



International Summer School on Usability-Driven Software Architecture, August 22-24, 2005

Tampere University of Technology, Tampere, Finland
Institute of Software Systems

Place: Tietotalo (Korkeakoulunkatu 1), auditorium TB109.

Introduction

Traditionally, usability has been incorporated into software design process as research results of the requirements definition phase. However, usability issues are often strongly related to the architecture of the system: certain architectural solutions may either support or hamper usability requirements. Thus, taking into account the crucial role of usability in the overall quality of a software system, usability issues should be systematically considered in architectural design.

Recently, several approaches have been taken to bridge the gap between usability and software architecture. The summer school gives an overview of some of the major approaches integrating usability requirements with the software development process, especially with architecture design. The lecturers are internationally well-known researchers representing a wide spectrum of viewpoints on usability and software architectures.

Lecturers & lecture topics

Len Bass (SEI, Carnegie-Mellon University, Pittsburgh, USA)

Usability Supporting Architectural Patterns

Jan Bosch (Nokia Research Center, Helsinki, Finland)

Usability as a Quality Factor for Software Architecture

Philippe Palanque (University Paul Sabatier, Toulouse, France)

Model-based Construction and Validation of Safety-critical Interactive Systems

Fabio Paternò (Institute of Information Science and Technologies, Pisa, Italy)

Linking User Tasks and User Interfaces: a Model-based Approach

<http://www.cs.tut.fi/ohj/tapahtumat/summer05>



The programme

	Mon, 22. August	Tue, 23. August	Wed, 24. August
9.00-11.00 (2h)	Fabio I <ul style="list-style-type: none"> • Introduction to usability • Task analysis and modelling • Tools for task modelling • Task-based design 	Len I <ul style="list-style-type: none"> • Motivation • Introduction to usability supporting architectural patterns • Cancel USAP 	Philippe III <ul style="list-style-type: none"> • Cases studies • PetShop: a plate-forms for engineering interactive systems • A roadmap for notations, tools and plate-forms for interactive systems
11.30-12.30 (1h)	Jan I <ul style="list-style-type: none"> • Introduction to software architecture and architecture design 	Philippe I <ul style="list-style-type: none"> • Introduction to interactive systems modelling • Introduction to modelling processes 	Len II <ul style="list-style-type: none"> • Feedback USAP • Information reuse USAP • Aggregating data USAP
13.30-14.30 (1h)	Jan II <ul style="list-style-type: none"> • Assessing software architecture 	Philippe II <ul style="list-style-type: none"> • Interaction techniques and interactive systems modelling • State of the art for interactive systems modelling • Design rationale and modelling 	Len III <ul style="list-style-type: none"> • USAP pattern language • What does it mean for a USAP “to work” • Evidence to date that USAPs “work”
15.00-17.00 (2h)	Jan III <ul style="list-style-type: none"> • Specifying usability • Assessing architecture for usability • Improving usability through architectural patterns 	Fabio II <ul style="list-style-type: none"> • Model-based design of multi-device interfaces • Semantic redesign for different interaction platforms • Migratory interfaces 	Discussion
17.15-	Panel Welcome Party	Dinner	



Lecture topics

Len Bass: Usability Supporting Architectural Patterns

Iterative design is a common practice when designing user interfaces. This leads to a sequence of changes to the user interface late in the software engineering life cycle. The software engineering response to these late changes is to use the Model View Controller (MVC) architectural pattern (or other separation based pattern). This has been effective and works well except for those changes that require software modifications that cut across the elements of MVC. The late-breaking requests for changes that cut across MVC are difficult to implement in a time fashion and tend not to get made. This suggests that these types of requirements should be identified early in the life cycle. Usability Supporting Architectural Patterns (USAPs) are an attempt to identify this class of requirement and provide information that will allow the development team to do a cost/benefit analysis about their inclusion early in the life cycle. This portion of the summer school will describe USAPs and give examples of their use and validation.

About the speaker: [Len Bass](#) is a Senior Member of the Technical Staff at the [Software Engineering Institute](#) (SEI) and participates in the [High Dependability Computing Program](#). He has written two award winning books in software architecture as well as several other books and numerous papers in a wide variety of areas of computer science and software engineering. He is currently working on techniques for the methodical design of software architectures and to understand how to support usability through software architecture. He has been involved in the development of numerous different production or research software systems ranging from operating systems to database management systems to automotive systems.

Jan Bosch: Usability as a Quality Factor for Software Architecture

Usability is one of the central quality attributes shaping software architecture. The lectures first discuss the quality-driven design and assessment of software architecture in general. The nature of usability as a quality attribute is studied, and the assessing of software architecture against usability requirements is elaborated. Finally, architectural patterns and their role in improving architectural support for the usability of a software system is discussed. A number of cases to illustrate the concepts is discussed throughout the lectures.

About the speaker: Prof. dr. ir. Jan Bosch is a VP and head of the Software and Application Technologies Laboratory at Nokia Research Center, Finland. Earlier, he headed the software engineering research group at the University of Groningen, The Netherlands, where he holds a [professorship in software engineering](#). He received a MSc degree from the University of Twente, The Netherlands, and a PhD degree from Lund University, Sweden. His research activities include software architecture design, software product families, software variability management and component-oriented programming. He is the

International Summer School on Usability-Driven Software Architecture

author of a book "Design and Use of Software Architectures: Adopting and Evolving a Product Line Approach" published by Pearson Education (Addison-Wesley & ACM Press), (co-)editor of several books and volumes in, among others, the Springer LNCS series and (co-)author of a significant number of research articles. He has organized numerous workshops, served on many programme committees, including the ICSR'6, CSMR'2000, ECBS'2000, GCSE, SPLC and TOOLS conferences and is member of the steering groups of the GCSE and WICSA conferences. He was the PC co-chair of the 3rd IFIP (IEEE) Working Conference on Software Architecture (WICSA-3), the general chair for WICSA-4 and PC co-chair for the 8th International Conference on Software Reuse (ICSR-8).

Philippe Palanque: Model-based Construction and Validation of Safety-critical Interactive Systems

Modeling is one of the ways of dealing with the intrinsic complexity of interactive systems. Modeling provides a way of describing at a higher level of abstraction than programming languages the behavior of interactive systems. If the notation for building models is formally grounded, this provides opportunity for verifying properties over the models. Such properties can easily be seen as necessary conditions for usable interactive systems (e.g. whatever state the system is in, there is at least one interactive object enabled on the user interface). However, iterative processes that are usually applied for designing interactive systems, call for ways of refining and modifying models in an easy and efficient way. To this end, software architectures are not only useful for making modifications local in the models but also for supporting the construction of large models by providing structuring frameworks. This part of the summer school presents the current state of the art in the field of interactive systems modelling as well as current research trends in formal description techniques for interactive systems.

The issue of modelling interactive systems is addressed through several case studies showing:

- the impact of the interaction technique with respect to the expressive power of the notations (WIMP, direct manipulation, multimodality)
- the impact of the expressive power on validation and verification techniques
- how software architectures applied to models can support efficiently structuring and modifiability of models

The case studies will be studied using two software plate-forms developed at LIHS

- PetShop environment for the formal description of interactive systems
- DREAM environment for supporting exploration of options and traceability of choices in the modelling phases

About the speaker: [Philippe Palanque](#) is a professor in Computer Science at University Paul Sabatier, Toulouse, France, and the head of Logiciels Interactifs et Interaction Homme-Système ([LIHS](#)). He has been working for about 15 years on notations and tools for the specification, prototyping, validation and implementation of Safety Critical Interactive Systems. He has been involved in the design of a notation called Interactive Cooperative Objects and in the implementation of the supporting case tool called PetShop, applying the approach to several application domains including Air Traffic Control, Military and Civil

International Summer School on Usability-Driven Software Architecture

aircraft Cockpits as well as several real-time command and control systems. He is currently chairing the IFIP 13.5 working group on Human Error, Safety and Systems Development.

Fabio Paternò: Linking User Tasks and User Interfaces: a Model-based Approach

The increasing availability of many interactive devices has forced designers to strive to make applications run on a wide spectrum of platforms in order to enable users to seamlessly access information and services regardless of the device they are using. This raises the fundamental issue of how to assist software designers and developers in building such applications, with the consequent need for novel methods and tools for the development of interactive software systems able to adapt to different targets while preserving usability. Comprehensive solutions can be obtained through the use of model-based approaches, which aim to support development through the use of meaningful abstractions to avoid dealing with low-level details, and then generate the specific version adapted for each device and modality through intelligent tools. We devote particular attention to the relations among tasks and the potential interaction platforms models, to see how the change of device can affect task performance. We also discuss how to represent task models for nomadic applications and analyse their content and their combined use with informal descriptions, such as scenarios. The final part will be dedicated to run-time adaptation to different devices, paying attention to migratory interfaces, which can change device dynamically preserving continuity of task performance, explaining what they are, why they can be important for users, how it is possible to obtain them.

About the speaker Fabio Paternò is a senior researcher and the head of the laboratory on [Human Interfaces in Information Systems](#) at ISTI, Pisa (Institute of Information Science and Technologies). He developed the [ConcurTaskTrees](#) notation for specifying task models and has also designed an associated environment ([CTTE](#)) to support the development and analysis of task models specified through this notation, which has been used in [various industries and universities](#). He was the coordinator of the [MEFISTO Long Term Esprit European Project](#) (September '97-December 2000) in which a number of methods for the design and evaluation of interactive safety-critical applications were developed and applied. He is currently involved in the European [ADVISES](#) TMR that is addressing similar topics. In these years he has also been working on methods and automatic support for usability evaluation. He is the scientific coordinator of the [CAMELEON R&D IST Project](#), a three-year project, which aims to develop methods and tools for the design of context-dependent interactive applications. His current research interests include [Migratory Interfaces](#), Methods and Tools for Multimodal User Interface Design and Evaluation, User Interfaces for Mobile Devices, Model-Based Design of Interactive Systems, [End-User Development](#), Usability Evaluation of Web Sites, and Design of User Interfaces for Safety Critical Interactive Systems. He is the author of the book on [Model-Based Design and Evaluation of Interactive Application](#)

Participation

Participation fee

Fee: 275 Euro

The fee covers proceedings, lunches, break drinks, welcome party, and dinner.

Accepting participants

To apply for participation, please fill in the form at

http://www.cs.tut.fi/ohj/tapahtumat/summer05/summer2005_participating.html

Or send email to issudsa@cs.tut.fi containing your full name, e-mail and your organization.

The registration is confirmed only after the receipt of the participation fee. The instructions how to pay the participation fee will be sent to you after you have applied for the participation.

The deadline for registration is June 17, 2005. Later applications may be accepted within the limits of space.

TUT PhD students

The PhD students of Tampere University of Technology will get 2 credit units after writing an accepted essay of the contents of the summer school and selected articles.

Contact persons

General Info (mail forwarded to all persons below): issudsa@cs.tut.fi

Researcher Jarmo Palviainen tel +358 3 3115 3076, GSM +358 40 7770 350

Prof. Kai Koskimies tel +358 3 3115 2940, GSM +358 40 849 0748

Prof. Kaisa Väänänen-Vainio-Mattila tel +358 3115 3901, GSM +358 40 849 0731