International Conference on Global Software Engineering

ICGSE 2010

Princeton, NJ, USA
August 23-26, 2010

Software development endeavors that cross national borders are pervasive in a wide range of industries from software and IT to automotives and aeronautics. The increased globalization of software development creates software engineering challenges due to the impact of a wide range of factors such as distance, time zones, or diversity of culture and communication. Addressing these challenges requires novel and effective technical and organizational approaches in order to archive the ever increasing demands on productivity and quality.

The 5th International Conference on Global Software Engineering (ICGSE) to take place in Princeton, New Jersey, August 23-26, 2010, provides the perfect setting for bringing together researchers and practitioners interested in understanding how globally distributed product development team work, the challenge they face and, more importantly, how to improve them.

The ICGSE conference series has a strong track record of presenting high quality research as well as best practices from academic and industrial communities and ICGSE 2010 is no exception. This year have a strong program composed of a doctoral symposium, posters, 3 workshops, 5 tutorials, 3 highly regarded keynote speakers, a panel discussion as well as 28 research, industrial and educational presentations. Conference attendees will get a unique opportunity to learn about the current state of practice distributed software development as well as new thoughts and trends that will shape the future of software development.

Keynote Speakers

Speculations on Coordination Models

Speaker: *Len Bass*
Date: Tuesday Aug 24th

Two modules that interact in a system set up a requirement for the development teams for each module to coordinate. From this basic idea one can derive a wide variety of different coordination models. In this theoretical talk, I will speculate about different coordination models and the circumstances in which they might be appropriate.

Len Bass is a senior member of the technical staff at the Software Engineering Institute (SEI) at Carnegie Mellon University. He has written two award- winning books in software architecture, in addition to several other books and numerous papers in a wide variety of areas of computer science and software engineering. He has more than 45 years experience in software development and research in multiple domains ranging from embedded systems to information systems.

Global transfer of work and its impact on quality and productivity

Speaker: *Audris Mockus*
Date: Wednesday Aug 25th

Present software development business practices are trying to emulate the success of manufacturing process by off-shoring software development to countries with lower labor costs and higher availability of workers. The relatively more complex domain of software development is making it difficult to achieve cost savings comparable to off-shored manufacturing and to maintain high standards of quality. We investigate code transfer to offshore locations and to outsourcing organizations and quantify the impact of such transfers on software quality and developer productivity.

Audris Mockus received B.S. in Applied Mathematics from Moscow Institute of Physics and Technology in 1988. In 1991 he received M.S. and in 1994 he received Ph.D. in Statistics from Carnegie Mellon University. He works at Avaya Labs Research. Previously he worked at Software Production Research Department of Bell Labs.

On the impact of SOA on GSD: open issues and challenges
Speaker: *Elisabetta Di Nitto*
Date: Thursday Aug 26th

Software systems based on a Service-Oriented Architecture (SOA) are designed in terms of composition of services. Services, therefore, represent the main unit of design and usually allow for a good decoupling among the various parts of the system. On the one side, such a decoupling can likely result in a good decoupling of the development tasks to be accomplished, and, therefore, make SOA suitable for Global Software Development (GSD). On the other side, the full adoption of the SOA philosophy also requires a change in the perspective of GSD and in the organization of GSD practices. Services, in fact, can be operated by third parties and simply used, not acquired, by the organization running the software system. Moreover, these services can easily be replaced by others during the operation of the system. These two aspects result in the fact that new roles need to be considered as relevant to GSD and new coordination and business models enter in the game. While in usual GSD practices the focus is on the coordination between roles focusing on the design, development, and validation of the software system, in the case of SOA, run-time roles, such as the service operator or the service intermediary, have to be considered as well. Indeed, the interaction among these roles may occur through the whole life cycle of the SOA-based system and may be regulated by specific business models (e.g., pay per use, service provision with SLA guarantee, ...). More in general, while GSD is typically focusing on the globalization of the development phases, SOA enables the globalization of the run-time phases thus introducing a new set of issues that GSD and, more in general, Global Software Engineering (GSE) have to take into account. In this keynote, starting from the relevant literature, I discuss on the above aspects and identify new challenges for future research.

Elisabetta Di Nitto is an associate professor at Politecnico di Milano, where she earned her Ph.D. in Computer Science. She teaches Software Engineering and Foundations of computer science. Her current research interests are on process support systems, SOA, dynamic software architectures, and autonomic, self-adaptive systems. She has been researcher at CEFRIEL and visiting professor at University of California, Irvine. She has published and presented various papers on the most important international journals and conferences and has served in the program committee of various international conferences. She is currently program co-chair of the 25th Automated Software Engineering Conference (ASE 2010), member of the Transactions on Software Engineering Editorial Board, and workshop co-chair for ICSE 2010. She has been the principal investigator for various projects funded by the EU and by the Italian ministry of research.

Tutorials

Requirements Engineering for Large and Very Large Globally Distributed Systems

Part I Processes and Tooling

Summary of Content
Requirements engineering processes in the large are very different from those on small to medium size systems. Most texts on requirements engineering, for example tend to describe small or medium scale systems (up to about 5K requirements), or if the systems are large, describe the processes associated with product development and product lines. Industrial projects, however, start at 20K requirements and up. Adding globalization further increases the difficulty of successfully eliciting and managing requirements. The creation of global, scalable processes that will support a large infrastructure project of the kind we are reading about in the headlines today (e.g. the A380 or Boeing Dreamliner) can be a daunting prospect. This tutorial will present some of the challenges associated with the requirements engineering of such large and very large projects, and describe how to setup processes and tooling to support them.

This is part 1 of a two part tutorial. The morning session describes how to set up processes and tooling for very large globally distributed industrial projects, typically dealing with 50 thousand or more requirements. The afternoon session describes the unique nature of requirements engineering processes for contract based projects. Either session may be taken independently, but the attendee will get the most out of the tutorial by attending both sessions.

Instructor's Bio

Brian Berenbach is the manager of the requirements engineering competency center at Siemens Corporate Research, and is an ACM distinguished engineer. His book, *Software and Systems Requirements Engineering: In Practice* has just been published by McGraw-Hill.

Intended Audience
This tutorial will be of interest to the practitioner who is involved with the management and execution of large and very large projects. It may also be of interest to educators desiring to understand the dynamics of requirements engineering on large projects.

Tutorial Date: Monday Aug 23rd

Tutorial Length: 1/2 Day

Tutorial Level: Intermediate

Requirements Engineering for Large and Very Large Globally Distributed Systems

Part II Contract Based Systems

Summary of Content

Requirements elicitation and management for contract based projects is significantly more complex than for product or product line development. For example, many practitioners are unaware of the fact that the traditional "V" model for requirements tracing does not work where there is a legal contract describing project deliverables; nearly every aspect of requirements engineering is more challenging, from elicitation to risk analysis and compliance management. Adding globalization makes the project even more challenging, for example, managing regulatory codes across multiple countires. This half-day tutorial will describe in some detail contract issues that are typically not discussed in requirements texts and courses.

This is part two of a two part tutorial. The morning session described how to set up processes and tooling for very large global industrial projects. This session describes the unique nature of requirements engineering processes for contract based projects that may span multiple legal entities, e.g. the U.S. and Canada. Either session may be taken independently, but the attendee will get the most out of the tutorial by attending both sessions.

Instructor's Bio

Brian Berenbach is the manager of the requirements engineering competency center at Siemens Corporate Research, and is an ACM distinguished engineer. His book, *Software and Systems Requirements Engineering: In Practice* has just been published by McGraw-Hill.

Intended Audience

This tutorial will be of interest to the practitioner who is involved with the management and execution of large and very large global projects, especially when they involve being either the customer or supplier for a contract based project. It may also be of interest to educators desiring to understand the dynamics of requirements engineering on large projects as it covers material typically not found in books or journal articles.

Tutorial Date: Monday Aug 23rd

Tutorial Length: 1/2 Day

Tutorial Level: Intermediate

Implementing Agile Software Development across Time Zones

Summary of Content

This half-day tutorial deals with implementing agile software development in time-zone challenged projects. Specifically, we describe the challenges and how they can be met using agile practices and governance mechanisms. The goals of the tutorial are:

- Learn the characteristics of working across time zones
- Learn how agile software development answers the challenges while working across time zones
- Examine how governance mechanisms assist in working across time zones

Instructors' Bios

Dr. Yael Dubinsky is affiliated with the Software and Services group in IBM Haifa Research Lab. For more than ten years she is also the instructor of project-based courses in the Department of Computer Science, Technion - IIT. Her research interests involve aspects in software engineering and information systems. Yael has a significant experience with guiding agile implementation processes in the industry and academia. She has experience in organizing workshops and facilitating tutorials since 2003. Her book on Agile Software Engineering co-authored with Orit Hazzan was published by Springer in 2008.

Professor Erran Carmel's area of expertise is globalization of technology. He studies global software teams, off-shoring of information technology, and emergence of software industries around the world. He is currently working on a book on coordination across time zones. His 1999 book "Global Software Teams" was the first on this topic and is considered a landmark in the field helping many organizations take their first steps into distributed tech work. His second book "Offshoring Information Technology" came out in 2005 and has been especially successful in outsourcing and offshoring classes. He has written over 80 articles, reports, and manuscripts. He consults and speaks to industry and professional groups.

Intended Audience

This tutorial is relevant for software practitioners who are interested in implementing agile software development in distributed projects.

Tutorial Date: Monday Aug 23rd

Tutorial Length: 1 Day

Tutorial Level: Intermediate

What Did You Say? Cultural Influences on Communication and Understanding

Summary of Content

What one says to compatriots in face-to-face conversation is often misunderstood; imagine the possibilities for misunderstandings with someone from halfway around the world, natively speaking another language, and living in a different culture! In such circumstances how can you be sure that your collocutor has understood you in face-to-face (hard), telephone (harder), and email (hardest) conversations? The ubiquity of English facilitates basic communication, but its use as a common language frequently disguises cultural differences. Regardless of language, clear communication is essential for success in any collaborative undertaking whether done by a small co-located group or by a globally dispersed team.

This tutorial describes frameworks useful in understanding cultural differences and gives real-life examples of misunderstandings due to such differences. Expect to take away practical tools to understand your own cultural biases and in-class practice to boost your communication abilities with colleagues from other cultures. You will also learn about frameworks for understanding other cultures based on work by Geert Hofstede, Fons Trompenaars, and others as well as on the presenter's own experiences.

Instructors' Bio

Frederick Zarndt has lived and worked in the USA, Germany, Switzerland, Libya, Argentina, India, and Israel and visited many, many other countries for business and pleasure. Both as an individual contributor and as a manager, he has more than 25 years experience in business and software development at companies ranging from Seismograph Service Corporation in Libya and Argentina, to Siemens-Albis in Switzerland and Germany, to Novell in the USA, to an internet startup in Utah USA, and to Planman Consulting, an Indian outsourcing firm. Frederick has presented similar tutorials at ICGSE 2007, ICGSE 2009, CIRCUS RE 2009, the 2008 74th IFLA General Conference in Quebec, and HICSS 2010. He has given the tutorial to companies in Germany, India, Singapore, and the USA and recently at the Institute of Systems Sciences, National University of Singapore. He has co-chaired the Virtual Global Teams track at the 42nd and 43rd Hawaii International Conference on Systems Sciences and will again co-chair the track for the 44th HICSS. Frederick has taught workshops about personal development and communications since 1988. Frederick has a Master of Science in Physics, Master of Science Computer Science, and undergraduate degrees in physics, mathematics, and computer science.

Intended Audience

Anyone can benefit, but especially those who frequently communicate with colleagues from other cultures. No prior knowledge is required, although experience with other cultures is helpful.

Tutorial Date: Monday Aug 23rd

Tutorial Length: 1/2 Day

Tutorial Level: Basic - Intermediate

Global Software Engineering Lessons from Industry:

Distributed Software Development, Outsourcing, and Supplier Management

Summary of Content

Software and IT industries are today truly global, and so is software engineering. Be it offshoring or outsourcing, component or service integration - managing global software engineering has rapidly become a key competence for successful engineers and managers. The diversity of suppliers, cultures and products require dedicated techniques, tools, and practices to overcome challenges. Session attendees will get an opportunity to explore the current state of practice in this area as well as new thoughts and trends that will shape the future.

Instructor's Bio

Christof Ebert is managing director and partner at Vector Consulting Services. He is helping clients worldwide to improve technical product development and to manage organizational changes. Prior to that, he held engineering and management positions for fifteen years in telecommunication, IT and transportation. While at Alcatel, he had global responsibility for processes, tools and software technology and was ramping up a captive offshore center on CMMI maturity level 5. Over the years he had set up several offshoring sites and supported numerous companies in mitigating global software engineering and outsourcing risks and thus get concrete benefits from such programs. He is the author of "Going Global - Distributed Software Development, Outsourcing, and Supplier Management" which appears in its second edition at Wiley IEEE in 2010. He serves on the executive board of the IEEE International Conference on Global Software Engineering (ICGSE) series, teaches at the University of Stuttgart and is a SEI certified CMMI Instructor.

Intended Audience

Software engineers, project and line managers.

Tutorial Date: Monday Aug 23rd

Tutorial Length: 1/2 Day

Tutorial Level: Basic - Intermediate

Workshops

Methods and Tools for Project/Architecture/Risk Management in Globally Distributed Software Development Projects (PARIS'10)

The challenging issues arising in the field of offshore software engineering projects require novel approaches in risk analysis, project planning, architecture and methods in order to handle the bounded financial and technical risks.

The goal of this workshop is to provide a forum for researchers and professionals interested in global software development to discuss and exchange ideas. In particular, this workshop takes the perspective of the practitioner and focuses on the techniques that will help software professionals to meet the unique challenges in a global development environment. Thus, the major goal of this workshop is to discuss novel methodologies for risk management for global software development. Additionally, we want to provide a platform bringing together researches and practitioners in order to share their knowledge and requirements in the field of offshore software development.

Workshop Organizers
Alberto Avritzer, Siemens Corporate Research, USA
Juho Mäkiö, Research Centre for Information Technologies, Karlsruhe, Germany
Stefanie Betz, University of Karlsruhe, Germany
Rafael Prikladnicki, PUC – RS, Brazil

Workshop Date: Monday Aug 23rd

Workshop Website: http://www.outshore.org/ICGSE/ICGSE10/tabid/66/Default.aspx

4th International Workshop on Tool Support Development and Management in Distributed Software Projects (REMIDI'10)

Distributed projects (often subsumed under terms like global software development (GSD), global collaboration, offshoring etc.) are common ways to overcome time and resource restrictions or lack of local expertise. In addition, current budget saving initiatives lead to higher international competition. Thus, software development projects take place in a global context. The participants will present and discuss project experiences, best practices, tool prototypes and new approaches - in academic research and in industry.

One of the objectives of this workshop is to structure the major research topics and to define a research agenda for further work in the area of "end-to-end" tool support in distributed system development. Besides that, there will be a demo session with presentations and live demonstrations of tools that are specifically dedicated to support distributed development projects.

Workshop Organizers
Patrick Keil, TU München, Germany
Marco Kuhrmann, TU München, Germany
Daniel Paulish, Siemens Corporate Research, USA

Workshop Date: Monday Aug 23rd

Workshop Website: http://www4.in.tum.de/~kuhrmann/remidi2010.shtml

Knowledge Engineering in Global software development (KNOWING'10)

Knowledge engineering is an evolutionary process of engineering artifacts and using them to gain new understandings. These new understandings are then used to further engineer or modify artifacts and the process continues. Knowledge engineering in global software development is being increasingly researched by the software and the knowledge management communities. To implement a good knowledge engineering strategy in a global software development environment, organizations have to overcome issues which have been identified in many previous studies. These issues make global software distribution a difficult and complex task. The effective and explicit engineering of knowledge had been presented as a key factor in the survival of companies in current business environments.

One major goal of the KNOWING workshop is to define a research agenda and establish a community around the topics of knowledge engineering, knowledge management, and knowledge sharing in global software development. Ultimately, we would like to develop a reference framework for knowledge engineering in overcoming the global software development challenges, in particular, how to relate social/economical aspects(e.g., cultural differences, geographic dispersion, time-zone difference, loss of communication richness, loss of teamness, co-ordination breakdown) with technological/software engineering aspects(e.g., product architecture, development methods/tools, collaborating technology, managerial technologies, communication infrastructure, team building).

Workshop Organizers

Ivan Mistrik, Heidelberg, Germany
Sarah Beecham, Lero/University of Limerick, Ireland

Workshop Date: Monday Aug 23rd

Workshop Website: http://www.lero.ie/KnowingWorkshop/Background.html