Critical Systems Modelling with UML and Model-based Simulation (Tutorial)

Jan Jürjens*
Software and Systems Engineering
TU Munich, Germany

1 Motivation

The high quality development of critical systems (be it dependable, security-critical, real-time, or performance-critical systems) is difficult. Many critical systems are developed, deployed, and used that do not satisfy their criticality requirements, sometimes with spectacular failures.

Systems whose correct functioning human life and substantial commercial assets depend on need to be developed very carefully. Systems that have to operate under the possibility of system failure or external attack need to be scrutinized to exclude possible weaknesses.

Part of the difficulty of critical systems development is that correctness is often in conflict with cost. Where thorough methods of system design pose high cost through personnel training and use, they are all too often avoided.

UML offers an unprecedented opportunity for high-quality critical systems modelling that is feasible in an industrial context.

- As the de-facto standard in industrial modeling, a large number of developers is trained in UML.
- Compared to previous notations with a user community of comparable size, UML is relatively precisely defined.
- A number of analysis, testing, simulation, transformation and other tools are developed to assist the every-day work using UML.

*juerjens@in.tum.de – http://www4.in.tum.de/~juerjens

However, there are some challenges one has to overcome to exploit this opportunity, which include the following:

- Adaptation of UML to critical system application domains.
- Correct use of UML in the application domains.
- Conflict between flexibility and unambiguity in the meaning of a notation.
- Improving tool-support for critical systems development with UML.

The tutorial aims to give background knowledge on using UML for critical systems development and to contribute to overcoming these challenges.
It includes an interactive tool demo with advanced tool support for UML.

2 Outline

The tutorial presents the current academic research and industrial best practice by addressing the following seven main subtopics (of each about 20-30 min. duration):

- UML basics, including extension mechanisms
- Applications of UML to
  - dependable systems
  - security-critical systems
  - real-time systems
  - performance-critical systems
- Extensions of UML (UML-RT, UMLsec, UMLsafe, ...)
- Using UML as a formal design technique for the development of critical systems.
- Critical systems development methods.
- Modeling, synthesis, code generation, testing, validation, and verification of critical systems using UML, in particular: Using the standard model interchange formats (XMI) for tool integration and to connect to validation engines. Existing tools.

- Case studies.
- Interactive tool demo.

As an example application domain, we focus on safety- and security-critical systems. We also show how to generalize the approach to the other application domains mentioned above.

3 Goals and Objectives

By the end of the tutorial, the participants will have knowledge on how to use the UML for a methodological approach to critical systems development. They will be able to use this approach when developing or analyzing critical systems, by making use of existing solutions and of sound methods of critical systems development.

4 Intended audience

The tutorial addresses practitioners (i.e. system and software developers, architects, and technical managers) and researchers interested in critical systems development using UML (in particular for dependable, security-critical, or real-time systems).

5 Expected background

Some basic knowledge of object-oriented software, UML and safety- or security-critical software is assumed.

Level: Beginner to Advanced.

6 Format

Lecture with examples of running programs. Generous time for question and answers will be provided.

7 Handouts

The following material will be distributed to each participant.

- tutorial notes

- printouts of slides
- a copy of the book "Secure Systems Development with UML" (Springer-Verlag, 2004) by the presenter (see http://www.springeronline.com/sgw/cda/frontpage/0,10735,1-40109-22-2903821-0,00.html)[1]

8 History

The proposed tutorial is an adaption and extension of a series of about 30 tutorials presented at international conferences (see http://www4.in.tum.de/~juerjens/csdumltut then click on History (for slides and audio, need: user Participant, password Iwasthere)).

Feedback from these tutorials was gained through questionnaires distributed and collected at the tutorials and was used to improved the tutorial.

9 Biography

Jan Jürjens is a researcher at TU Munich (Germany). He is the author of a book on Secure Systems Development with UML (Springer-Verlag, 2004) and about 40 papers in international refereed journals and conferences, mostly on computer security and safety and software engineering, and has given several invited talks at international conferences. He has created and lectured a course on secure systems development at the University of Oxford and about 30 tutorials at international conferences. He is the initiator and current chair of the working group on Formal Methods and Software Engineering for Safety and Security (FoMSESS) within the German Society for Informatics (GI). He is a member of the executive board of the Division of Safety and Security within the GI, a member of the advisory board of the Bavarian Competence Center for Safety and Security, a member of the working group on e-Security of the Bavarian regional government, and a member of the IFIP Working Group 1.7 "Theoretical Foundations of Security Analysis and Design". He has been leading various security-related projects with industry.

See http://www4.in.tum.de/~juerjens for more details.

[1] Conference budget permitting.

10 Equipment

Beamer (for laptop), overhead projector, flipchart.