11 the members of the AtlanMod team have been in charge of a 360-hour diploma on MDE [8].

Apart from this, individual members have also had teaching duties in other courses:

- License: J. Cabot is in charge of a 24-hour course on MDE in the CS degree at the Ecole des Mines de Nantes .
- License: M. Tisi. Programmation, 15h, L1, Ecole des Mines de Nantes
- License: M. Tisi. Interface Homme-Machine : conception, 13,75h, L2, Ecole des Mines de Nantes
- License: M. Tisi. Bases de données, 20h, L2, Ecole des Mines de Nantes
- License: M. Tisi. Structures de données, 17,5h, L2, Ecole des Mines de Nantes
- License: M. Tisi. Programmation Impérative, 22,5h, L1, Ecole des Mines de Nantes
- License: Programmation Par Objets, 9h, L1, Ecole des Mines de Nantes

[8] http://www.mines-nantes.fr/fr/Formations/Formation-specialisee/MDE

10. 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, p. 47-67.

[2] J. BÉZIVIN, F. BÜUTTNER, 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, p. 440–453.

[3] J. BÉZIVIN, F. JOUAULT, P. ROSENTHAL, P. VALDURIEZ. *Modeling in the Large and Modeling in the Small*, in "Proceedings of the European MDA Workshops: Foundations and Applications, MDAFA 2003 and MDAFA 2004, LNCS 3599", U. ASSMANN, M. AKSIT, A. RENSINK (editors), Springer-Verlag GmbH, 2005, p. 33–46.

[4] 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, p. 283-302.

[5] J. CABOT, R. PAU, R. RAVENTÓS. *From UML/OCL to SBVR specifications: A challenging transformation*, in "Inf. Syst.", 2010, vol. 35, n^{o} 4, p. 417-440.

[6] 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/.

[7] 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), p. 31–39.

[8] F. JOUAULT, I. KURTEV. *On the Interoperability of Model-to-Model Transformation Languages*, in "Science of Computer Programming", 2007, vol. 68, n^{o} 3, p. 114-137, Special Issue on Model Transformation.

[9] R. LEMESLE. *Transformation rules based on meta-modeling*, in "Proceedings of the Second International Enterprise Distributed Object Computing conference (EDOC'98)", 1998, p. 113–122.

[10] 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, http://www.springerlink.com/content/j212100qq405n5q1/ [*DOI :* 10.1007/S10270-011-0191-2], http://hal.archives-ouvertes.fr/hal-00589277/en/.

Publications of the year

Articles in International Peer-Reviewed Journal

[11] M. ALBERT, J. CABOT, C. GÓMEZ, V. PELECHANO. *Generating operation specifications from UML class diagrams: A model transformation approach*, in "Data & Knowledge Engineering / Data and Knowledge Engineering", April 2011, vol. 70, n^{o} 4, p. 365-389 [*DOI :* 10.1016/J.DATAK.2011.01.003], http://hal.inria.fr/inria-00573717/en.

[12] J. CABOT, R. CLARISÓ, M. GOGOLLA, B. WOLFF. *Preface (OCL 2011 Proceedings).*, in "Electronic Communication of the European Association of Software Science and Technology", October 2011, vol. 44, http://hal.inria.fr/hal-00647006/en.

[13] J. CABOT, M. TISI. *The MDE Diploma: First International Postgraduate Specialization in Model-Driven Engineering*, in "Journal of Computer Science Education", November 2011, vol. 21, n° 4, p. 389-402 [*DOI :* 10.1080/08993408.2011.630131], http://hal.inria.fr/hal-00646898/en.

[14] 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.

[15] P. FRATERNALI, M. TISI. *Using Traceability Links and Higher Order Transformations for Easing Regression Testing of Web Applications*, in "Journal of Web Engineering", 2011, vol. 10, n° 1, p. 1-20, http://hal.inria.fr/inria-00602420/en.

[16] J. PATE, R. TAIRAS, N. KRAFT. *Clone Evolution: A Systematic Review*, in "Journal of Software Maintenance and Evolution: Research and Practice", September 2011, http://hal.inria.fr/inria-00622949/en.

[17] S. ROYCHOUDHURY, J. GRAY, F. JOUAULT. *A Model-Driven Framework for Aspect Weaver Construction*, in "LNCS Transactions on Aspect-Oriented Software Development", June 2011, http://www.springer.com/computer/lncs?SGWID=0-164-2-109318-0, http://hal.inria.fr/hal-00589274/en.

[18] 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° 1 [*DOI :* 10.1007/S10270-011-0191-2], http://www.springerlink.com/content/j212100qq405n5q1/, http://hal.inria.fr/hal-00589277/en.

Articles in Non Peer-Reviewed Journal

[19] H. BRUNELIÈRE. *How to Deal with your IT Legacy? Reverse Engineering using Models - MoDisco in a Nutshell!*, in "JavaTech Journal", August 2011, n° 10, p. 21-24, http://hal.inria.fr/hal-00615269/en.

International Conferences with Proceedings

[20] J. CABOT, P. ALBERT, G. DUPÉ, M. DIDONET DEL FABRO, S. LEE. *MELO 2011 - 1st Workshop on Model-Driven Engineering, Logic and Optimization*, in "ECMFA 2011 - Seventh European Conference on Modelling Foundations and Applications", Birmingham, Royaume-Uni, June 2011, http://hal.inria.fr/inria-00582526/en.

[21] C. CLASEN, F. JOUAULT, J. CABOT. *VirtualEMF: a Model Virtualization Tool*, in "ER 2011 - 30th International Conference on Conceptual Modeling", Brussels, Belgique, November 2011, http://hal.inria.fr/inria-00609613/en.

[22] G. DOUX, P. ALBERT, G. BARBIER, J. CABOT, M. DIDONET DEL FABRO, S. LEE. *An MDE-based approach for solving configuration problems: An application to the Eclipse platform*, in "ECMFA 2011 - Seventh European Conference on Modelling Foundations and Applications", Birmingham, Royaume-Uni, June 2011, http://hal.inria.fr/inria-00582506/en.

[23] S. HURTADO, S. SEN, R. CASALLAS. *Reusing Legacy Software in a Self-adaptive Middleware Framework*, in "Adaptive and Relfective Middleware Workshop, Middleware 2011", Lisbon, Portugal, October 2011, http://hal.inria.fr/hal-00643088/en.

[24] W. Kling, F. Jouault, D. Wagelaar, M. Brambilla, J. Cabot. *MoScript: A DSL for querying and manipulating model repositories*, in "Software Language Engineering (SLE2011)", Braga, Portugal, October 2011, http://hal.inria.fr/inria-00609673/en.

[25] P. Langer, K. Wieland, M. Wimmer, J. Cabot. *From UML Profiles to EMF Profiles and Beyond*, in "TOOLS 2011", Zurich, Suisse, Springer, June 2011, http://hal.inria.fr/inria-00610761/en.

[26] E. Planas, J. Cabot, C. Gómez. *Lightweight Verification of Executable Models*, in "30th International Conference on Conceptual Modeling (ER 2011)", Brussels, Belgique, November 2011, http://hal.inria.fr/inria-00610208/en.

[27] E. Planas, J. Cabot, C. Gómez. *Two Basic Correctness Properties for ATL Transformations: Executability and Coverage*, in "3rd International Workshop on Model Transformation with ATL", Zurich, Suisse, July 2011, http://hal.inria.fr/inria-00610205/en.

[28] A. Randak, S. Martínez Perez, M. Wimmer. *Extending ATL for Native UML Profile Support: An Experience Report 49-62*, in "3rd International Workshop on Model Transformation with ATL (MtATL-2011)", Zürich, Suisse, M. T. Ivan Kurtev, D. Wagelaar (editors), CEUR-WS.org, June 2011, vol. vol-742, p. 49-62, http://hal.inria.fr/inria-00609024/en.

[29] R. Tairas, J. Cabot. *Cloning in DSLs: Experiments with OCL*, in "International Conference on Software Language Engineering", Braga, Portugal, July 2011, http://hal.inria.fr/inria-00619057/en.

[30] R. Tairas, F. Jacob, J. Gray. *Representing Clones in a Localized Manner*, in "International Workshop on Software Clones", Waikiki, États-Unis, May 2011 [*DOI :* 10.1145/1985404.1985414], http://hal.inria.fr/inria-00606911/en.

[31] M. Tisi, S. Martínez Perez, F. Jouault, J. Cabot. *Lazy Execution of Model-to-Model Transformations*, in "Model Driven Engineering Languages and Systems, 14th International Conference, MODELS 2011", Wellington, Nouvelle-Zélande, October 2011, http://hal.inria.fr/inria-00609458/en.

[32] D. Wagelaar, M. Tisi, J. Cabot, F. Jouault. *Towards a General Composition Semantics for Rule-Based Model Transformation*, in "MoDELS", Wellington, Nouvelle-Zélande, 2011, p. 623-637, http://hal.inria.fr/hal-00642066/en.

[33] M. Wimmer, G. Kappel, A. Kusel, W. Retschitzegger, J. Schönböck, W. Schwinger, D. Kolovos, R. F. Paige, M. Lauder, A. Shürr, D. Wagelaar. *A Comparison of Rule Inheritance in Model-to-Model Transformation Languages*, in "ICMT2011 - International Conference on Model Transformation", Zurich, Suisse, June 2011, http://hal.inria.fr/inria-00582465/en.

National Conferences with Proceeding

[34] C. Clasen, F. Jouault, J. Cabot. *Virtual Composition of EMF Models*, in "7èmes Journées sur l'Ingénierie Dirigée par les Modèles (IDM 2011)", Lille, France, July 2011, http://hal.inria.fr/inria-00606374/en.

[35] V. Cosentino, G. Doux, P. Albert, G. Barbier, J. Cabot, M. Didonet Del Fabro, S. Uk-Jin Lee. *Une approche de MDE pour la résolution de problèmes de configuration : Une application à la plate-forme Eclipse*, in "Journées nationales IDM, CAL, et du GDR GPL", Lille, France, June 2011, http://hal.inria.fr/inria-00613338/en.

[36] S. Martínez Perez, M. Wimmer, F. Jouault, J. Cabot. *Catálogo de refactorizaciones para transformaciones Modelo a Modelo*, in "XVI Jornadas de Ingeniería del Software y Bases de Datos (JISBD)", A Coruña, Espagne, September 2011, http://hal.inria.fr/inria-00609451/en.

Conferences without Proceedings

[37] 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.

[38] H. Brunelière. *MoDisco & ATL*, in "Eclipse DemoCamp Indigo 2011 - Nantes", Nantes, France, June 2011, Lien vers la page officielle de l'évènement (slides du talk) : http://wiki.eclipse.org/Eclipse_DemoCamps_Indigo_2011/Nantes#Age http://hal.inria.fr/hal-00610556/en.

[39] H. Brunelière, G. Dupé. *Virtual EMF - Transparent Composition, Weaving and Linking of Models*, in "EclipseCon Europe 2011", Ludwigsburg, Allemagne, November 2011, Lien vers la page officielle de la conférence (slides du talk) : http://www.eclipsecon.org/europe2011/sessions/virtual-emf-%E2%80%93-transparent-composition-weaving-and-linking-models, http://hal.inria.fr/hal-00639555/en.

[40] F. Madiot, H. Brunelière. *Spy On Your Models*, in "EclipseCon 2011", Santa Clara, Californie, États-Unis, March 2011, Lien vers la page officielle de la conférence (slides du talk) : http://www.eclipsecon.org/2011/sessions/?page=sessions&id=2093, http://hal.inria.fr/hal-00610552/en.

Scientific Books (or Scientific Book chapters)

[41] J. Cabot, E. Visser. *Theory and Practice of Model Transformations - 4th International Conference, ICMT 2011, Zurich, Switzerland, June 27-28, 2011. Proceedings*, Springer, July 2011 [*DOI :* 10.1007/978-3-642-21732-6], http://hal.inria.fr/inria-00610760/en.

Research Reports

[42] V. Mahe, S. Martínez Perez, G. Doux, H. Brunelière, J. Cabot. *PORTOLAN: a Model-Driven Cartography Framework*, INRIA, February 2011, n^{o} RR-7542, http://hal.inria.fr/inria-00568186/en.

[43] M. Tisi, S. Martínez Perez, F. Jouault, J. Cabot. *Refining Models with Rule-based Model Transformations*, INRIA, March 2011, n^{o} RR-7582, http://hal.inria.fr/inria-00580033/en.

References in notes

[44] 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, p. 277–286.

[45] H. Brunelière, J. Cabot, C. Clasen, F. Jouault, J. Bézivin. *Towards Model Driven Tool Interoperability: Bridging Eclipse and Microsoft Modeling Tools*, in "6th European Conference on Modeling Foundations and Applications (ECMFA 2010)", France Paris, Jun 2010, p. 32-47, http://hal.inria.fr/hal-00534373/en.

[46] H. BRUNELIÈRE, J. CABOT, C. CLASEN, F. JOUAULT, J. BÉZIVIN. *Towards Model Driven Tool Interoperability: Bridging Eclipse and Microsoft Modeling Tools*, in "ECMFA", T. KÜHNE, B. SELIC, M.-P. GERVAIS, F. TERRIER (editors), Lecture Notes in Computer Science, Springer, 2010, vol. 6138, p. 32-47.

[47] 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, p. 350–354.

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

[49] 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, p. 273-280.

[50] 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.

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

[52] J. CABOT, R. CLARISÓ, D. RIERA. *UMLtoCSP: a tool for the formal verification of UML/OCL models using constraint programming*, in "ASE", R. E. K. STIREWALT, A. EGYED, B. FISCHER (editors), ACM, 2007, p. 547-548.

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

[54] 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.

[55] M. DIDONET DEL FABRO, P. VALDURIEZ. *Towards the efficient development of model transformations using model weaving and matching transformations*, in "Software and System Modeling", 2009, vol. 8, n^{o} 3, p. 305-324.

[56] 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, p. 171–185.

[57] F. JOUAULT, J. BÉZIVIN, I. KURTEV. *TCS: a DSL for the Specification of Textual Concrete Syntaxes in Model Engineering*, in "GPCE'06: Proceedings of the fifth international conference on Generative programming and Component Engineering, Portland, Oregon, USA", 2006, p. 249–254.

[58] 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, p. 602–616.