Title : Model repository

Thesis :
Repositories of modeling artifacts have recently gained increasing attention for the enforcement of reuse in software engineering[3,4]. Repository-centric development processes are adopted in software/system development, such as architecture- or pattern-centric development processes. The proposed framework for building a reuse repository is based on metamodeling techniques that enable the specification of the repository structure and interfaces on content in the form of modeling artifacts and model transformation techniques for the purpose of generation [1,2]. We begin by specifying a conceptual model of the desired reuse repository and proceed by designing modeling languages that are appropriate for the content. The results of these efforts are used to specify and build the repository. The next step is devoted to populating the repository by defining appropriate modeling artifacts. We have proposed an operational architecture for the implementation of a reuse repository. In addition, the tool suite promotes the separation of concerns during the development process by distinguishing the roles of different stakeholders. Access to the repository is customized regarding the development phases, the stakeholder's domain and his system knowledge. The approach for developing a repository of models is generic; however, the discussion and implementation in the context of our experiment is based on the Eclipse platform, Ecore and a CDO-based repository. We have implemented a prototype named SemcoMDT to support the approach as an Eclipse plug-in.

During the implementation and deployment experience, we encounter several technical problems and limitations due to the CDO-based implementation. We plan to study the usage of other infrastructures to support a reuse model repository and their integration with other MDE tools, primarily the access tool, with regard to the development phases and the stakeholders' domain and system knowledge. Additionally, more sophisticated techniques to derive artifacts relationships can be implemented, possibly using different domains, to reduce the complexity to design systems of modeling artifacts. We seek new technologies combined with modeling. Therefore, we will investigate implementation based on an FCA (Formal Concept Analysis) [5]. Each level of abstraction is structured with a lattice, and each lattice is linked. These lattices provide the architect or developer with intelligible classifications for model features. Thus, they enable the search of a model to be indexed, which verifies certain types of properties at a desired level of abstraction. This approach has three advantages: (1) Usability: possibility to dynamically and easily insert a new model; extracting a model can be simply and rapidly realized; (2) Implementation: the space that is required to implement this structure can be optimized; and (3) Visualization: the obtained lattices can be used not only as a component index to ease a search but also as a way for visualizing the content of a reuse repository using a graphical interface.

Bibliography :

[1] Hamid, B., 2016b. A Model Repository Description Language - MRDL. In: International Conference on Software Reuse (ICSR). Vol. 9679 of LNCS. Springer.

[2] Hamid, B., 2016a. A Model-Driven Approach for Developing a Model Repository: Methodology and Tool Support. Future Generation Computer Systems, Elsevier.

[3] Burégio, V., de Almeida, E., Ludrédio, D., Meira, S., 2008. A Reuse Repository System: From Specification to Deployment. In: Mei, H. (Ed.), High Confidence Software Reuse in Large Systems. Vol. 5030 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 88–99.

[4] France, R. B., Bieman, J. M., Cheng, B. H. C., 2006. Repository for Model Driven Development (ReMoDD). In: MoDELS Workshops'06. pp. 311–317.

[5] Ganter, B., Stumme, G., Wille, R., 2005. Formal Concept Analysis. Foundations and Applications. Springer.

To apply, please send your CV at: hamid at irit dot fr

Remarques : The internship will be remunerated. It can be extended to a PhD grant on the same project.

Title : Interplay of security and software architecture

Thesis :

In a recent paper [1], we proposed pattern and properties specification languages to support the pattern-based development of secure software systems. Providing clear, precise, correct and implementable pattern specification is not enough for using a pattern in an automatic way in software engineering development processes. Patterns usually are used in the form of collections of patterns. The pattern of each system and the interaction of patterns within each system are usually well captured and managed. The problem is unpredictable interactions between pattern systems or even between individual patterns. Every system has its particular set of patterns and a much bigger set of unspecified pattern interaction potentials and therefore pattern interaction errors. The interplay between requirements engineering and architecting has been well established but we lack methods and tools to support it. There has been a renewed interest in how to support the Twin Peaks model [2] in a wide range of aspects, such as theoretical frameworks for relating requirements and architecture, tools and techniques such as goal-oriented inference and uncertainty management, problem frames and service composition. There are also approaches for applying the Twin Peaks model in the context of security [3]. There has also been a discussion of the similarities between the problem and solution space and the way of interpreting requirements and design decisions based on the viewpoint of a stakeholder [4].

The goal of this work is to improve this research by investigating more concepts and more semantics to define a new formal modeling paradigm for compositional security within a pattern-based approach as a foundation for novel security engineering practices. We will use concepts such as tactics [5], which have been applied for architectural patterns but not yet to architecture/security composition and integration.

Bibliography :

[1] Hamid, B., Gürgens, S., Fuchs, A., 2016. Security patterns modeling and formalization for pattern-based development of secure software systems. Innovations in Systems and Software Engineering, Springer 12 (2), 109–140.

[2] Avgeriou, P., Grundy, J., Hall, J. G., Lago, P., Mistrík, I. (Eds.), 2011. Relating Software Requirements and Architectures. Springer.

[3] Heyman, T., Yskout, K., Scandariato, R., Schmidt, H., Yu, Y., 2011. The security twin peaks. Proceedings of the International Symposium on Engineering Secure Software and Systems (ESSoS). Vol. LNCS 6542 of Lecture Notes in Computer Science. Springer, pp. 167–180.

[4] Schmidt, H., Jürjens, J., 2011. Connecting Security Requirements Analysis and Secure Design Using Patterns and UMLsec. In: 23rd International Conference on Advanced Information Systems Engineering (CAiSE). Vol. 6741 of LNCS. Springer, pp. 367–382.

[5] Bass, L., Clements, P., Kazman, R., 2013. Software Architecture in Practice (3rd Edition). Addison Wesley.

To apply, please send your CV at: hamid at irit dot fr

Remarques : The internship will be remunerated. It can be extended to a PhD grant on the same project.

Title : Interplay of security and safety

Thesis :

In system engineering, security and safety may be compromised in several system layers and life cycle stages. Usually, security and safety are considered when design decisions are made, leading to potential conflict. This brings tremendous challenges during system design and integration. Just consider CPS and the added complexity and connectivity they offer. For example, the security of cars has already been compromised with the possibility to interact with different safety-related functionality, like releasing the brakes while driving. Thus, security and safety in CPS can only be addressed holistically.

The ideas of system architecture, safety and security modeling and analysis are not new [1], but to the best of our knowledge, the interplay and integration of system security, safety and the rest of the architecture has not been well addressed. In this work, we will focus particularly on the interplay between safety, security and the system architecture; we aim at providing methodological and tool support for their design in unison.

The patterns that are at the heart of our system and software engineering process reflect security and safety solutions at several levels of abstractions (e.g., different systems engineering life-cycle stages, different architecture layers). In our vision, a security or safety pattern [2,3] is a subsystem exposing pattern functionalities through interfaces and solutions behavior and targeting security and safety properties. In this project, we propose a preliminary modeling framework of security and safety properties of design patterns and some of their interplay primitives. The proposed interplay specification makes an attempt to model the resulting effect between security and safety attributes of two interacting patterns. The interplay specification structure can capture the results of combined security and safety specifications of two participating patterns in an interaction. The targeted security and safety modeling syntax will provide a simple formalism for specifying the security and safety properties of individual patterns on which the interplay relationship among patterns can be established. At the core of the framework is a set of Domain Specific Modeling Languages (DSML) and model transformations. Emphasis will be placed on formally defining abstract and concrete syntaxes, as well as the semantics of the modeling languages, e.g., by translation to existing formal languages. This will enable us to verify models using formal analysis.

Bibliography :

[1] Kriaa, S., Pietre-Cambacedes, L., Bouissou, M., Halgand, Y., 2015. A survey of approaches combining safety and security for industrial control systems. Reliability Engineering System Safety 139, 156–178.

[2] Hamid, B., Gürgens, S., Fuchs, A., 2016. Security patterns modeling and formalization for pattern-based development of secure software systems. Innovations in Systems and Software Engineering, Springer 12 (2), 109–140.

[3] Schmidt, H., Jürjens, J., 2011. Connecting Security Requirements Analysis and Secure Design Using Patterns and UMLsec. In: 23rd International Conference on Advanced Information Systems Engineering (CAiSE). Vol. 6741 of LNCS. Springer, pp. 367–382.

To apply, please send your CV at: hamid at irit dot fr

Remarques : The internship will be remunerated. It can be extended to a PhD grant on the same project.