Final report 2008 – 2018

Highlights from a decade of EASE

Industry Excellence Center - Embedded Applications Software Engineering

Year 2008 visions to the year 2018 world

In the fall of 2007, industry and academia partners together wrote a proposal. "Based on emergent needs in industry for knowledge and skills, an Industrial Excellence Center for Embedded Applications Software Engineering is proposed." The center was established in the fall of 2008 and has now operated for ten years, with collaboration and joint funding from industry, academia, and Vinnova.

The final year of operation – coinciding with the 50th anniversary of Software Engineering as a dicipline – concluded with a workshop. In a panel discussion, industry leaders unanimously confirmed the importance of industry-academia collaboration on software engineering to ensure their companies' needs for new and renewed competency.

Competency provisioning is – to a large extent – a regional eco-system. Companies want to hire well-educated students, getting good education at the universities, which in turn requires good teachers, being on the research front, which requires industry contacts to ensure relevance of the research. Research funding and industry champions are critical sources of fuel for the eco-system, and EASE has been such fuel for the regional eco-system in southern Sweden.

Software engineering has still many challenges to address after 50 years of research, and the needs for competency has increased during the duration of the ten years of the EASE center. We, at this point, close the center in its current form but keep working to gear up research and collaboration activities to meet the continuously growing needs in industry for software engineering knowledge and skills.

Lund and Karlskrona,
February 2019

Prof. Per Runeson
Director

Prof. Claes Wohlin
Codirector

EASE aimed to be a world class applied software research facility for embedded software applications. The objective was to ensure that industrial partners got a competitive advantage with respect to competency and innovation of novel solutions and effective engineering of embedded software applications with physical and logical mobility.

The centre impacted on the innovation system through provisioning of competency, via a continuous exchange loop between industry and academia, involving research challenges, industry personnel, researchers, students and research results. These range from technical solutions that can be used in products to improved work procedures for the development.

Costs

Industry cash	33 MSEK
Industry in-kind	25 MSEK
Academia	31 MSEK
Vinnova	18 MSEK
Total	107 MSEK

Outcomes

~500 industry-academia meetings
200+ papers
1 patent
10 annual workshops
8 PhDs

+ 100's of software engineers – each year!

EASE

A research program for industrial excellence supported by

AXIS COMMUNICATIONS, SONY, SOFTHOUSE, ERICSSON, VINNOVA, LUNDS UNIVERSITET Lunds Tekniska Högskola, BLEKINGE TEKNISKA HÖGSKOLA BTH

Project Summary

The project mainly referred to here is the third phase of a ten-year applied research programme run jointly between academia and industry. The project focused on three themes that have been formulated collaboratively between industry and academia:

A) Configuration and interaction in Internet-of-Things
B) Parallel execution for embedded systems using machine learning
E) Increased efficiency in software development through decision-support in the testing process

At the beginning of this phase, the earlier themes C) Efficient Software Methods (OSS, Agile) and D) Aligning Requirements and Verification, were continued in the joint theme E.

The themes have had regular meetings with their respective reference group. Through research studies and master thesis projects, the work has interconnected researchers, PhD students and other students with the companies, and the companies with each other.

Several methods and prototype implementations have been evaluated in an industrial context throughout the project. Some of these have been established as industrial practice at the participating companies. Furthermore, the iterative way of working has resulted in collaborative knowledge exchange between academia and industry. The exchange has inspired the companies to stepwise improvements and academia to new research challenges and improved educational programmes and courses.

The backbone of the programme has been PhD students working in close collaboration with the companies; in total eight PhD exams have been awarded based on the ten-year programme. More than 200 scientific publications have been published, and more than 500 meetings with people from industry and academia have taken place, which is on average more than one meeting per week over ten years.

The project closed with a with a workshop where high-level technical managers from the companies, in a panel, emphasised the value of continuous collaboration between industry and academia to ensure getting a competitive advantage and recruitment of highly competent people.

Project Retrospective

A retrospective exercise was performed with each theme at the end of the programme. The objective was to identify gains and good practices for industry-academy collaboration by reflecting on our research collaboration. A retrospective method based on pre-prepared times lines of project history was used; a method previously designed within EASE and applied to industrial software development projects.

Each theme discussed what and how the research was performed and identified results and gains for the partners, including both companies and academia. The timelines were used to prompt memory, and to identify chains of events, e.g. how new topics and studies emerged based on industry needs and discussion of these within the project, and how research results were conveyed and applied by partner companies.

E. Bjarnason, A. Hess, R. Svensson, B. Regnell, and J. Doerr (2014). Reflecting on evidence-based timelines. IEEE Software, 31(4):37-43. DOI:10.1109/MS.2014.26

Discussions within EASE Theme E while updating the timelines during the Project retrospective evaluation in October 2018 Photo: Julia Schön

Project Achievements

The overall goals of the programme concern:

- Availability of competent personnel
- Making results useful for industry
- Research excellence

The availability of competent personnel has been ensured through the close collaboration between practitioners from the companies and researchers and students from academia, as may be exemplified with:

- A literature study summarising the state-of-the-art
- Identification of improvement potential through in-depth analyses in the companies
- Development of solutions that have been evaluated at the companies
- Collaborative prototype work

Research studies and master theses conducted at the companies have contributed to making results useful in an industrial context. Furthermore, students on different levels also form a basis for recruitment.

The programme has, during its ten years, had the right balance between continuity and renewal. A core of people has been with the programme during its duration, although there has also been variation, in particular on the company side. The changes of personnel on the company side have occasionally slowed down the work since it takes some time for people to get into the programme. By having multiple industrial partners, the effects have to some extent been mitigated. The intensity of the collaboration has also been affected by the recruitment of new generations of PhD students, i.e. as PhD students defend their dissertations new PhD students are recruited. Potentially the effect of new PhD students could have been addressed if, for example, recruiting one new PhD student per year instead of recruiting several PhD students at the same time.

A distinguishing factor for the success of the programme regarding the usability of results, is that decisions about research studies have been taken locally. A local decision also means that the schedule may be determined jointly between the parties involved in the study. The length of the programme (ten years) has made the joint research independent of the schedules of different funding agencies.

The programme clearly shows that it is possible to combine applied research with scientific excellence. Several researchers from the programme have recently been top-ranked in an article published in the Journal of Systems and Software. Researchers from the programme have been ranked in several categories including, for example, experienced researchers and "rising stars".

Technical Outcomes

Although the general competence provisioning has been the key results, specific results emerge from the project as well. Below, we list some of these, mostly focused around one PhD student, provided with a key publication reference.

Interact with Things

Dr. Günter Alce

Internet of Things is a phenomenon that gradually melds into our daily lives. However, it comes with challenges of not-so-tech-savvy end-users who have to discover, configure and directly interact with numerous connected things at home, at work, and in new environments. We have explored core concepts in interaction models, as well as in systems models, to understand what constitutes future proof models. Some of these has been evaluated in experimental demo settings, which are available to try out.

G. Alce, A. Espinoza, T. Hartzell, S. Olsson, D. Samuelsson & M. Wallergård (2018). UbiCompass: An IoT interaction concept, Advances in Human-Computer Interaction, vol. 2018, 5781363. DOI: 10.1155/2018/5781363

Mine commit data in code repositories

Dr. Alma Orucevic-Alagic

As part of managing software development, it is necessary to understand how development is conducted in organisations, for example when procedures from OSS development is transferred to industrial settings. We have investigated the possibility of mining commits in code repositories to understand how developers have collaborated during development. To do this, it was necessary to define the concept of collaboration in terms of how to work in the same code units and to understand what graph-metrics are suitable. Based on the research we conclude that it is possible to understand collaboration from commit data.

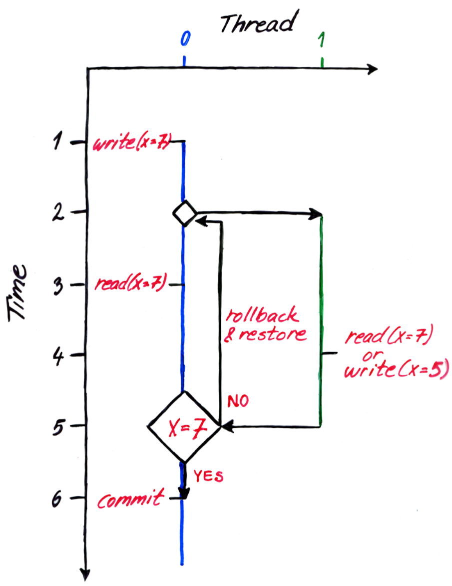
A. Orucevic-Alagic & M. Höst (2014). Network analysis of a large scale open source project. In 40th EUROMICRO Conference on Software Engineering and Advanced Applications, Verona, Italy, pp. 25-29. IEEE Computer Society. DOI: 10.1109/SEAA.2014.50

Parallelize computing

Dr. Jan Kasper Martinsen

Parallel computing is based on dividing computational tasks into smaller ones that can be executed in parallel on multiprocessors. However, dividing the tasks is often a labour intensive and disturbing job for programmers. What if the execution environment takes a linear program and speculates what can be parallelized? We explored this challenge, developed a patented solution and found that it does not work well for high-performance benchmarks, but improves the performance significantly for many applications in daily use, such as web applications written in JavaScript.

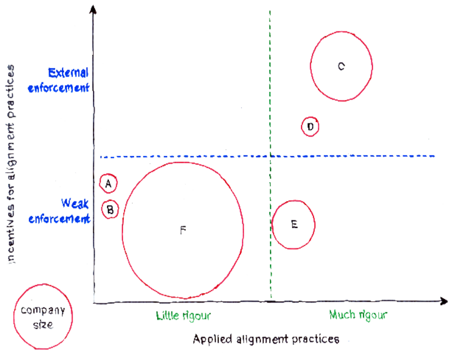
J. K. Martinsen, H. Grahn & A. Isberg (2017). Combining thread–level speculation and just-in-time compilation in Google's V8 JavaScript engine, Concurrency and computation: practice and experience, 29(1). DOI: 10.1002/cpe.3826



Mind the gaps

Dr. Elizabeth Bjarnason

Software engineering is conceptually described as a smooth flow of progress from an iniial product idea to a delivered product. However, whether developed stage-based or iteratively, there are always gaps in the flow, where information is lost or misinterpreted. We explored these gaps and found, e.g. cognitive and geographical distances creating gaps, which we encapsulated as “a theory of distances”. This work has opened a new line of research with the International Workshop series on Requirements Engineering and Testing, and specific outcomes on test effectiveness and exploratory testing.

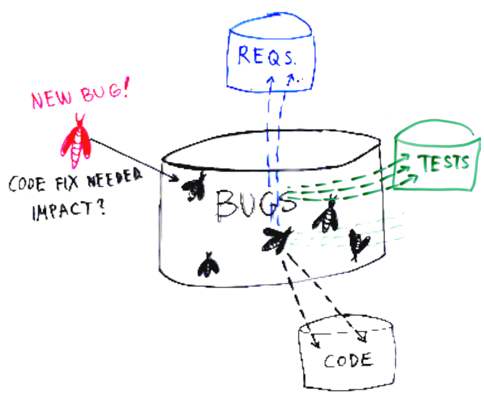


E. Bjarnason, K. Smolander, E. Engström & P. Runeson (2016). A Theory of Distances in Software Engineering, Information and Software Technology, vol. 70, p. 204-219. DOI: 10.1016/j.infsof.2015.05.004

Let the machine assign bugs

Dr. Markus Borg

Software engineering management includes many decisions, from overarching strategic ones to assigning a bug to the right team to fix. With the trend of machine learning solutions, we explored whether the machine may assist of some of the decisions. We conducted experiments on industrial large-scale data, assigning bugs to teams, that demonstrated an accuracy matching the current manual processes, and thus the company decided to adopt a version of the bug assignment into their tool set.



L. Jonsson, M. Borg, D. Broman, K. Sandahl, S. Eldh & P. Runeson (2015). Automated Bug Assignment: Ensemble-based Machine Learning in Large Scale Industrial Contexts, Empirical Software Engineering, vol. 21, no. 4, pp 1533–1578. DOI: 10.1007/s10664-015-9401-9

Align requirements and testing

Dr. Michael Unterkalmsteiner

The development of large, software-intensive systems is a complex undertaking that is generally tackled by a divide and conquer strategy. As a consequence, companies face the challenge of coordinating individual aspects of software development, in particular between requirements engineering and software testing. We approached this problem from two perspectives, testing the underlying hypothesis that information is the driving force of coordination. First, we developed and evaluated an assessment framework, that combines process and documentation information to identify coordination bottlenecks and gaps. Second, we investigated the effectiveness of information retrieval techniques to recover links between artefacts that would allow for better coordination between requirements engineers and testers.

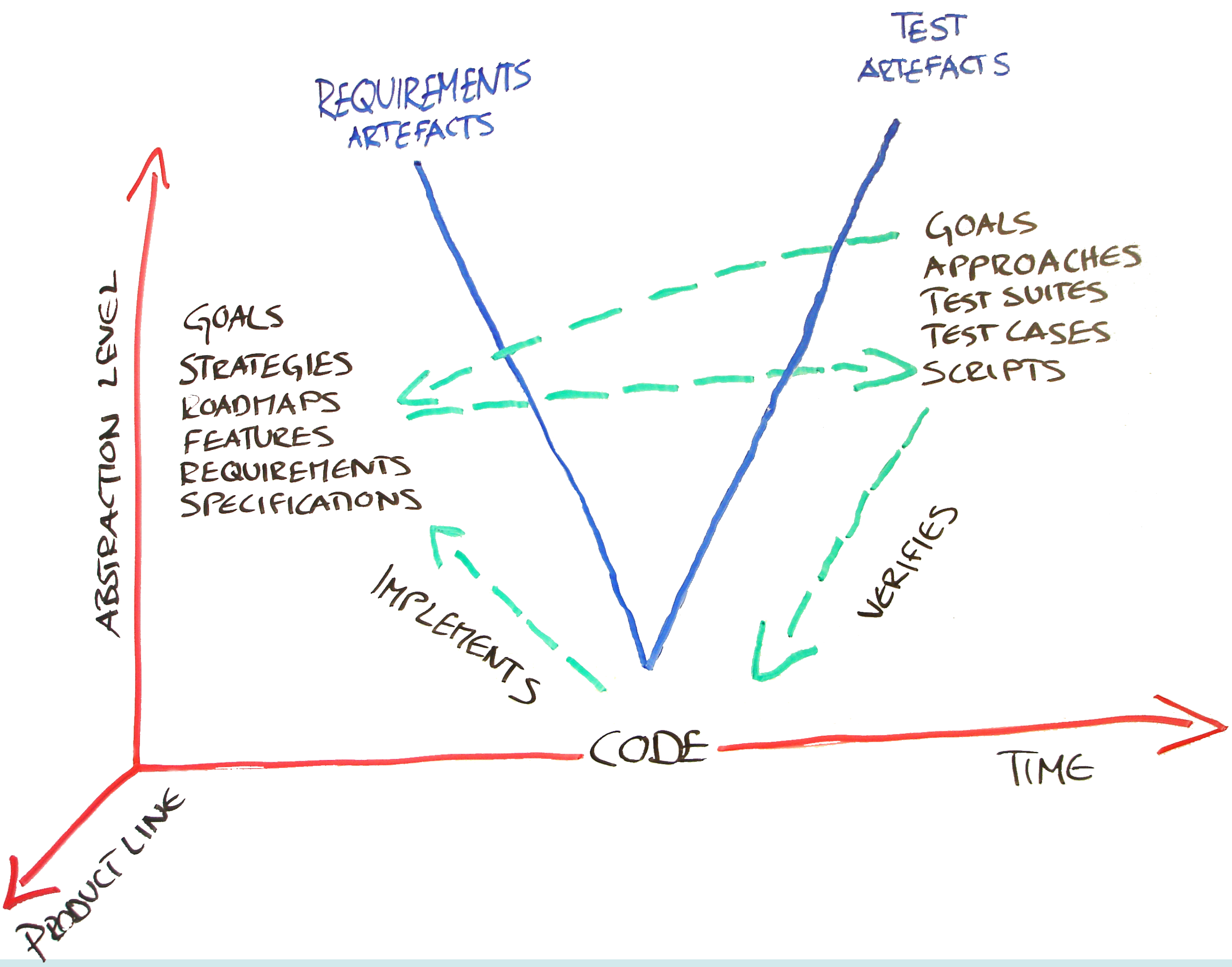
M. Unterkalmsteiner, T. Gorschek, R. Feldt & E. Klotins (2015). Assessing requirements engineering and software test alignment - Five case studies, Journal of Systems and Software, vol. 109, pp. 62-77. DOI: 10.1016/j.jss.2015.07.018

Test decision support

Drs. E. Bjarnason, E. Engström, A. N. Ghazi, N. bin Ali, Profs. P. Runeson, C. Wohlin

The time and the effort used to test large software-intense systems need to be spent wisely to minimise the risk of releasing products with serious errors and inferior functionality or quality. Companies face the challenge of making decisions regarding what, how and to which extent to perform software testing to obtain sufficient confidence in the quality of the product to release it. Through a case study at the partner companies, we identified different types of decisions made within testing both at the operational level (e.g. which test cases to run) and at the strategic level regarding what type of testing to perform (e.g. manual or automated). For the strategic decisions, we investigated the choice of performing scripted versus exploratory testing and found some factors that are affected by this. We developed a method for recommending the level and amount of exploratory testing to perform and applied to this at a company. The method provided value to the case company both in the recommendations it produced and in acting as a vehicle in creating a shared understanding of this recommendation.

A. N. Ghazi, K. Petersen, E. Bjarnason & P. Runeson (2018). Levels of Exploration in Exploratory Testing: From Freestyle to Fully Scripted. IEEE Access, vol. 6, pp. 26416-26423. 2018.2834957



EASE

Embedded Applications Software Engineering

ease.cs.lth.se

Produced 2019 by: Jonas.Wisbrant@cs.lth.se • Illustrations: Lhinn.Nilsson-Nyman@cs.lth.se and Camilla Lekebjer • Version: 1.0