Modeling Method Conceptualization within OMiLAB: The Secure Tropos Case

Dominik Bork, Michalis Pavlidis, Wilfrid Utz

Published in:
IEEE Eleventh International Conference on Research Challenges in Information Science 2017, Brighton, UK
Modeling Method Conceptualization within OMiLAB: The Secure Tropos Case

Dominik Bork  
Faculty of Computer Science  
Research Group Knowledge Engineering  
University of Vienna  
Waehringer Street 29, 1090 Vienna, Austria  
Email: dominik.bork@univie.ac.at

Michalis Pavlidis  
School of Computing  
Engineering, and Mathematics  
University of Brighton  
Lewes Road, Brighton, UK  
Email: m.pavlidis@brighton.ac.uk

Wilfrid Utz  
Faculty of Computer Science  
Research Group Knowledge Engineering  
University of Vienna  
Waehringer Straße 29, 1090 Vienna, Austria  
Email: wilfrid.utz@univie.ac.at

Abstract—Development of domain specific modeling languages is observed by a growing number of groups emphasizing the implementation of individual modeling languages, methods and approaches for a variety of application domains. Domain orientation allows to express focused models using tailored domain concepts. To raise benefits of domain-specific modeling and in particular the use of developed models, tool support must not be limited to model editors realizing a certain language, but instead must provide full-fledged functionality for domain-specific modeling methods. This paper introduces the Open Models Laboratory (OMiLAB, www.omilab.org), an open environment for method engineering and tool development. In particular, the paper reports on the Secure Tropos method engineering case in OMiLAB, realized using the ADOxx meta-modeling platform as an implementation environment. Secure Tropos is a security-aware software systems development methodology, which combines requirements engineering concepts with security engineering concepts under an unified process to support the analysis and development of secure and trustworthy software systems.

Index Terms—Domain-specific Conceptual Modeling; Meta modeling; OMiLAB; Secure Tropos

I. INTRODUCTION

Digital transformation and disruptive technologies like Internet of Things (IoT) and Cyber-physical Systems (CPS) enable innovative product service systems that will change the way humans use, interact, and employ information technology enormously. These changes also come with challenging requirements for enterprises when thinking of the design and development of secure systems.

Security is nowadays of paramount importance. Enterprises that process sensitive data need to make sure that their systems are secure and prepared to manage possible threats and attacks. It is therefore expected that software system developers are able to develop and deploy secure systems. However, software developers are not necessarily also security experts. It is therefore advisable to tackle the complexity of designing and developing secure software systems by means of a conceptual modeling method.

Conventional and de-facto industry standard modeling languages like UML or BPMN can neither cover such domain-specific requirements adequately nor in an agile manner. It is therefore one recent focus of information science research to explore novel domain-specific conceptual modeling methods and to develop modeling tools that enable efficient application of these methods [1], [2]. Simultaneously can be recognized, that an increasing amount of interest is put on openness and information technology.

The paper at hand bridges those movements by proposing the Secure Tropos modeling method, realized within the Open Models Laboratory (OMiLAB, www.omilab.org). OMiLAB is an open innovation environment for modeling method conceptualization. Over 40 domain-specific modeling methods have already been successfully realized within the OMiLAB. The considered domains are completely divers, comprising e.g., enterprise modeling [3], product service systems for the Internet of Things [4], semantic-based modeling for information systems [5], or creative services in Japanese sushi restaurants [6], [7] provides an overview of existing methods and tools. OMiLAB employs the open use meta modeling platform ADOxx [8] (www.adoxx.org) as a technological foundation for tool development. Feasibility of the OMiLAB in designing domain-specific modeling methods, and of ADOxx in realizing modeling tools, is demonstrated in this paper by a comprehensive case study, hereby introducing the Secure Tropos modeling method and tool (a comprehensive description can be found in [9]).

The paper at hand is structured as follows: Section II provides a brief introduction to the Open Models Laboratory. Section III then provides a comprehensive description on how ADOxx can be utilized to develop modeling tools. The realization of the Secure Tropos modeling method within OMiLAB and the development of the corresponding modeling tool with ADOxx is then focused in Section IV. Finally, some conclusions are drawn and future research is described in Section V.

II. THE OPEN MODELS LABORATORY (OMiLAB)

The Open Models Laboratory (OMiLAB, www.omilab.org) strives for establishing an environment for an open community that enables prototyping, experimentation, and practice-oriented information science research related to method engineering. OMiLAB targets at cross-domain modeling enthusiasts, aiming to address upcoming needs by designing
domain-specific conceptual modeling methods. The goal is to act as a facilitator similarly to Wikipedia [10] for common knowledge creation or platforms like Apple iOS and Google Android for mobile apps [11]. Modeling methods within OMiLAB are realized in a conceptualization lifecycle [12] comprising multiple phases, spanning from the design of a modeling language to the deployment of a full-fledge modeling tool. Following the definition of [13], OMiLAB focuses on conceptual modeling methods that foster understanding of human beings by applying abstraction.

The OMiLAB community shares experience, discusses novel ideas, proposes approaches, and realizes prototypical solutions related to current and emerging challenges. Research results - in the case of OMiLAB, modeling methods, tools and models - are made openly available to all who are interested. First results of the OMiLAB are documented in the recently published Springer book on domain-specific conceptual modeling [7]. Each chapter of the book is accompanied by a freely available modeling tool realized within the OMiLAB.

The OMiLAB method engineering environment builds upon three pillars: the Innovation Environment, the Collaborative Environment, and the Technological Environment. These pillars support method engineers comprehensively by means of an open community, open content, and open technology. In the following, we will concentrate on the collaborative environment of OMiLAB. Moreover, one major benefit of the technological environment, the open meta modeling platform ADOxx [8], will be introduced in Section III.

A. OMiLAB Collaborative Environment

Collaboration of practice-oriented IS researchers and practitioners within OMiLAB can be conducted at a plethora of activities like scientific workshops and tutorials at national and international conferences or in trainings in modeling method engineering and tool development. Two physical OMiLAB’s exist that enable also physical interaction, hereby fostering collaboration. One is located at the University of Vienna, Austria (http://austria.omilab.org/) and one at the University of Chonbuk, Korea (http://asia.omilab.org/). Both OMiLAB’s aim at becoming the primary address for method engineers and tool developers in their respective region.

We plan to further extent the OMiLAB network as such facilities, distributed around the world, enable local bonding whilst ensuring international collaboration and visibility. Organizers of an OMiLAB can approach local contacts in an appropriate tone and even with the same language. This fosters collaboration and bridges entry barriers.

The international distribution of OMiLAB contributors is documented in a wide variety of free available modeling tools and models created with them. The tools are developed by research groups distributed around the world. Conceptualization and development is solely their responsibility. OMiLAB only acts as a facilitator by providing the tools and knowledge to enable those interested to realize their own ideas. A major pillar of the collaborative environment and a meeting point for the OMiLAB community is the Next-generation Enterprise Modeling Summer School (NEMO), which will be detailed in the following.

1) The Next-generation Enterprise Modeling Summer School: The yearly Next-generation Enterprise Modeling Summer School (NEMO, http://nemo.omilab.org/), funded through the Erasmus+ Strategic Partnership Project OMI-KA2, is one vital aspect of the OMiLAB collaborative environment. Since 2014, up to 50 Professors from different countries having varying domain backgrounds present their view on enterprise modeling. Every year, up to 70 international postgraduate and PhD students are enrolled.

NEMO comprises theoretical lectures of the Professors and practical exercise sessions. The exercises require students to think out of the box. Their theoretical knowledge needs to be transferred towards practical application in emerging and challenging domains like Smart Cities [14], [15]. Moreover, students apply open OMiLAB modeling tools to solve practical exercises aligned to the theoretical lectures. This is only possible, as several lecturers of the Summer School also developed an open modeling tool within the OMiLAB. NEMO enhances the community with the next-generation of practice-oriented IS researchers and initiates new collaborations, ultimately leading to the creation of new open content and open modeling tools. NEMO enables the community, albeit being distributed around the world, to have a yearly meeting point to physically interact, share experience, and discuss future collaboration possibilities.

For the students, NEMO provides the opportunity to interact with students coming from completely different cultural and educational backgrounds. Moreover, NEMO students are encouraged to get in contact and discuss with the Professors. All Professors are therefore asked to stay at the location for a couple of days. The packed two week’s program is beneficial for broadening the spectrum and enables reflecting on different perspectives on certain topics which would not be possible at regular university courses.

2) OMiLAB Trainings: OMiLAB recently works on establishing a dedicated training area. The trainings target two user groups: On the one hand, method engineers shall be provided with training material on theoretical foundations and practical applications of designing a domain-specific modeling method. This comprises aspects on the modeling language, e.g., how to design meaningful and intuitive graphical visualizations (cf. [26, 28]), how to specify type and inherent semantics [17], or how to specify mechanisms & algorithms that process the knowledge that is codified in the models. On the other hand, OMiLAB aims to establish a training concept and create supporting material for tool developers. This comprises information on how to realize a modeling method specification with the ADOxx meta modeling platform. The experience gained through conducting multiple physical trainings in the OMiLAB Vienna will help in providing high quality material that will also enable people to study without any physical interaction.
III. THE ADOXX META MODELING PLATFORM

Within OMiLAB, the ADOxx platform [8] is used as a development and configuration environment of modeling methods to realize full-fledged modeling tools. The platform allows an efficient proof-of-concept implementation of novel and innovative modeling concepts following a meta modeling approach. Method engineers are enabled to build upon vast pre-existing functionality on platform level that is provided without any additional implementation effort (e.g., persistence, graphical and tabular model editors, query engine, management functionality for content, users and authentication/authorization). In addition, an extensive collection of configuration options to adapt generic functionality to domain-specific requirements of a specific modeling method is available (e.g., configurable simulation algorithms, publishing and reporting mechanisms). Lastly, arbitrary and customized model processing techniques can be realized by utilizing the stable scripting language AdoScript which is accessible via a comprehensive API.

![Fig. 1. Generic Modelling Method Specification Framework [16]](image)

In the following, the modeling method engineering environment is briefly introduced focusing on the development and deployment phases as defined in Agile Modeling Method Engineering [17] (AMME) lifecycle. The section concludes by introducing the support community on ADOxx.org.

A. Modeling Method Engineering in ADOxx

The modeling method engineering approach in ADOxx builds on the theoretical baseline of the “Generic Modelling Method Specification Framework” introduced in [16]. The framework acts as a thinking paradigm for specification and implementation of full-fledged modeling tools, graphically shown in Fig. 1. The notion of a modeling method is understood as a composition of a) modeling language artifacts (model types, classes, relations, attributes/properties) defining the syntax, semantics and notation during the implementation process, b) model processing algorithms and mechanisms on a generic, hybrid and specific level, and c) a modeling procedure defined as guidelines how language artifacts and processing functionality is applied.

Each of the above components is supported by the platform through programming and configuration languages. The GraphRep language is used for implementing the notation, Expressions for dynamic attributes and properties, and AdoScript is used for the realization of domain-specific algorithms and mechanisms. All languages and related development support tools are integrated in the common ADOxx Development Environment. The resulting configuration and implementation is provided as an application library represented in ADOxx Library Language (ALL) format.

B. Deployment of Modeling Tools

A core feature of ADOxx relates to release and deployment support. In contrast to other platforms, the realized modeling method can be released and deployed as a professional tool to users. Deployment services in ADOxx are responsible for verification of implementation results, personalizing the tool appearance (splash screens, icons, naming), adding documentation and example models, and finally packaging the tool as a ready-to-distribute installation archive. This deployment service enables developers to release their proof-of-concept implementations to the community of domain-specific modelers without further implementation effort. Moreover, developers are exempted from gaining and maintaining knowledge on how to create installation archives on evolving operating systems.

C. ADOxx.org Community

The ADOxx.org platform acts as the community space for modeling method engineers using the platform. Since its release it has attracted around 1500 developers (as of April 2017) from around the globe interacting and sharing results in virtual development spaces and a moderated frequently asked question section. In addition to community spaces, ADOxx.org hosts relevant development resources and services to enable a quick and efficient learning of platform concepts and functionality.

IV. CONCEPTUALIZATION OF SECURE TROPOS

The Secure Tropos methodology [18] is based on the principle that security should be analyzed and considered from the early stages of the software system development process and not added as an afterthought. As such, the methodology provides a modeling language, a security-aware process and a set of algorithms to support the analysis and consideration of security from the early stages of the development process.

A. Realization of Secure Tropos with ADOxx

The Secure Tropos language consists of a set of concepts from the requirements engineering domain, such as actor, goal, and plan, enriched with concepts from security engineering such as security constraint, security objective, vulnerability, threat, and attack method. Also, the methodology consists of a number of relationships between the concepts such as dependency, restricts, satisfies, implements, and attacks. In Figure 2, the Secure Tropos meta model is depicted that includes all the concepts and relationships of the methodology. The white classes show the concepts of the methodology while the grey classes show the relationships between the concepts.

Furthermore, Secure Tropos contains analysis methods that check whether the security constraints under development are satisfied and whether the security threats have been mitigated. In addition, if the system under development is planned to be a
cloud-based system, Secure Tropos has an analysis method to identify the Cloud provider that satisfies most of the identified security mechanisms.

A Secure Tropos meta model already existed along with a description of the classes and the relationships between the classes. Therefore, these had to be structured according to the Generic Modeling Method Specification Framework in order to be supported by the platform through programming and configuration languages. Table I shows the concepts of security constraint, security mechanism, threat, and attacks link of Secure Tropos implemented in ADOxx along with their notation and syntax of the classes by means of attributes.

B. Illustration of Secure Tropos realization with ADOxx

The Secure Tropos methodology has a modeling procedure where at each phase of the procedure a specific diagram is constructed. The diagrams are included in the five views of the methodology: organisational view; security requirements view; security components view; security attacks view; and cloud analysis view. Figures 3, 4, and 5 show examples of an organisational view, security requirements view, security attacks view, respectively. The system under development is a Virtual Learning Environment system for a higher education organisation, such as an university. It’s intention is to support organisational processes and to enhance student learning experience.

In the organisational view the organisation architecture is modeled that includes the various entities that exist in the environment of the system along with their dependencies and security constraints.

In the security requirements view the focus is on the system under development where the software engineer is analysing the security constraints by identifying relevant security objectives and mechanisms.

In the security attacks view the threats are refined by modeling attackers and ways to mitigate attacks on vulnerabilities of the system.

V. Conclusion

Changing environments force enterprises to adjust their enterprise systems and computer systems. One focus needs to be put on the security of such systems. One approach, particularly designed to cover such domain-specific requirements on security is Secure Tropos.
The paper at hand described the conceptualization of the Secure Tropos modeling method within the Open Models Laboratory (OMiLAB, www.omilab.org). First, the three pillars of the OMiLAB environment, the collaborative environment, the innovation environment, and the technological environment were introduced. Afterwards, the ADOxx meta modeling platform has been introduced. ADOxx enables the efficient development of modeling tools following a meta modeling approach. Consequently, the realization of the Secure Tropos
modeling tool on ADOxx was described. The paper then showed, how the tool can be used in a comprehensive case study.

Future research will concentrate on using the Secure Tropos tool in university teaching in order to evaluate its practicability. Moreover, the feedback gained from the users will result in further requirements that can be realized in new versions of the tool. The tool can be downloaded free of charge from the corresponding Secure Tropos project page within the OMiLAB portal (http://austria.omilab.org/psm/content/sectro/info).

REFERENCES


