Automated Software Testing: From Parasitism to Mutualism

Human-Machine Interaction

Automated software quality assurance is an activity that involves machines but is driven by human users. The human provides the machine with instructions of what and how to verify the software and the machine complies. Hence, a parasitic relationship where only the human's cognitive ability is used to solve testing tasks. Per definition, this relationship is ineffective/inefficient and results in high development, execution and evolution costs.

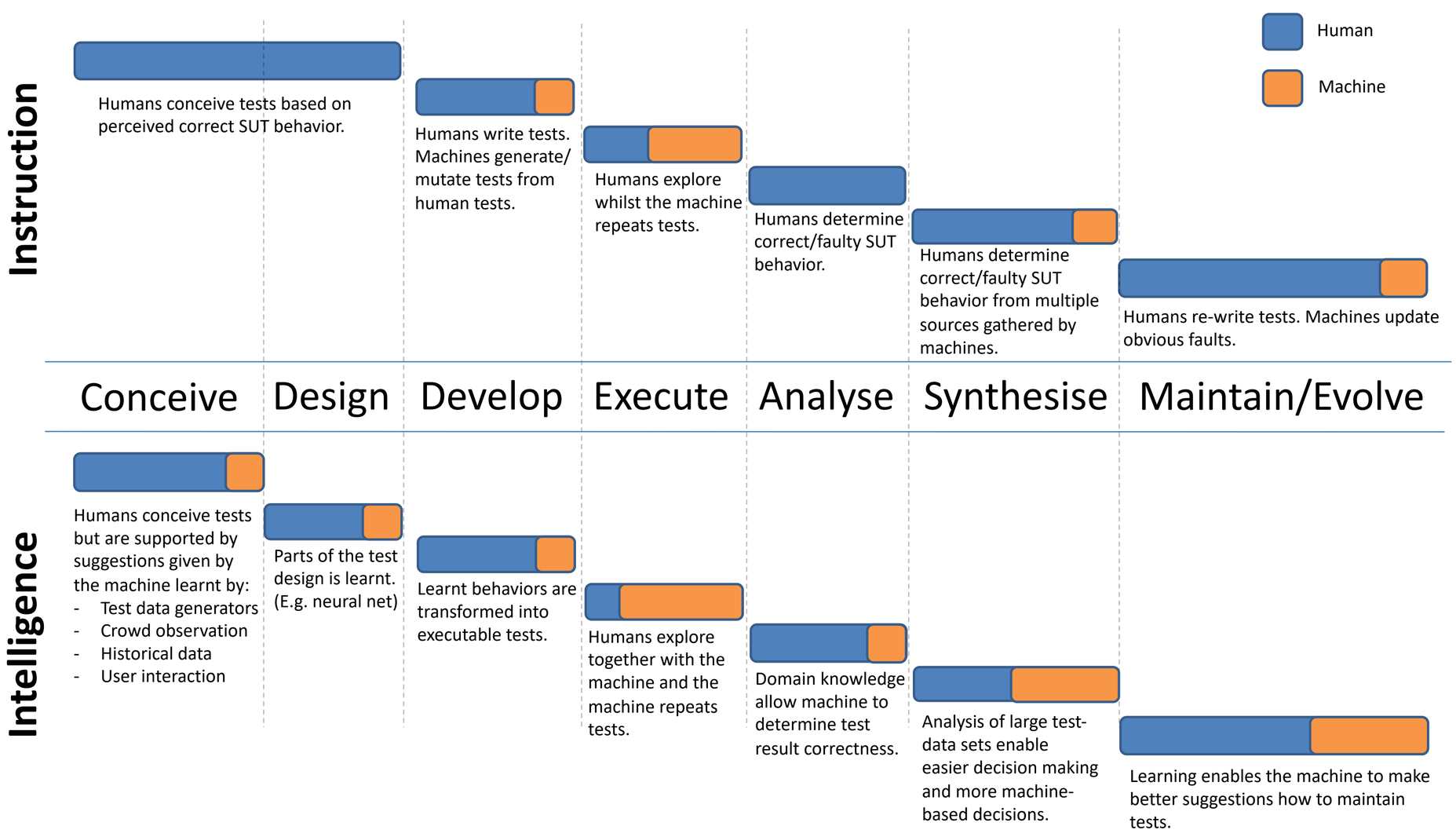
But, what if we could push the cognitive load of testing from the human user towards the machine? In this project we aim to achieve this by utilizing technical advances in machine-learning and artificial intelligence to transform human-machine interaction into human-machine mutualism where human and machine instead collaborate to solve testing tasks.

Emil Alégroth
(Sub-project Leader)
Assistant Professor
emil.alegroth@bth.se

Research Team

Robert Feldt
Michel Nass
Michael Unterkalm-steiner

Human Instruction vs Machine Intelligence



Previous experience

- E. Alégroth and R. Feldt, "*On the long-term use of visual gui testing in industrial practice: a case study*", Empirical Software Engineering, no 22, p2937-p2971, 2017.
- E. Alégroth, R. Feldt, P. Kolström, "*Maintenance of automated test suites in industry: An empirical study on Visual GUI Testing*", Information and Software Technology, no 73, p66-p80, 2016.
- E. Alégroth, M. Nass, H.H. Olsson, "*JAutomate: A tool for system- and acceptance-test automation*", 6th International conference on Software Testing, Verification and Validation, p439-p446, 2013.
- R. Feldt, R. Torkar, T. Gorschek, W. Afzel, "*Searching for cognitively diverse tests: Towards universal test diversity metrics*", Software Testing Verification and Validation Workshop, p178-p186, 2008.
- Femmer, Unterkalmsteiner, Gorschek(2017). "*Which requirements artifactquality defects are automaticallydetectable? A case study*". 4th International Workshop on Artificial Intelligence for Requirements (AIRE), pp.400-406, Lisbon, Portugal

Human-Machine Mutualism

- **WHAT:** Approaches, techniques and tools that alleviate the cognitive load of the human user (tester) by leveraging advanced technologies for test generation, visualization, usage and analysis of test results with the goal of improving test effectiveness and efficiency.
- **HOW:** Development and industrial evaluation, with a human perspective, of new, advanced, technical solutions that enable human-machine mutualism.
- **STAKEHOLDERS:** Developers and testers responsible for quality assurance of the software.

Benefits and Outcomes

- New approaches, techniques and tools that improve the effectiveness/efficiency of automated software testing
- Technical and humanistic solutions that include the human into the test environment and alleviate the human's cognitive load
- New avenues of research that can advance human-machine collaboration towards mutualism
- Evaluation/identification/mitigation of factors that affect trust in, and use of, mutualistic automated testing

SERL Sweden
LEADING SOFTWARE ENGINEERING
SOFTWARE ENGINEERING RETHOUGHT
Knowledge Foundation

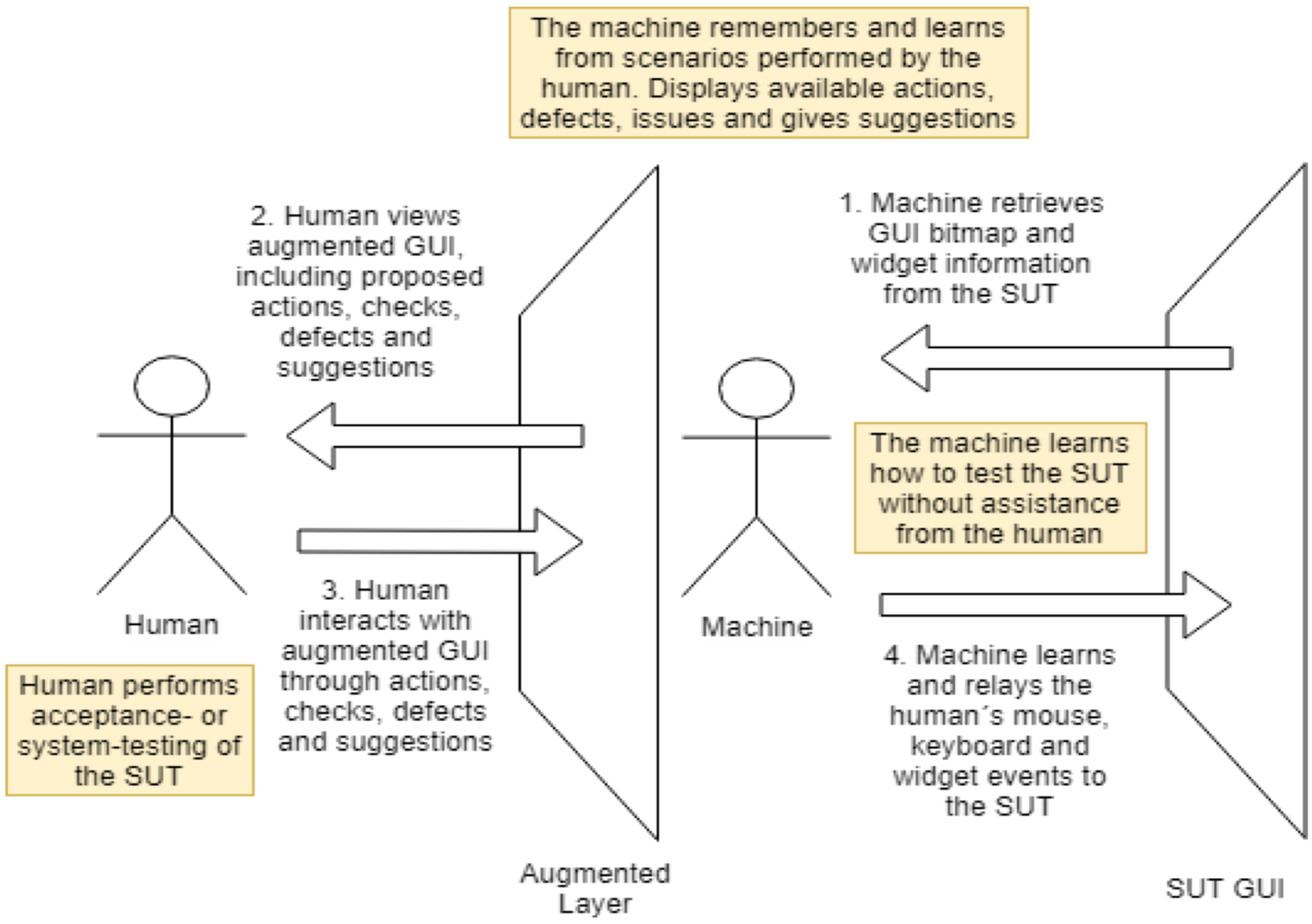
Augmented Test Automation

GUI Test Automation

Manual testing is currently the primary approach in industry to perform system- and acceptance-testing. To mitigate the problems and drawbacks with manual testing, test automation has been proposed as the solution, i.e. unit testing, record and replay, etc. However, these automation techniques have different limitations that restrict their use for system- and acceptance test automation, or make the automated tests costly to maintain. Hence, leaving a gap for a simple, high-level, test automation technique for system- and acceptance-testing.

MICHEL NASS
Projektassistent
michel.nass@bth.se

Human Machine Mutualism (HMM)



Previous experience

- Extensive industrial experience as a test consultant and educator.
- Founder and developer of tools for Visual GUI Testing, including EyeAutomate, EyeSel and EyeScout.
- Industry-Academia collaborator
- E. Alégroth, M. Nass, H.H. Olsson, "*JAutomate: A tool for system- and acceptance-test automation*", 6th International conference on Software Testing, Verification and Validation, p439-p446, 2013.

Achieving HMM in practice

- **WHAT:** A tool (machine) that learns how to use and test an application through the user interface by collaborating with the human through an augmented user interface placed between the actual system under test (SUT) and the user. The human will get feedback and suggestions from the machine through an augmented interface and the machine will learn from user's actions performed on this interface.
- **STAKEHOLDERS:** Development and empirical evaluation of new technical solutions that approach test automation through emphasis on human-machine collaboration.
- **WHO:** Developers and testers responsible for quality assurance of the software.

Outcome and Benefits

- A tool that utilizes an augmented user interface to enable ease-of-use of advanced backend technologies that are currently out-of-scope due to their technical prerequisites and requirements.
- Lower development and maintenance costs of automated testing by removing the need for scripting.
- Raised effectiveness of exploratory testing by providing test engineers with suggestions of what and how to test the system under test.
- Raised efficiency of test execution through automated replay of test scenarios captured by multiple users.
- More realistic testing through easy capture of end-user interaction with the system under test.

SERL Sweden
LEADING SOFTWARE ENGINEERING
SOFTWARE ENGINEERING RETHOUGHT
Knowledge Foundation

Tester Experience of *new, smarter or intelligent* Test Tools and Techniques

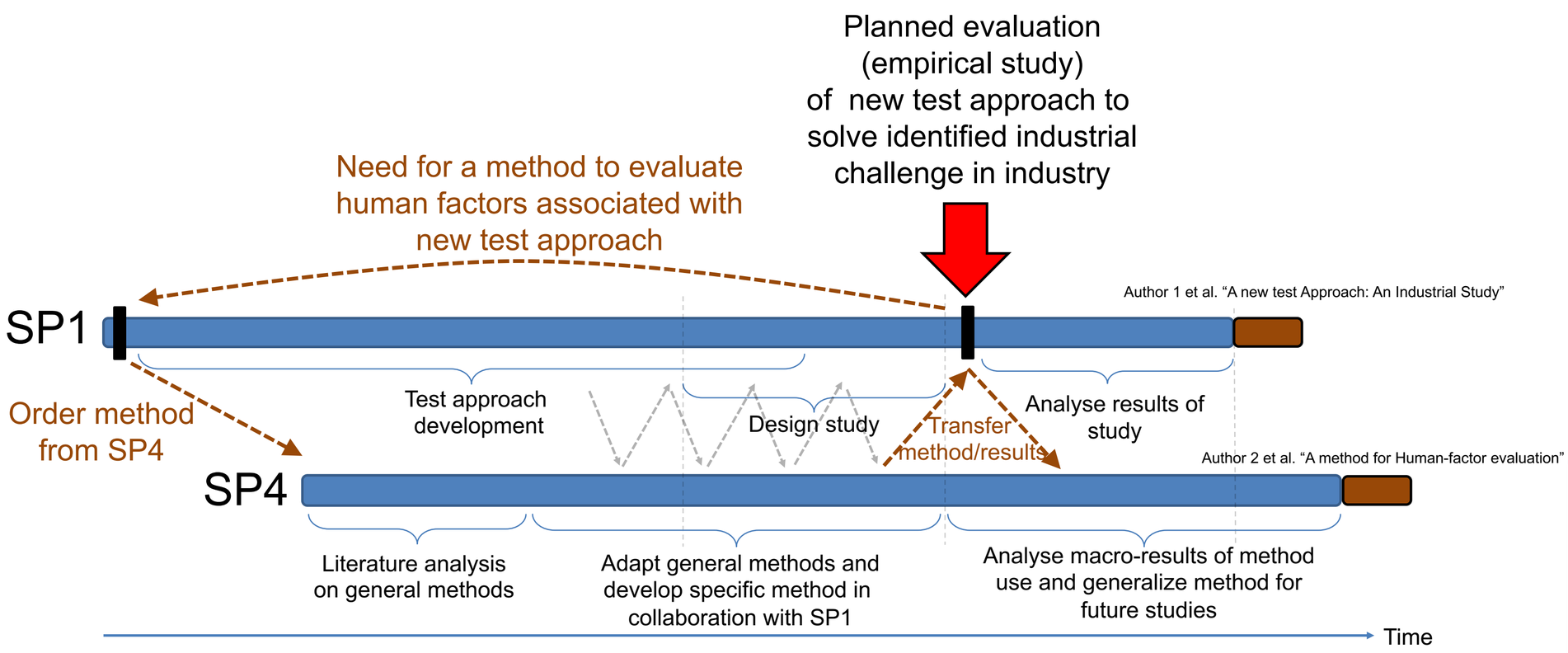
Investigating the human perspective

Automated testing brings many benefits such as improved accuracy and higher efficiency. However, most testing tasks require an element of creativity for test design, evolution or analysis. Hence, testing requires a human in the loop, a factor that is generally overlooked in both academic research and industrial practice. Instead of automating everything, it is often most beneficial to only provide automated assistance to the tester, effectively augmenting the tester with an artificial intelligence solution.

However, how do testers experience *new, smarter or intelligent* testing tools and techniques? Do they trust automatic suggestions and outcomes? Will tester efficiency increase? Sub-projects 1 and 4 jointly investigate these questions to identify the challenges, pitfalls and solutions to adoption, use and evolution of next generation, smarter, software testing tools and techniques.

Fabian Fagerholm
Associate professor
fabian.fagerholm@bth.se

Sub-project collaboration on Human factors



Emil Alégroth
Assistant Professor
emil.alegroth@bth.se

WHY: Solutions that solve industrial challenges must operate together with other technical solutions and with human users. Considering the entire environment is therefore necessary to achieve high usability, effectiveness and thereby efficiency.

Previous experience

- F. Fagerholm and J. Münch, "*Developer experience: Concept and definition*", International conference on Software and System Process, p73-p77, 2012.
- D. Graziotin, F. Fagerholm and P. Abrahamsson, "*What happens when software developers are (un)happy*", Journal of Systems and Software, no140, p32-p47, 2018.
- E. Alégroth, R. Feldt and L. Ryrholm, "*Visual GUI testing in practice: Challenges, problems and limitations*", Empirical Software Engineering, no 20, p694-744, 2015.
- E. Alégroth and R. Feldt, "*On the long-term use of visual gui testing in industrial practice: a case study*", Empirical Software Engineering, no 22, p2937-p2971, 2017.

Research Aims and Approach

- **WHAT:** Enable improved tester experience and efficiency of new testing approaches, tools and techniques.
- **HOW:** SP1 and SP4 jointly develop means to evaluate tester experience and efficiency of new testing techniques. Thus, adding a new, human-centric perspective to the evaluations.
- **STAKEHOLDERS:** Researchers and practitioners collaborating within the SERT sub-projects.
- **WHEN/HOW LONG:** 2019-2024

Expected Outcomes

- Method for evaluating tester experience and efficiency with new, smarter and intelligent test tools and techniques
- Improved tooling for next generation test automation
- Increased knowledge of how to achieve human-machine symbiosis in Software Engineering tasks

Sub-project collaboration

- The illustration also exemplifies one way how collaboration between sub-projects will take place in the SERT research Profile.

SERL Sweden
LEADING SOFTWARE ENGINEERING
SOFTWARE ENGINEERING RETHOUGHT
Knowledge Foundation