SAC 2015 – Tutorial Proposal

Interaction Design for Specifying Requirements

Dr. Hermann Kaindl, Professor
Vienna University of Technology, ICT
Gusshausstr. 27-29
A-1040 Vienna, Austria
Phone: +43 1 58801-38416
Fax: +43 1 5 99666-938416
Email: kaindl@ict.tuwien.ac.at
Web: http://www.ict.tuwien.ac.at/kaindl
**Title**
Interaction Design for Specifying Requirements

**Presenter**
Dr. Hermann Kaindl, Professor  
Vienna University of Technology, ICT  
Gusshausstr. 27-29  
A-1040 Vienna, Austria  
Phone: +43 1 58801-38416  
Fax: +43 1 5 99666-938416  
Email: kaindl @ ict.tuwien.ac.at  
Web: http://www.ict.tuwien.ac.at/kaindl

Hermann Kaindl joined the Institute of Computer Technology at the Vienna University of Technology in early 2003 as a full professor, where he also serves in the Senate. Prior to moving to academia, he was a senior consultant with the division of program and systems engineering at Siemens AG Austria. There he has gained more than 24 years of industrial experience in software development and human-computer interaction. He has published five books and more than 170 papers in refereed journals, books and conference proceedings. He is a Senior Member of the IEEE, a Distinguished Scientist member of the ACM, a member of the AAAI, and is on the executive board of the Austrian Society for Artificial Intelligence.

He has previously held tutorials at CAiSE’00, RE’01, RE’02, HICSS-36, INCOSE’03, RE’03, CADUI-IIU’04, INCOSE’04, RE’04, HICSS-38, IRMA’05, INCOSE’05, AAAI’06, HCI’06, OOPSLA’06, HICSS-40, ICONS’07, INCOSE’07, AAAI’07, IFIP Interact’07, OOPSLA’07, HICSS-41, ICCGI’08, RE’08, ICSEA’08, ICWW’09, IFIP Interact’09, SMC’09, HICSS-43, ACHI’10, ACM EICS’10, ICSEA’10, TdSE’10, HICSS-44, ACM SAC’11, INCOSE’11, AAAI’11, RE’11, ICSEA’11, HICSS-45, ACM SAC’12, ACM CHI’12, PROFES’12, BCS HCI’12, IEEE APSEC’12, HICSS-46, ACM SAC’13, NexComm’13, PROFES’13, ICSEOFT’13, IEEE Africon’13, IEEE APSEC’13, HICSS-47, ACM SAC’14 and WEB’14.

Several of these tutorials were related to the one proposed here, most strongly the one at HICSS-47. Note, that this tutorial is about more recent and more advanced material than the related one I gave at SAC’12.

In addition, Hermann Kaindl organized and chaired several panels at major conferences, such as the one at CHI 2001 “Methods and Modeling: Fiction or Useful Reality?”, as well as the one at RE’08 entitled “How to Combine Requirements Engineering and Interaction Design?”.

**Duration**
Half day

**Abstract**
When the requirements and the interaction design of a system are separated, they will most likely not fit together, and the resulting system will be less than optimal. Even if all the real needs are covered in the requirements and also implemented, errors may be induced by
human-computer interaction through a bad interaction design and its resulting user interface. Such a system may even not be used at all. Alternatively, a great user interface of a system with features that are not required will not be very useful as well.

This tutorial explains joint modeling of (communicative) interaction design and requirements, through discourse models and ontologies. Our discourse models are derived from results of human communication theories, cognitive science and sociology (even without employing speech or natural language). While these models were originally devised for capturing interaction design, it turned out that they can be also viewed as specifying classes of scenarios, i.e., use cases. In this sense, they can also be utilized for specifying requirements. Ontologies are used to define domain models and the domains of discourse for the interactions with software systems. User interfaces for these software systems can be generated semi-automatically from our discourse models, domain-of-discourse models and specifications of the requirements. This is especially useful when user interfaces for different devices are needed. So, interaction design facilitates requirements engineering to make applications both more useful and usable.

Motivation, target audience, and interest for the SAC community
The primary motivation for giving this tutorial is to improve software development in practice both regarding requirements engineering and (communicative) interaction design.

The target audience is interaction designers, Web designers, requirements engineers, software engineers, systems engineers, or project managers. Also educators will benefit from this tutorial.

The application area of this tutorial is clearly software development, more precisely requirements engineering and interaction design. It closely relates to the SAC 2015 Tracks HCI, RE and SE.

Outline
This outline is adopted from previously held tutorials:

5min  Introduction
15min  Background
  - Requirements
  - Scenarios / use cases
  - Interaction design
  - Widgets for user interfaces
  - Ontologies
  - Speech acts

45min  Interaction design based on discourse modeling
  - Communicative Acts
  - Adjacency Pair
  - Rhetorical Structure Theory (RST) relations
  - Procedural constructs
  - Conceptual Discourse Metamodel

35min  Use case specification
  - Use-case report (RUP)
  - Use-case diagram
  - Sketch of flow of events through scenarios
  - Specification based on discourse modeling
45min  Exercises
   - Try to understand the model sketch of a discourse
   - Try to model a discourse yourself

30min  Sketch of automated user-interface generation
   - Integration and use of ontologies
   - Process of user-interface generation
   - Device tailoring through optimization
   - Examples of generated user interfaces
   - Unified Communication Platform

5min  Summary and conclusion

Specific goals and objectives
This course is targeted towards people who are supposed to work on the requirements or the interaction design in systems development. Whatever the roles of the tutorial participants actually are in their daily work, they should get a better understanding of “other” viewpoints and tasks and, in particular, a common approach.

This tutorial has the primary objective to address a potential separation of requirements engineering and interaction design. In order to improve the development of useful and usable software systems, you will learn about an approach to precisely specify use cases in terms of interaction design. As a positive 'side-effect', supporting user interfaces can be generated semi-automatically.

Expected background of the audience
The assumed attendee background is some familiarity with scenarios / use cases as well as interest in requirements and interaction design. There are no pre-requisites such as knowledge about Human-Computer Interaction or Requirements Engineering in general. Also the small selection of UML notation used will be explained.

Audio Visual equipment needed for the presentation
For the instructor, a computer screen projector (to be connected with my laptop computer) will be needed, capable of at least 1024 x 768. In addition, a flip-chart with pens of various colors is required.

   For the course attendees, paper for the flip-chart and a sufficient number of pens are needed for the group exercises.

Selected publications of the proposer related to this tutorial


