Call for Papers:

Special Issue of Transactions on Aspect-Oriented Software Development
A Common Case Study for Aspect-Oriented Modeling Approaches

Special issue editor: Jörg Kienzle, McGill University, Montreal, Canada

Motivation

The idea behind Aspect-oriented Modeling (AOM) is to apply aspect-oriented techniques to (software) models with the aim of modularizing crosscutting concerns. This can be done within different modeling notations, at different levels of abstraction, and at different moments during the software development process. To date, there exist many AOM techniques for different modeling notations (e.g., for class diagrams, sequence diagrams, state diagrams, protocol machines, component diagrams, live sequence charts, use cases, etc..) and many AOM approaches that can be applied at different phases of software development (requirements, analysis, architecture, design, implementation) or to different domains (e.g., to software product lines, modeling security, modeling fault tolerance, etc...).

Despite the many sources of information on AOM, it is not easy for people who want to use AOM to chose an appropriate AOM technique. Comparing different AOM approaches with each other to highlight the advantages and disadvantages of each one is not trivial, because each research team applies their AOM approach within a different context. Finally, it is not clear nowadays how several AOM approaches that apply to different phases of software development could be combined to produce a coherent *aspect-oriented software development process.*

This special issue solicits papers that demonstrate the power of (existing) AOM techniques and approaches *by applying them to the same case study.* Each paper should critically analyze the strengths and weaknesses of the described approach. The resulting collection of papers will provide a valuable resource for many researchers:

- Researchers who want to learn about AOM will find a concise collection of descriptions of solid and mature AOM approaches. They will only have to understand one case study in order to appreciate the sample models shown in every paper.

- Researchers who want to apply AOM for a particular purpose and are looking for the most appropriate AOM technique will be able to identify the most promising approach(es) easily. Identifying similarities between their problem and the case study will help them to determine candidate approaches.

- AOM experts can readily identify approaches that were able to handle concerns that their own approach can not handle elegantly. This stimulates cross-fertilization between approaches and collaborative research, and brings the AOM community closer to the definition of an aspect-oriented software development process that covers all software development phases.

The Case Study

The broad domain of the case study is *crisis management systems*, i.e. software that facilitate coordination of activities and information flow between all stakeholders and parties that need to work together to handle a crisis. Crisis management systems can be used to handle many types of crises (e.g., natural disasters, epidemics, accidents, attacks, etc...) and may have to interface and interoperate with different types of external services (e.g., military systems, police systems, government, medical services, etc...). Crisis management systems also exhibit many non-functional properties, e.g. they must be highly available, dependable and secure.

The detailed description of the case study can be downloaded at this webpage: http://www.cs.mcgill.ca/~joerg/taosd/ . The document contains the following non-aspect-oriented models:

- Textual requirements describing the general functional and non-functional requirements of crisis management systems

- Feature diagrams describing many variations of crisis management systems

- Textual requirements describing the functionality of a car crash crisis management system

- A use case model describing some of the use cases of the car crash crisis management system

- A domain model of the car crash crisis management system

- An informal description of a possible architecture for the car crash crisis management system

- Some detailed design models for the car crash crisis management system backend

Submission Guidelines and Process

The preface of the special issue will describe the case study in detail, and provide some of the (non-aspect-oriented) requirement, architecture, and design models of the case study document, and refer the interested reader to the TAOSD webpage where he can download the complete description. Therefore, submitted papers do not have to describe the case study again, but only focus the reader to the particular part of the case study that is of interest for the illustration of the AOM approach.

AOM approaches that apply to early phases of software development should either work with the general crisis management system requirements, or use the more specific car crash crisis management system requirements. If the AOM approach you want to present applies to a late software development phase such as design, we request that you apply your approach to the car crash crisis management system backend / server.

You are *not allowed* to add new functional or non-functional requirements to the case study. On the other hand, you *are allowed* to correct errors or clarify ambiguities in the document and the models provided you justify the need to do so.

The first part of the paper should describe in detail how the AOM approach can be applied to the case study, at what phase(s) of the software development process the approach is useful, and show sample models obtained by applying the AOM approach to the case study. The idea in this first part is *not* to write a background section that presents the approach in general, followed by a presentation of the models created for the case study. For a general, detailed overview of your approach the reader should be referred to one of the previously published research papers on the approach. In this part of the paper, the ideas of the approach should be introduced and described using example models of the case study.

The second part of the paper should contain an in-depth analysis of the obtained result, describing what concerns were successfully modularized, and what concerns have proven to be difficult to modularize.

Authors should use a significant portion of the case study to demonstrate the practicality of their approach against one or more of the following criteria: scalability (size of individual models and composition complexity), usability, correctness, reuse, evolution, support for variability, expressivity of composition, detecting and handling of interactions between aspects, traceability, testability, and cost (speed of development, training, money). As a guideline, we suggest to build models that contain on the order of 10 non-orthogonal aspects. A complete set of these models does not have to be presented in the paper, but must be made available on a clearly cited webpage.

Please note that the evaluation criteria for this special issue are slightly different than usual. It is not important that the paper contain new research contributions to the AOM approach that it is describing. Each paper is evaluated on how well it describes the application of the AOM technique to the crisis management system case study, the appropriate number of created models, and on the quality of the analysis of advantages and disadvantages of the AOM approach.

Manuscripts should be formatted using the LNCS formatting guidelines, and should be submitted in *pdf format* (or zipped postscript) to `Joerg.Kienzle@mcgill.ca`. Each submission will be reviewed by at least three reviewers.

Important Dates

Submission deadline: August 31st 2009
First Round Review Notification: October 31st 2009
Re-Submission Revised Papers: December 20th 2009
Second Round Review Notifications: February 15th 2010
Submission of Camera Ready Copy: March 15th 2010
Projected Publication Date of Special Issue: mid 2010