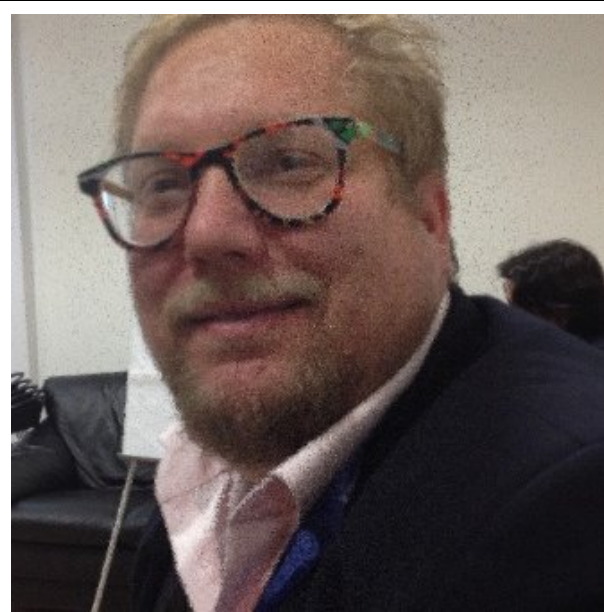# The Future of Research
Bolzano, December 1st and 2nd

Organiser: Barbara Russo

December 1st 2016,

Seminar room 1.0.1

11:00 - 12:00 Pekka Abrahamsson, Norwegian University of Science and Technology, Finland
Title: Software Engineering considered obsolete

Dr. Pekka Abrahamsson is currently serving as a professor of software engineering at Norwegian University of Science and Technology in Trondheim, Norway. He was previously the dean and a full professor of
computer science at Free University of Bozen-Bolzano in Italy. His background is in the software process improvement, agile software development and empirical software engineering research. He is the originator of the Mobile-D development methodology for mobile applications. Abrahamsson has researched and taught software startups
since 2011. He is the chairman of the Global Software Startup Research Network and was awarded the Nokia Foundation award in 2007.

Abstract: For the past fifty years every software research paper has been motivated by stating how crucial role the software and software powered systems play in the society. Industry has been worried about the so called software productivity gap for decades Indeed, Gartner estimated that there will be 21 billion Internet-of-Things (IoT) devices around us by 2020. Each of these devices needs to program ed to work according to pre-defined specifications. We tend to ask too quickly: Where do we find the software developers to do this enormous task? The answer is that we do not need to. Until now we have wanted to be more efficient in providing a computer with instructions on how to behave in various situations. There are hundreds of programming languages and human defined processes available to do that task. What if we could instruct the computers in some other way? Industry has already demonstrated that tomorrow's computers are in fact learned to behave in an appropriate way (rather than programmed). Instruction-based software construction of today will become a niche business needed only in certain areas. We need to finally stop arguing whether Agile is good enough process or if Java is more useful than C++, and think how to develop systems in the new era of computing.

16:00 - 17:00 Bernhard Peischl Technische Universitaet Graz, AT
Title: Climbing the stairways to quality: from test automation to intelligent systems

Bernhard Peischl is a senior scientist at the Institute of Software Technology and the scientific coordinator for the industrial competence network Softnet Austria (managing the innovation programs K-net Softnet Austria and COMET K-Projekt Softnet Austria II). He received a MS in telecommunications engineering (2001) and a Ph.D in computer science (2004) from the Graz University of Technology, Austria. He is responsible for managing the network's R&D activities and a number of applied research projects dealing with software test, fault localization and quality assurance. He has co-authored over 85 scientific articles on peer-reviewed workshops, conferences, magazines and in scientific journals.

Abstract: Quality assurance (QA) is no longer only about getting a product or service to market quickly and with as few defects as possible. Instead it is about the shifting of quality responsibilities along with automating critical QA enablement components, such as test data, test infrastructure and test environments. We believe automation of QA activities is not only required but it is a core enabler to achieve scale and build continuous QA. In the future, intelligent systems will assist in analyzing the root cause of defects, and the efficiency of test suites, in predicting test estimation effort based on requirements as well as in planning the priority of test cases. In continuous QA, intelligent systems thus will enable the processing and interpretation of design- as well as run time data.

In this talk, we will report on industry-academia collaboration in the field of test automation and in particular discuss applied research in test automation and the lessons learnt. Moreover, the talk presents an abductive approach to locate the root cause of a defect. In particular, this approach strives to overcome the gap between simulation models we create during system development, and the models we require for analyzing the root cause of defects in the deployed system.

17:00 Xiao's statement on future of research - Panel

19:30 Get together - dinner

December 2nd 2016
Seminar room 1.0.1

10:30 - 11:30 Brian Fitzgerald, Lero Irish Software Research Centre at University of Limerick, Ireland

Title: Two's Company, 1.6 is a Crowd: A Case Study of Crowdsourcing Software Development

Brian Fitzgerald is Chief Scientist at Lero – the Irish Software Research Centre. He also holds an endowed professorship, the Krehbiel Chair in Innovation in Business & Technology, at the University of Limerick (UL), and served as *Vice President Research* at UL from 2008-2011.

He has also held visiting positions in Italy, Austria, Sweden, US and UK. He holds a PhD from the University of London and his research interests lie primarily in software development, encompassing open source and inner source, crowdsourcing software development, agile and lean software development, and global software development.

His publications include 14 books, and over 150 peer-reviewed articles in the leading international journals and conferences in both the Information

Abstract: Crowdsourcing is an emerging and promising approach which involves delegating a variety of tasks to an unknown workforce—a crowd. Crowdsourcing has been applied quite successfully in various contexts – from basic tasks on Amazon Mechanical Turk to solving complex industry problems, e.g. Innocentive. Companies are increasingly using crowdsourcing to accomplish specific software devel-opment tasks. However, very little research exists on this specific topic. Our case study in a global ICT multinational company highlights on a number of challenges that arise when crowdsourcing software development. For example, the crowdsourcing development process is essentially a waterfall and this has to be eventually integrated with the agile approach used by the company. Crowdsourcing works better for specific software development tasks – those that do not have complex interdependencies. The development cost was much greater than originally expected, overhead in terms of company effort to prepare specifications and answer crowdsourcing community queries was much greater, and the time-scale to complete contests, review submissions and resolve quality issues was significant. Finally, quality issues were pushed later in the lifecycle given the lengthy process necessary to identify quality issues and have them resolved by the community. Given the emphasis in software engineering on identifying bugs as early as possible, this is quite problematic.

Systems and Software Engineering fields, including *MIS Quarterly (MISQ)*, *Information Systems Research (ISR)*, *IEEE Transactions on Software Engineering (TSE)* and *ACM Transactions on Software Engineering Methodology (TOSEM)*. Prior to taking up an academic position, he worked in the software industry for about 12 years, in a variety of sectors (including finance, telecommunications, manufacturing, bespoke software development) in a number of countries (Ireland, Belgium, Germany). He has been very successful in winning competitive research grants from a variety of funding agencies, including the EU, Science Foundation Ireland and Enterprise Ireland. Overall, these projects have received total funding of almost €100 million, with over €13 million received directly as Principal Investigator (PI).	

14:00 - 15:00 Kieran Conboy, Lero Irish Software Research Centre at NUI Galway, Ireland
Title: The Concept of Time in Software Methods Research

Kieran Conboy (@conboyk) is a Professor in Information Systems and leader of the Lero Irish Software research centre at NUI Galway. He previously worked for Accenture Consulting and the University of New South Wales in Australia. He has worked with organisations such as Atlassian, Cisco Systems, Suncorp, and Fidelity Investments, as well as many SMEs. Kieran has published over 100 articles in leading international journals and conferences including Information Systems Research, the European Journal of Information Systems, and the Journal of the AIS. A key focus of his current research is the examination of flow and agility across stages of method adoption. He is an editor of the European Journal of Information Systems and has chaired many international conferences in his field.

Abstract: There is no doubt that value time is a key metric in software development. It underpins not only the performance targets that most teams aspire to but also how these teams organise themselves and the work they do. However, time is often ignored in software research or is often viewed and studied in an overly simplistic manner. Complexities such as temporal synchronisation, temporal perceptions, our experience of time are often overlooked. This seminar examines the concept of temporality and these underlying complexities, drawing on literature from Greek mythology to contemporary frameworks for temporal evaluation. We will first discuss the importance of these concepts in software development and identify interesting topics for future software research. We will also examine how researchers can reflect on and improve their research methods to cater for these temporal complexities.

16:00 Closing

Makerspace
14:00 20:00 Makerspace event