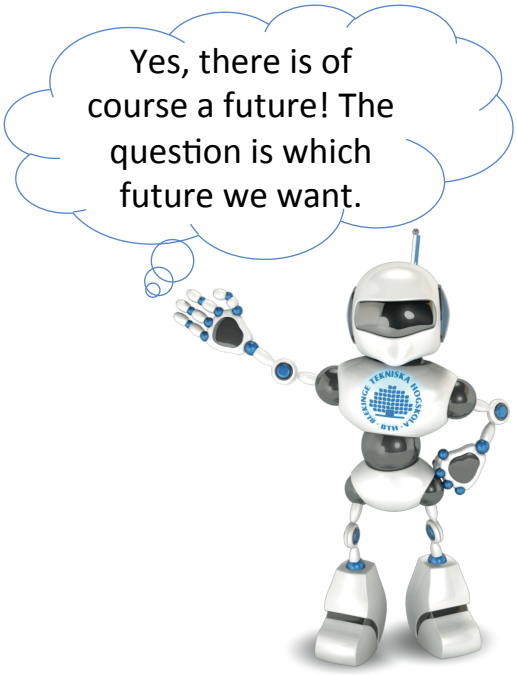
Is there a Future for Empirical Software Engineering?

Claes Wohlin, Blekinge Institute of Technology
Karlskrona, Sweden

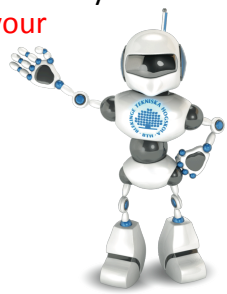


Two alternatives

What is needed for this future?

- Status quo
  - You got a paper accepted
  - And you are happy ☺
  - Your main impact on industry is through infiltration (your students)

I was happy



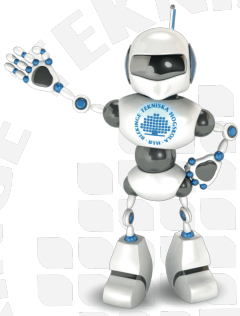
- Change and improvement
  - Your paper contributes to our joint body of knowledge
  - We can combine results from different studies to learn
  - We can better understand software engineering, including creating theories
  - We have a real impact on industry

The keyword is ”we”

- Not you or I

Back to your accepted paper ☺

1. Do you have a good title and a clear abstract?
2. Did you use the standard terminology?
3. Did you use the guidelines available for conducting the research?
4. Did you consider whether your paper would be found in a systematic literature study?
   - If so, is it easy to extract the key information from the paper?
   - Did you describe the context of the empirical work clearly?
5. Did you consider the usefulness of the paper for other researchers?
6. Did you consider whether or not the content is useful from an industrial perspective?



A terminology example

- We conducted an SLR on ”Agile practices in global software engineering” – first a database search, and then an ”independent” snowballing search. The comparison was published by Jalali and Wohlin.

Some challenges:

- Database: search terms, incl. synonyms
- Snowballing: start set of papers

Cross-Continent Development using Scrum and XP

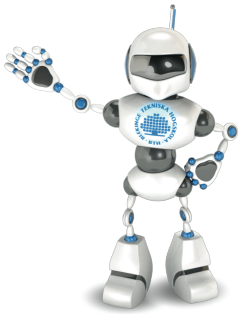
The paper is from 2003 so the authors are excused, but it is an illustrative example. We did not conduct SLRs in 2003.

- The paper was not found in the database search, since we were unable to imagine the wording
- The paper was found in snowballing, since someone else had found the paper and referred to it

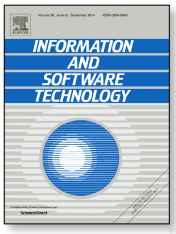
We are in my opinion not excused any longer to be too inventive when it comes to terminology.

Lessons learned

- Definitions and terms are critical
- Write clear titles and abstract (make papers findable)
- In short, write for synthesis



Two mapping studies



- Two independent papers with exactly the same title were submitted close to simultaneously to Information and Software Technology. They were reviewed in parallel and both were accepted, although the authors of the second accepted paper were asked to revise the title.

So, how similar were the papers and the findings?

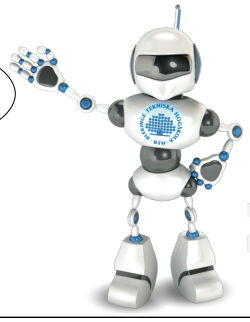
The two papers were analysed and the results published in Wohlin et al.

Comparison of papers included

Table 1
A summary of the results regarding papers in the two systematic maps.

	Brazilian/USA	Swedish	Comments
Included	45	64	Differences in years and inclusion/exclusion criteria
Common	33	33	Peer-reviewed papers published 2008 or before
Not included in other study	5	31	Papers not included may be so for two reasons:
Not found in other study	4	21	1. Papers are not found due to search strategy
Excluded in other study	1	10	2. Papers are excluded due to different judgements
Not possible to find in other study	7	0	Papers published after 2008, or non-peer-reviewed
Potentially in common	33+1	33+10	33+10+1=44

Why differences
in the common papers?



Some explanations

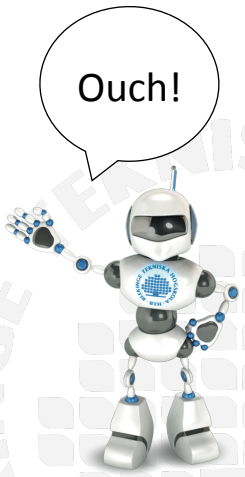
- Different phrasings of the research questions
- Different search strategies
- Different search strings -> indirectly different perceptions on software testing: Should static analysis be included?
- Different judgments

But, they research the same area.

Both studies classified the papers using the classes by Wieringa et al.

Classification of paper types

Number in A.1	Validation research	Evaluation research	Solution proposal	Philosophical papers	Opinion papers	Experience papers
1	Swedish	Brazilian	0	0	0	0
2	Brazilian, Swedish	0	0	0	0	0
3	Brazilian	0	Swedish	0	0	0
4	0	0	Brazilian, Swedish	0	0	0
5	0	0	Brazilian, Swedish	0	0	0
6	Brazilian, Swedish	0	0	0	0	0
7	Swedish	Brazilian	0	0	0	0
8	0	Swedish	Brazilian	0	0	0
9	Swedish	0	Brazilian	0	0	0
10	0	0	Brazilian, Swedish	0	0	0
11	0	Swedish	Brazilian	0	0	0
12	0	0	Brazilian, Swedish	0	0	0
13	0	0	Brazilian, Swedish	0	0	0
14	0	0	Brazilian, Swedish	0	0	0
15	0	0	Brazilian	Swedish	0	0
16	0	0	Swedish	0	Brazilian	0
17	0	0	0	0	Swedish	Brazilian
18	0	0	0	Swedish	Brazilian	0
19	0	Swedish	Brazilian	0	0	0
20	0	Swedish	Brazilian	0	0	0
21	0	Brazilian	0	Swedish	0	0
22	0	Swedish	Brazilian	0	0	0
23	0	0	Brazilian	0	Swedish	0
24	Swedish	0	Brazilian	0	0	0
25	Swedish	0	Brazilian	0	0	0
26	0	0	Brazilian, Swedish	0	0	0
27	0	0	0	Swedish	Brazilian	0
28	0	0	0	Swedish	Brazilian	0
29	0	Brazilian, Swedish	0	0	0	0
30	Swedish	0	Brazilian	0	0	0
31	0	Brazilian, Swedish	0	0	0	0
32	0	Swedish	Brazilian	0	0	0
33	0	0	Swedish	Brazilian	0	0

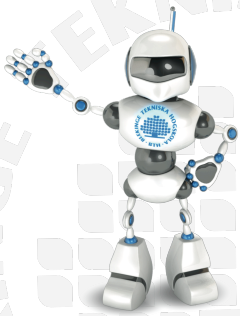


Sample in a systematic literature study

Actual population	Study population	Sample
Definition of area	Search strategy: • Where to search? • Snowballing? • Keywords in searches? • Contact authors?	Search strategy: • Construction of search string based on keywords?
Research type	Inclusion/exclusion criteria: • Focus to be included? • Level of evaluation?	Inclusion/exclusion criteria: • Procedure for individual judgment of criteria? • Procedure for combining individual judgments?
Years	Quality evaluation criteria • Thresholds?	Quality evaluation criteria • Procedure for individual judgment of criteria? • Procedure for combining individual judgments?
Language		
Publication type		

Lessons learned

- Secondary studies will not find (exactly) the same papers. They will find samples of the literature.
- Secondary studies may come to the same general conclusions regarding an area even if the papers found are not the same.
- Secondary studies are not reliable per se; they are highly dependent on the context of the secondary study, for example the area studied, definition of the area, researchers conducting the study, search approach and data available from the primary studies.



A wish from an industrial partner

Setting: Software development should be transferred from one site to another site in a different country. Time and effort for the transfer were decided by high level management.

Gut feeling: The person being responsible for the transfer was convinced about: it will take longer to transfer, and productivity and defect detection will go down more and for a longer time than anticipated (by some).

Wish: Is there evidence in the research literature on this? In other words, is there support for my gut feeling.

Our evidence search

Unfortunately, we did not find much. We found two studies:

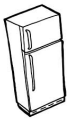
- One in banking – generally supporting the gut feeling
- One in an unknown domain – similar results, although differences in changes in productivity and defect detection

Comment from collaborator: My manager thinks we should do better than in banking and the other one is from an unknown domain. I need evidence from our domain!

Lessons learned

- We must be better in understanding potential generalizability. In this case, in particular between domains.
- We must describe the context as good as possible to ensure usefulness by others, in particular industry.





Formulate theories in software engineering

- We need an in-depth understanding of different phenomena to be able to formulate a theory.
- We need many pieces to formulate a theory.
- Preferably different researchers contribute with different pieces.

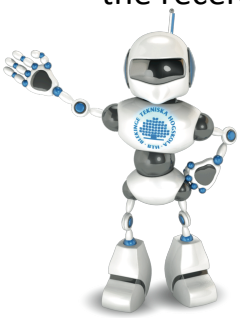
Given the challenges noted when it comes to both primary and secondary studies, and the challenges with identifying evidence, it is non-trivial to build a theory. In particular. to use ”pieces” from each other.

Example of building a theory

- Unfortunately, the example is based on ”pieces” from relatively few people.
- The theory is based on five years of research with industry, and the insights gained in these years.
- It is also based on being exposed, in collaboration with other researchers, to the concept of intellectual capital.
- Finally, it is based on a eureka moment where the insights gained from industrial collaboration (suddenly) fitted into the concepts related to intellectual capital.

Some observations from software transfers

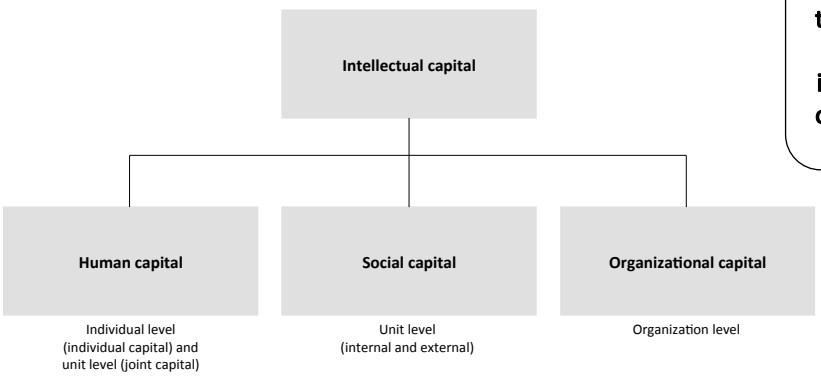
- Example 1: Product documentation was improved before a transfer.
- Example 2: Joint development between sites was organized before transferring the software product.
- Example 3: Temporary relocation of experts from the sending site to the receiving site.



So what? This sounds like regular management.

Agree, but …

Intellectual capital



Eureka: The actions conducted in the examples can be mapped to the concepts related to intellectual capital. It may seem obvious, but it certainly was not for us.

The theory

Balancing Human, Social and Organizational Capitals for Software Development and Evolution

Two tasks

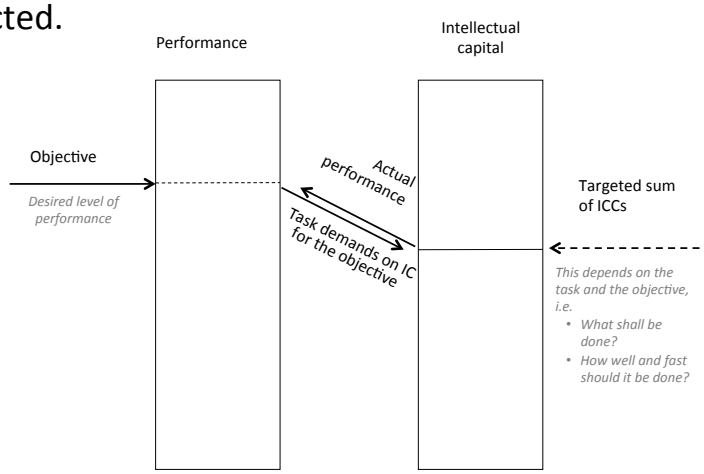
Let the size of each glass represent the difficulty to perform a task in a given situation.

Your intellectual capital

Let each bottle represent one type of intellectual capital: human, social and organizational

Fill the glasses with your mixture of capitals

- Your ”drink” is your way of ensuring that a task can be successfully conducted.



Lessons learned

- The observations were needed.
- The knowledge related to intellectual capital were needed.
- The insight of how observations and intellectual capital map to each other is needed.

These came from different persons, although collaborators.

➢ It would be preferable if we can build more easily on each other findings including publications.

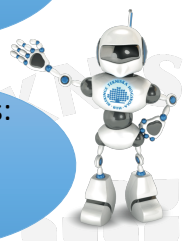
➢ Progress will be slow if we have to experience everything ourselves.

We never know when we get a eureka moment. It requires having a sufficient number of pieces of knowledge, and then … maybe …

Community needs to commit

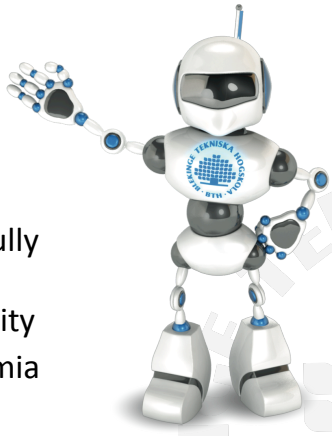
Bad news: Nobody to blame

Good news: It is in our power to improve



Ten commitments needed for the future

1. Consistent usage of terminology
2. Use accepted definitions
3. Capture and describe context
4. Use and follow guidelines
5. Write for synthesis
6. Review for synthesis
7. Conduct systematic literature studies carefully and wisely
8. Try to understand and express generalizability
9. Consider usefulness for others, both academia and industry (aim for industrial impact)
10. Look for patterns to be able to build theories



References in the presentation

- S. Jalali and C. Wohlin, "Systematic Literature Studies: Database Searches vs. Backward Snowballing", Proceedings 6th Int. Symp. on Empirical Software Engineering and Measurement (ESEM), pp. 29-38, Lund, Sweden, September 2012.
- C. Wohlin, P. Runeson, P. A. da Mota Silveira Neto, E. Engström, I. do Carmo Machado and E. S. de Almeida, "On the Reliability of Mapping Studies in Software Engineering", Journal of Systems and Software, Vol. 86, No. 10, pp. 2594-2610, 2013.
- R. Wieringa, N. Maiden, N. Mead and C. Rolland et al., ”Requirements Engineering Paper Classification and Evaluation Criteria: A Proposal and a Discussion. Requirements Engineering 11(1): 102-107. (2006).
- C. Wohlin, D. Smite and N. B. Moe, "A General Theory of Software Engineering: Balancing Human, Social and Organizational Capitals”, Journal of Systems and Software, Vol. 109, No. 11, pp. 229-242, 2015. Open access - DOI: http://dx.doi.org/10.1016/j.jss.2015.08.009

Some additional sources of inspiration can be found in the keynote abstract (ACM).

Thanks for your attention!

I am happy to take questions and discuss.