Workshop

Aspect-Oriented Development:
Analysis of Security and Performance

When: Wednesday, May 17, 9:30 am - 4:00 pm
Where: Carleton University, Minto Centre, Faculty Board Room 2014

IMORTANT: The attendance is free, but if you plan to attend
please reply to Dorina C. Petriu <petriu@sce.carleton.ca> a.s.a.p.
We need to estimate the number of attendees so that we can order enough food for lunch.

PROGRAM:

9:30: Dr. Robert France, Colorado State University, http://www.cs.colostate.edu/~france/
The AOMDF: Supporting Model-Driven Development of Dependable Systems

Taming the complexity of developing highly-dependable software systems is the focus of much research in the software engineering community. In particular, research on model-driven development (MDD) focuses on raising the level of abstraction at which systems are developed through the use of models. In this talk I present an MDD framework that uses an aspect-oriented modeling technique to support the specification, composition and transformation of pervasive dependability features. The Aspect-Oriented Model-Driven Development Framework (AOMDF) is currently being developed by researchers at CSU and SINTEF, Norway.
==========

10:30: Dr. Geri Georg, Colorado State University
Aspect-Oriented Risk-Driven Development (AORDD) Framework

This talk will present a background of the AORDD framework, its current status, and future direction. The background includes the goals of the framework and fundamental concepts. The framework fits into an evolutionary development or a more traditional waterfall process. Risk assessment activities are defined and demonstrated in early stages of development, specifically in conception/requirements activities, architectural or high-level design, and low-level design. The risk treatment trade-off tool used in risk assessment cycles will be discussed, in its current implementation as a Bayesian Belief Network (BBN), and he current version of the BBN topology will be presented. The AORDD framework was developed primarily with security risks as its focus, so extensions required to include other non-functional properties (specifically performance) will also be presented.
==========

11:30: Dr. Jan Jürjens, TU München, http://www4.in.tum.de/~juerjens/
Model-based Security Engineering with Aspects

The current state of the art in security-critical software is far from satisfactory: New security vulnerabilities are detected on an almost daily basis. To improve this situation, we develop techniques and tools that perform an automated analysis of software artefacts for security requirements (such as secrecy, integrity, and authenticity). These artefacts include specifications in the Unified Modeling Language (UML), annotated source code, and run-time data such as security permissions. This talk will focus on how to make use of ideas from aspect-oriented development in this context.
==========

12:30: LUNCH (provided)

==========

1:30 pm: Dr. Murray Woodside, Carleton University,
http://www.sce.carleton.ca/faculty/woodside.html

Performance by Unified Model Analysis (PUMA)
Evaluation of non-functional properties of a design (such as performance, dependability, security, etc.) can be enabled by design annotations specific to the property to be evaluated. Performance properties, for instance, can be annotated on UML designs by using the "UML Profile for Schedulability, Performance and Time (SPT)" standard. However the communication between the design description in UML and the tools used for non-functional properties evaluation requires support, particularly for performance where there are many alternative analysis tools that might be applied. This talk describes a tool architecture called PUMA, which provides a unified interface between different kinds of design information and different kinds of performance models.
==========

2:30 pm: Dr. Dorina Petriu, Carleton University, http://www.sce.carleton.ca/faculty/petriu.html

Performance Analysis of Aspect-Oriented Models

The talk presents an approach for analyzing the performance effects of a given aspect on the overall system performance. The first step is to add SPT performance annotations to the UML primary and aspect model(s). A generic aspect model will be converted into a context-specific aspect model with concrete values assigned to its performance annotations. By composing the latter with the primary model, a complete annotated UML model is obtained. Applying known techniques such as PUMA, the complete UML model is transformed into a performance model, which can be analyzed with existing solvers. The proposed approach is illustrated with a case study system, whose primary model is enhanced with security features treated as aspects.
==========

3:30 - 4:00 pm: Discussions and Conclusions

==========