IN5140: Smart processes and agile methods in software engineering

Empirical Research Methods in Software Engineering

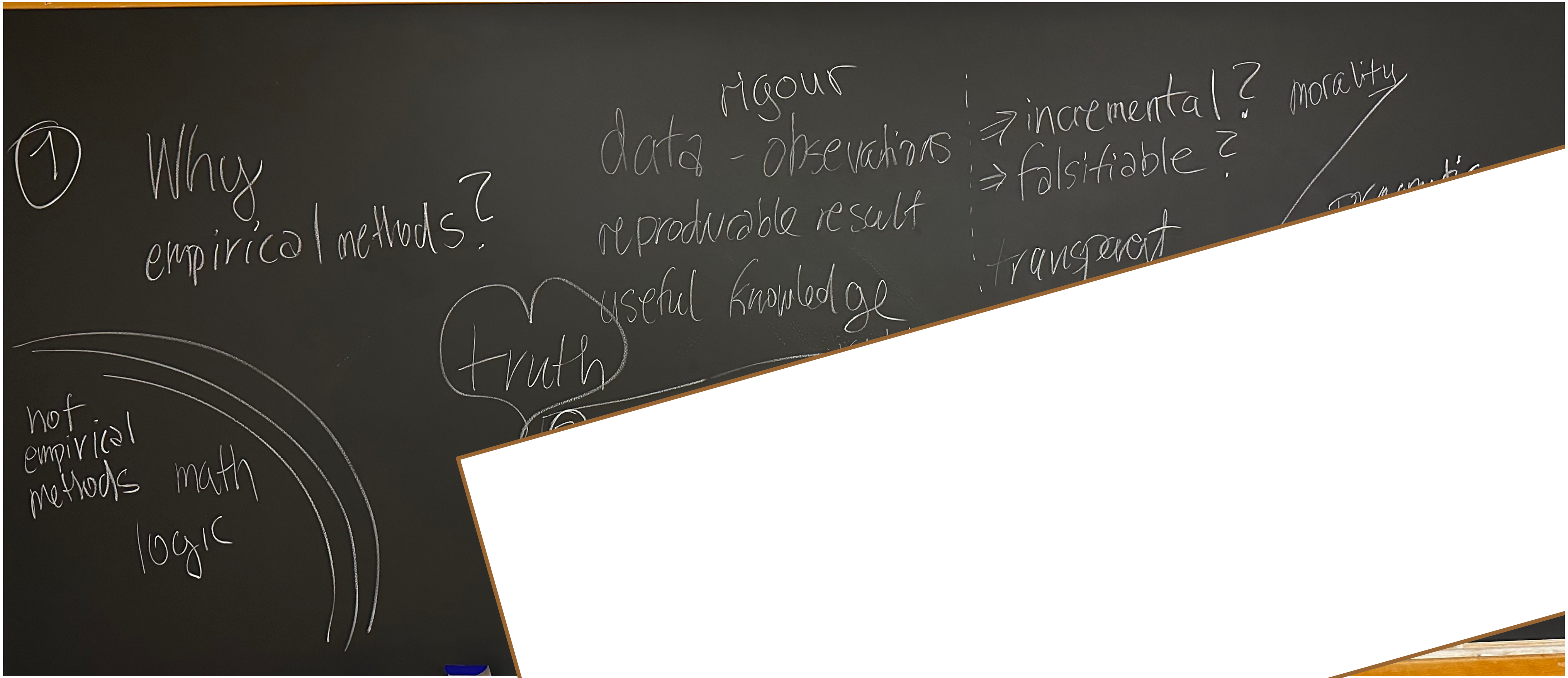
Associate professor
Gunnar Bergersen

gunnab@ifi.uio.no

25.10.2023

UiO : Department of Informatics
University of Oslo

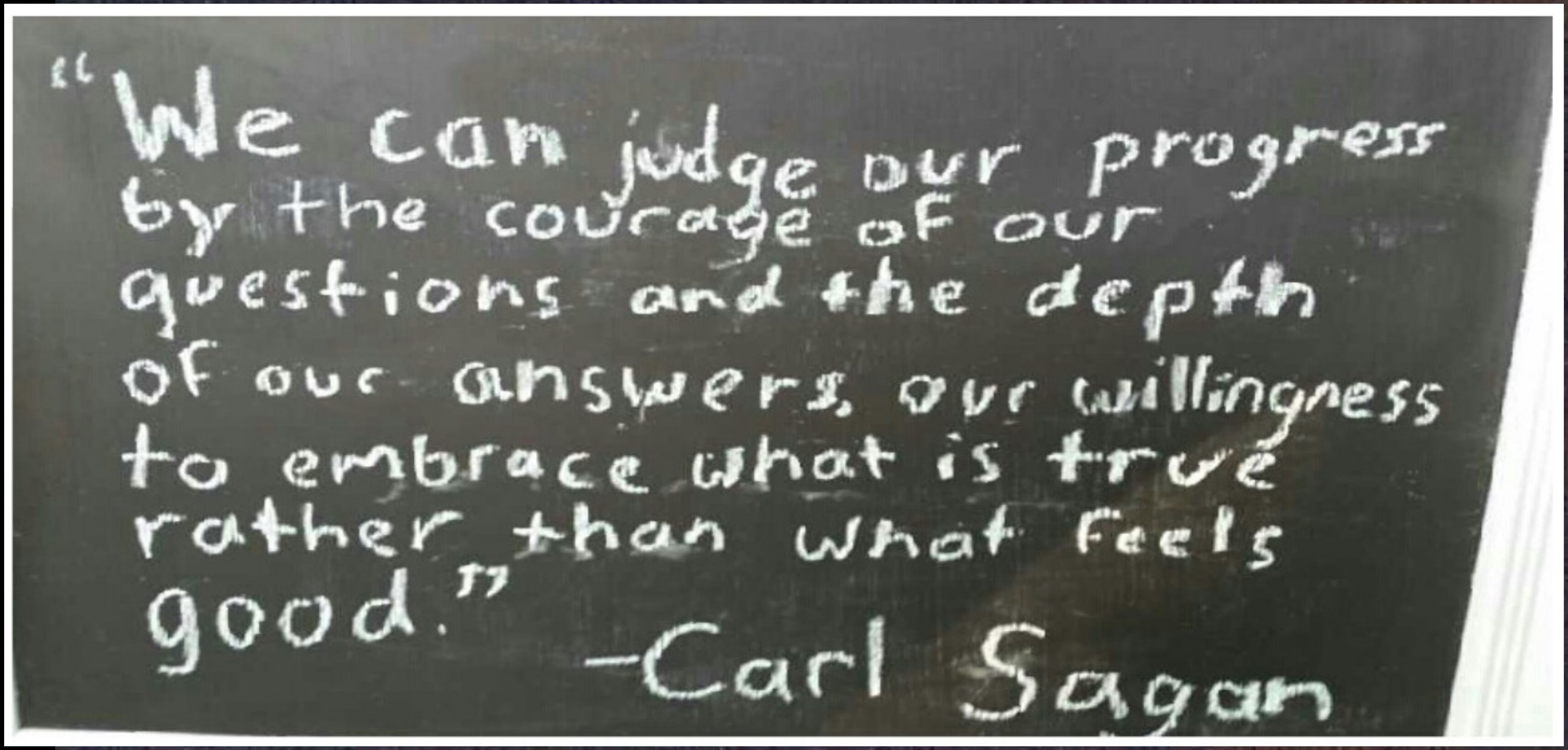
Q: Why use «empirical methods» ?


1
Why
empirical methods?
rigour
data - observations
reproducable result
useful knowledge
truth
→ incremental?
→ falsifiable?
transparent
morality
pragmatic
not empirical methods
math
logic

Truth

”Pale blue dot”

https://www.youtube.com/watch?v=GO5FwsbIpT8



"We can judge our progress
by the courage of our
questions and the depth
of our answers, our willingness
to embrace what is true
rather than what feels
good."

-Carl Sagan

”We don’t need any scientific report that tells us …”

Experience trumps research?

NRK Folkeopplysningen 5:7

strawman

Misrepresenting someone's argument to make it easier to attack.

By exaggerating, misrepresenting, or just completely fabricating someone's argument, it's much easier to present your own position as being reasonable, but this kind of dishonesty serves to undermine rational debate.

After Will said that we should put more money into health and education, Warren responded by saying that he was surprised that Will hates our country so much that he wants to leave it defenceless by cutting military spending.

false cause

Presuming that a real or perceived relationship between things means that one is the cause of the other.

Many people confuse correlation (things happening together or in sequence) for causation (that one thing actually causes the other to happen). Sometimes correlation is coincidental, or it may be attributable to a common cause.

Pointing to a fancy chart, Roger shows how temperatures have been rising over the past few centuries, whilst at the same time the numbers of pirates have been decreasing; thus pirates cool the world and global warming is a hoax.

appeal to emotion

Manipulating an emotional response in place of a valid or compelling argument.

Appeals to emotion include appeals to fear, envy, hatred, pity, guilt, and more. Though a valid, and reasoned, argument may sometimes have an emotional aspect, one must be careful that emotion doesn't obscure or replace reason.

Luke didn't want to eat his sheep's brains with chopped liver and brussels sprouts, but his father told him to think about the poor, starving children in a third world country who weren't fortunate enough to have any food at all.

the fallacy fallacy

Presuming a claim to be necessarily wrong because a fallacy has been committed.

It is entirely possible to make a claim that is false yet argue with logical coherency for that claim, just as it is possible to make a claim that is true and justify it with various fallacies and poor arguments.

Recognising that Amanda had committed a fallacy in arguing that we should eat healthy food because a nutritionist said it was popular, Alyse said we should therefore eat bacon double cheeseburgers every day.

slippery slope

Asserting that if we allow A to happen, then Z will consequently happen too, therefore A should not happen.

The problem with this reasoning is that it avoids engaging with the issue at hand, and instead shifts attention to baseless extreme hypotheticals. The merits of the original argument are then tainted by unsubstantiated conjecture.

Colin Closet asserts that if we allow same-sex couples to marry, then the next thing we know we'll be allowing people to marry their parents, their cars and even monkeys.

ad hominem

Attacking your opponent's character or personal traits in an attempt to undermine their argument.

Ad hominem attacks can take the form of overtly attacking somebody, or casting doubt on their character. The result of an ad hom attack can be to undermine someone without actually engaging with the substance of their argument.

After Sally presents an eloquent and compelling case for a more equitable taxation system, Sam asks the audience whether we should believe anything from a woman who isn't married, was once arrested, and smells a bit weird.

tu quoque

Avoiding having to engage with criticism by turning it back on the accuser - answering criticism with criticism.

Literally translating as 'you too' this fallacy is commonly employed as an effective red herring because it takes the heat off the accused having to defend themselves and shifts the focus back onto the accuser themselves.

Nicole identified that Hannah had committed a logical fallacy, but instead of addressing the substance of her claim, Hannah accused Nicole of committing a fallacy earlier on in the conversation.

personal incredulity

Saying that because one finds something difficult to understand, it's therefore not true.

Subjects such as biological evolution via the process of natural selection require a good amount of understanding before one is able to properly grasp them; this fallacy is usually used in place of that understanding.

Kirk drew a picture of a fish and a human and with effusive disdain asked Richard if he really thought we were stupid enough to believe that a fish somehow turned into a human through just, like, random things happening over time.

special pleading

Moving the goalposts or making up exceptions when a claim is shown to be false.

Humans are funny creatures and have a foolish aversion to being wrong. Rather than appreciate the benefits of being able to change one's mind through better understanding, many will invent ways to cling to old beliefs.

Edward Johns claimed to be psychic, but when his 'abilities' were tested under proper scientific conditions, they magically disappeared. Edward explained this saying that one had to have faith in his abilities for them to work.

loaded question

Asking a question that has an assumption built into it so that it can't be answered without appearing guilty.

Loaded question fallacies are particularly effective at derailing rational debates because of their inflammatory nature - recipients of a loaded question are compelled to defend themselves and may appear flustered or on the back foot.

Grace and Helen were both romantically interested in Brad. One day, with Brad sitting within earshot, Grace asked in an inquisitive tone whether Helen was having any problems with a fungal infection.

burden of proof

Saying that the burden of proof lies not with the person making the claim, but with someone else to disprove.

The burden of proof lies with someone who is making a claim, and is not upon anyone else to disprove. The inability, or disinclination, to disprove a claim does not make it valid (however we must always go by the best available evidence).

Bertrand declares that a teapot is, at this very moment, in orbit around the Sun between the Earth and Mars, and that because no one can prove him wrong his claim is therefore a valid one.

ambiguity

Using double meanings or ambiguities of language to mislead or misrepresent the truth.

Politicians are often guilty of using ambiguity to mislead and will later point to how they were technically not outright lying if they come under scrutiny. It's a particularly tricky and premeditated fallacy to commit.

When the judge asked the defendant why he hadn't paid his parking fines, he said that he shouldn't have to pay them because the sign said 'Fine for parking here' and so he naturally presumed that it would be fine to park there.

the gambler's fallacy

Believing that 'runs' occur to statistically independent phenomena such as roulette wheel spins.

This commonly believed fallacy can be said to have helped create a city in the desert of Nevada USA. Though the overall odds of a 'big run' happening may be low, each spin of the wheel is itself entirely independent from the last.

Red had come up six times in a row on the roulette wheel, so Greg knew that it was close to certain that black would be next up. Suffering an economic form of natural selection with this thinking, he soon lost all of his savings.

bandwagon

Appealing to popularity or the fact that many people do something as an attempted form of validation.

The flaw in this argument is that the popularity of an idea has absolutely no bearing on its validity. If it did, then the Earth would have made itself flat for most of history to accommodate this popular belief.

Shamus pointed a drunken finger at Sean and asked him to explain how so many people could believe in leprechauns if they're only a silly old superstition. Sean, however, had had a few too many Guinness himself and fell off his chair.

appeal to authority

Saying that because an authority thinks something, it must therefore be true.

It's important to note that this fallacy should not be used to dismiss the claims of experts, or scientific consensus. Appeals to authority are not valid arguments, but nor is it reasonable to disregard the claims of experts who have a demonstrated depth of knowledge unless one has a similar level of understanding.

Not able to defend his position that evolution 'isn't true' Bob says that he knows a scientist who also questions evolution (and presumably isn't herself a primate).

composition /division

Assuming that what's true about one part of something has to be applied to all, or other, parts of it.

Often when something is true for the part it does also apply to the whole, but because this isn't always the case it can't be presumed to be true. We must show evidence for why a consistency will exist.

Daniel was a precocious child and had a liking for logic. He reasoned that atoms are invisible, and that he was made of atoms and therefore invisible too. Unfortunately, despite his thinky skills, he lost the game of hide and go seek.

no true scotsman

Making what could be called an appeal to purity as a way to dismiss relevant criticisms or flaws of an argument.

This fallacy is often employed as a measure of last resort when a point has been lost. Seeing that a criticism is valid, yet not wanting to admit it, new criteria are invoked to dissociate oneself or one's argument.

Angus declares that Scotsmen do not put sugar on their porridge, to which Lachlan points out that he is a Scotsman and puts sugar on his porridge. Furious, like a true Scot, Angus yells that no **true** Scotsman sugars his porridge.

genetic

Judging something good or bad on the basis of where it comes from, or from whom it comes.

To appeal to prejudices surrounding something's origin is another red herring fallacy. This fallacy has the same function as an ad hominem, but applies instead to perceptions surrounding something's source or context.

Accused on the 6 o'clock news of corruption and taking bribes, the senator said that we should all be very wary of the things we hear in the media, because we all know how very unreliable the media can be.

black-or-white

Where two alternative states are presented as the only possibilities, when in fact more possibilities exist.

Also known as the false dilemma, this insidious tactic has the appearance of forming a logical argument, but under closer scrutiny it becomes evident that there are more possibilities than the either/or choice that is presented.

Whilst rallying support for his plan to fundamentally undermine citizens' rights, the Supreme Leader told the people they were either on his side, or on the side of the enemy.

begging the question

A circular argument in which the conclusion is included in the premise.

This logically incoherent argument often arises in situations where people have an assumption that is very ingrained, and therefore taken in their minds as a given. Circular reasoning is bad mostly because it's not very good.

The word of Zorbo the Great is flawless and perfect. We know this because it says so in The Great and Infallible Book of Zorbo's Best and Most Truest Things that are Definitely True and Should Not Ever Be Questioned.

appeal to nature

Making the argument that because something is 'natural' it is therefore valid, justified, inevitable, good, or ideal.

Many 'natural' things are also considered 'good', and this can bias our thinking; but naturalness itself doesn't make something good or bad. For instance murder could be seen as very natural, but that doesn't mean it's justifiable.

The medicine man rolled into town on his bandwagon offering various natural remedies, such as very special plain water. He said that it was only natural that people should be wary of 'artificial' medicines like antibiotics.

anecdotal

Using personal experience or an isolated example instead of a valid argument, especially to dismiss statistics.

It's often much easier for people to believe someone's testimony as opposed to understanding variation across a continuum. Scientific and statistical measures are almost always more accurate than individual perceptions and experiences.

Jason said that that was all cool and everything, but his grandfather smoked, like, 30 cigarettes a day and lived until 97 - so don't believe everything you read about meta analyses of sound studies showing proven causal relationships.

the texas sharpshooter

Cherry-picking data clusters to suit an argument, or finding a pattern to fit a presumption.

This 'false cause' fallacy is coined after a marksman shooting at barns and then painting a bullseye target around the spot where the most bullet holes appear. Clusters naturally appear by chance, and don't necessarily indicate causation.

The makers of Sugarette Candy Drinks point to research showing that of the five countries where Sugarette drinks sell the most units, three of them are in the top ten healthiest countries on Earth, therefore Sugarette drinks are healthy.

middle ground

Saying that a compromise, or middle point, between two extremes must be the truth.

Much of the time the truth does indeed lie between two extreme points, but this can bias our thinking: sometimes a thing is simply untrue and a compromise of it is also untrue. Half way between truth and a lie, is still a lie.

Holly said that vaccinations caused autism in children, but her scientifically well-read friend Caleb said that this claim had been debunked and proven false. Their friend Alice offered a compromise that vaccinations cause some autism.

thou shalt not commit logical fallacies

A logical fallacy is a flaw in reasoning. Strong arguments are void of logical fallacies, whilst arguments that are weak tend to use logical fallacies to appear stronger than they are. They're like tricks or illusions of thought, and they're often very sneakily used by politicians, the media, and others to fool people. Don't be fooled! This poster has been designed to help you identify and call out dodgy logic wherever it may raise its ugly, incoherent head. If you see someone committing a logical fallacy online, link them to the relevant fallacy to school them in thinkiness e.g. yourlogicalfallacyis.com/strawman

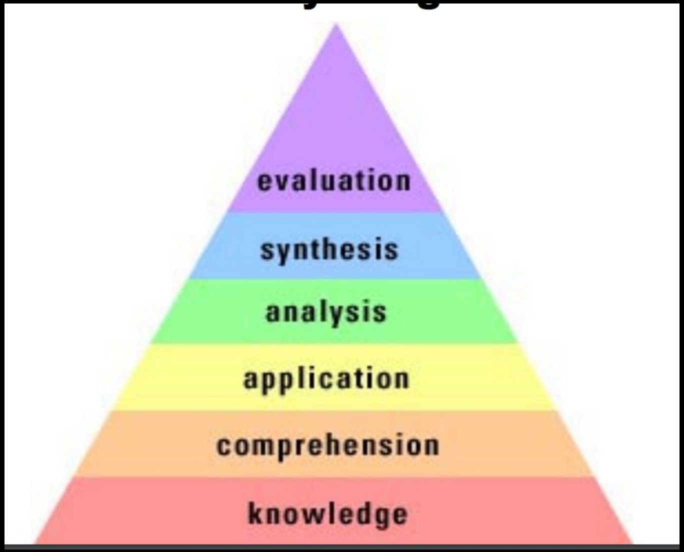
You can download this poster for free at yourlogicalfallacyis.com/poster

About me

- Current:
  - Associate professor at Software Engineering
- Education
  - MSc (2001) and PhD (2015) from UiO
  - PhD thesis: “Measuring programming skill”
- Prior work experience
  - Programmer
  - IT Project leader two companies
  - CEO three companies
  - Startup based on my PhD

Golonka et al (2023): the construct of cuteness https://doi.org/10.3389/fpsyg.2023.1068373

Learning objectives



After this lecture, you should be able to …

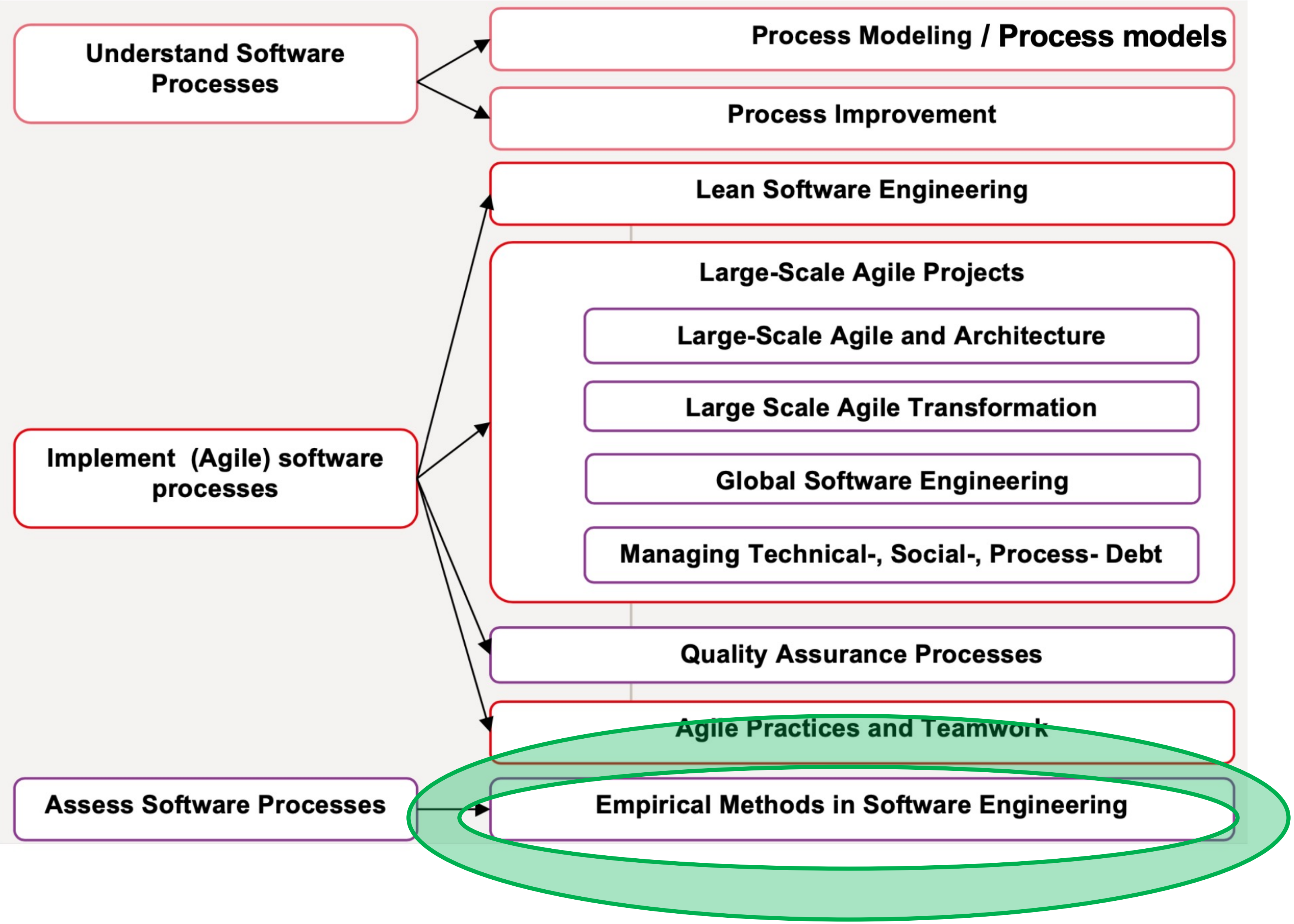
- *Describe* the central elements of e**mpirical research** and
  - *Explain* the steps involved in **Evidence-based software engineering** and provide *critique* of claims that based on use of theory and empirical results
  - *Discuss* the strengths and weaknesses of different **empirical research methods** and *suggest* what method(s) to use for a specific situation
  - *Identify* how changes in **context** may affect the answer to a research question

Structure

- Empirical research
  - Evidence-based software engineering
  - Empirical research methods
    - Validity
  - The importance of context

Note: I only present a subset of the slides you will have to study

Context



Understand Software Processes

Process Modeling / Process models

Process Improvement

Implement (Agile) software processes

Lean Software Engineering

Large-Scale Agile Projects

Large-Scale Agile and Architecture

Large Scale Agile Transformation

Global Software Engineering

Managing Technical-, Social-, Process- Debt

Quality Assurance Processes

Agile Practices and Teamwork

Assess Software Processes

Empirical Methods in Software Engineering

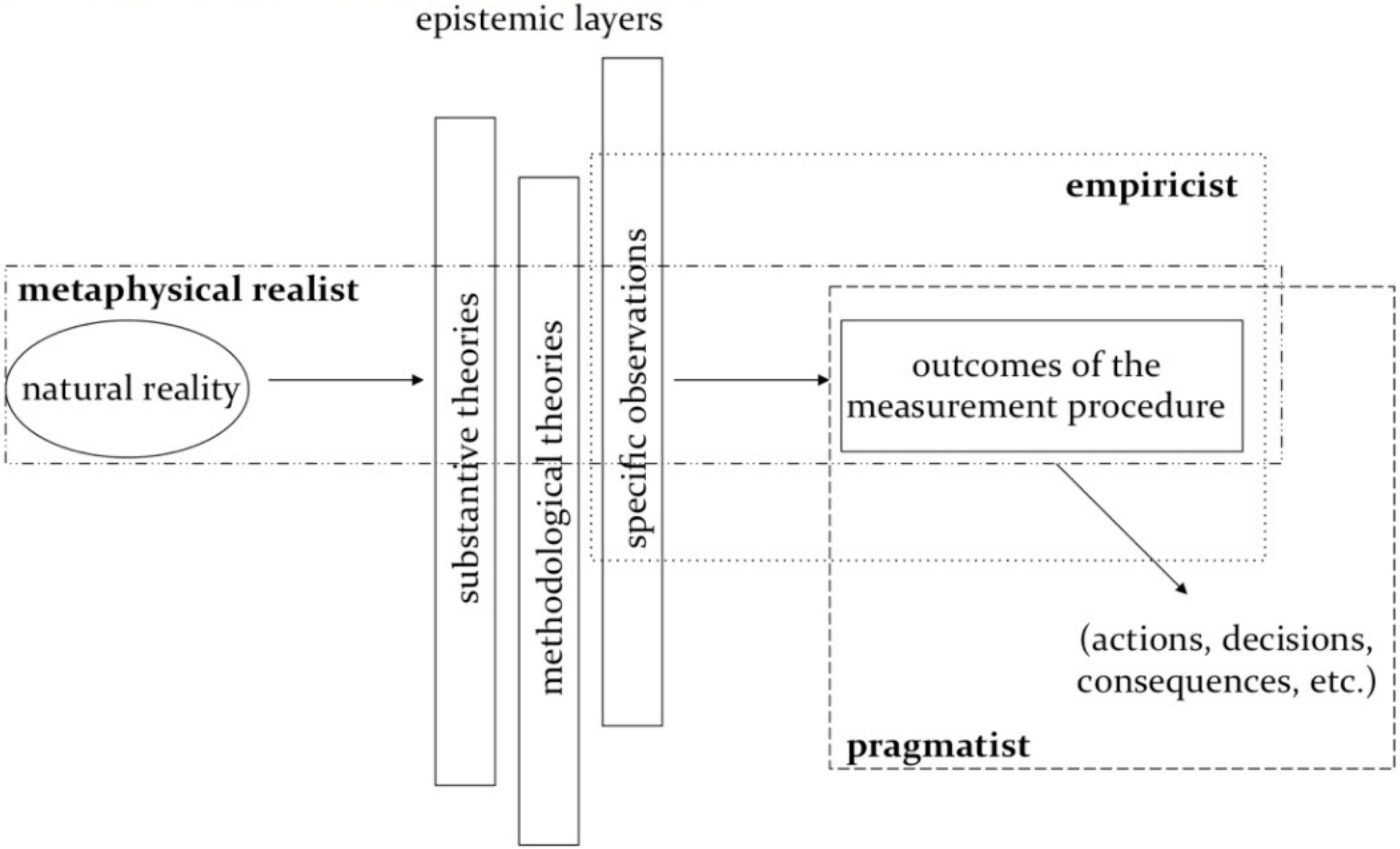
Empirical research

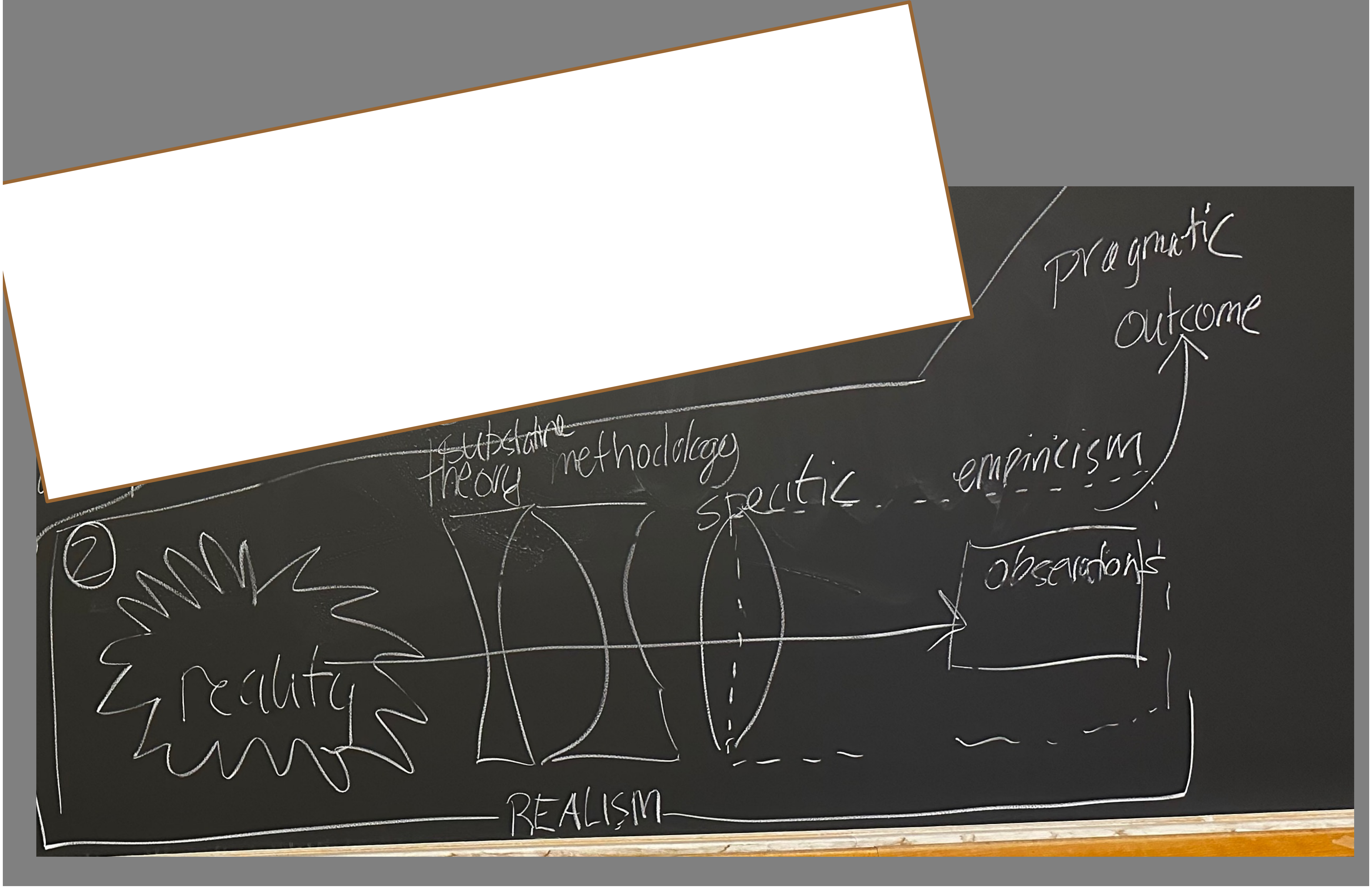
- Empirical research concerns the acquisition of knowledge by empirical methods
- Empirical research seeks to explore, describe, predict, and explain natural, social, or cognitive phenomena by using evidence based on observation or experience

- What constitutes knowledge, and the methods for acquiring it, rests on basic assumptions regarding:
  - Ontology, i.e., what we believe to exist,
  - Epistemology i.e., how beliefs are acquired and what justifies them,
  - Methodology, e.g., the inductive or the hypothetico-deductive method.

Maul et al. 2013, On the conceptual foundations of psychological measurement, Journal of physics, vol. 459(1), p. 012008,
doi: 10.1088/1742-6596/459/1/012008

Figure 1: A pragmatic-realist view of measurement

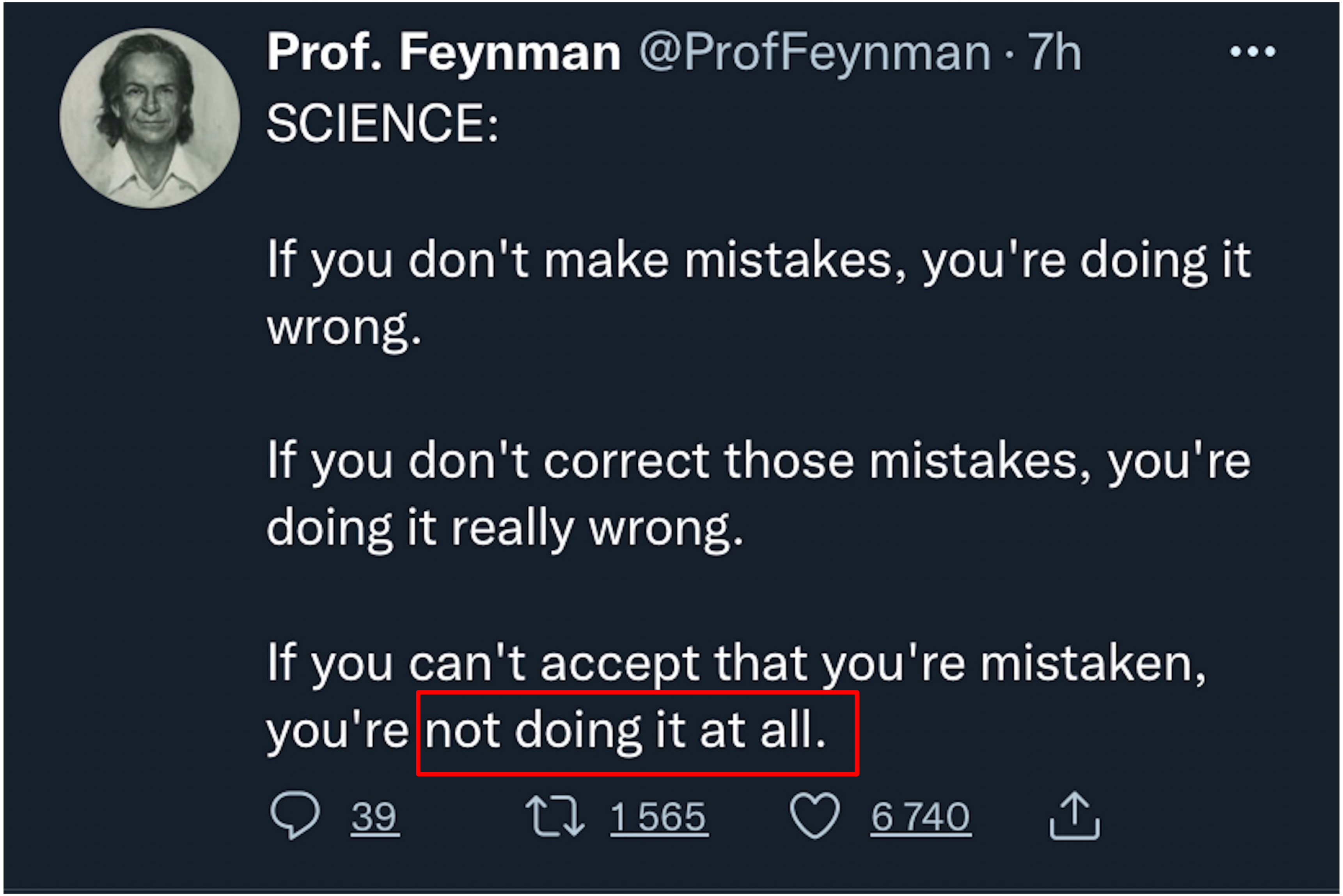



pragmatic
outcome
substatre
theory
methodology
specific
empiricism
2
reality
observations
REALISM

Empirical evidence

- Empirical evidence is the data on which a conclusion or judgment may be based.
- Interpreting and judging such evidence depends on the "eye of the observer".
- Much research applies to groups of individuals or populations, and are not always relevant or valid for other situations.
- Accurate prediction or absolute proof of causality applicable to individuals or to real-life settings are virtually impossible.
- The contributions of empirical research to any situation depend on the context, judgment and values, understanding of probability, and tolerance for uncertainty.

Falsifiability (and opening up for the possibility of being wrong, c.f. “ontology”)



Prof. Feynman @ProfFeynman · 7h

SCIENCE:

If you don't make mistakes, you're doing it wrong.

If you don't correct those mistakes, you're doing it really wrong.

If you can't accept that you're mistaken, you're not doing it at all.

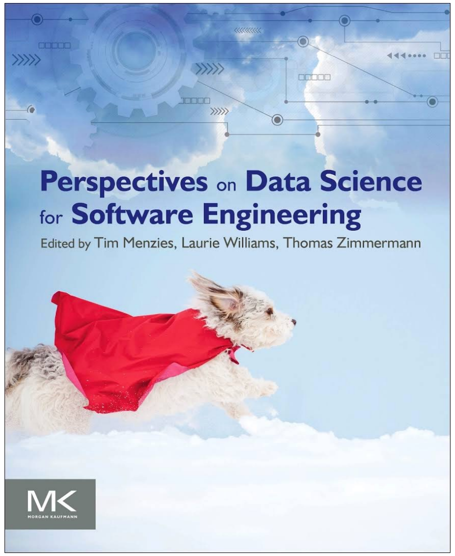
What is SE practice based on?

- Mostly, the SE discipline is based on a combination of human authority and anecdotal experience:*
  - We know that a particular technique is good because John Doe, who is an authority in the field, says that it is good (human authority); and that
  - John Doe knows that it is good because it worked for him (anecdotal experience).

*C. Michael Holloway, Software Engineering and Epistemology, *Software Engineering Notes*, 1995, 20(2): 20-21.

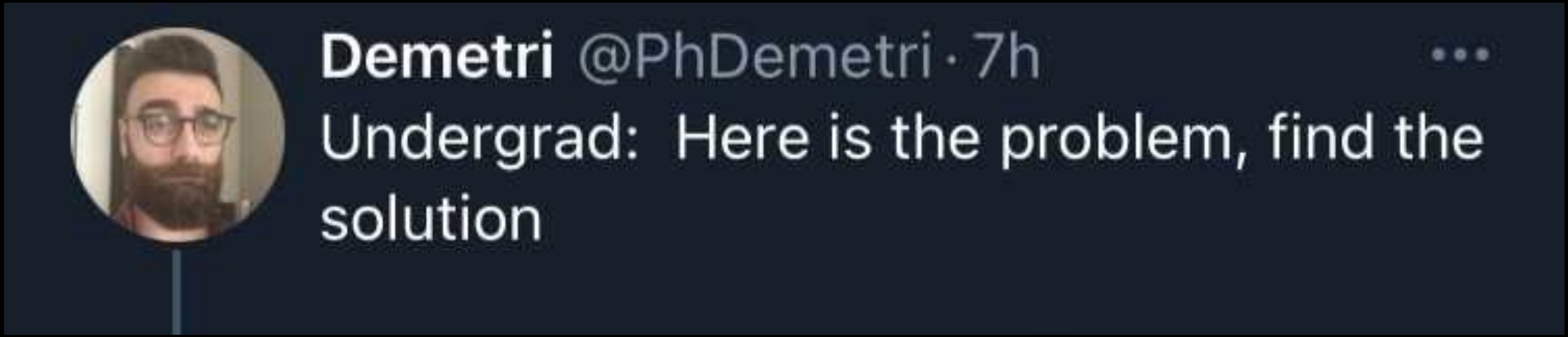
Current problems of software development

- The prevalence of fads more typical of the fashion industry than an engineering discipline.
- The lack of a sound, widely accepted theoretical basis.
- The huge number of methods and method variants, with differences little understood and artificially magnified.


Perspectives on Data Science for Software Engineering
Edited by Tim Menzies, Laurie Williams, Thomas Zimmermann
MK

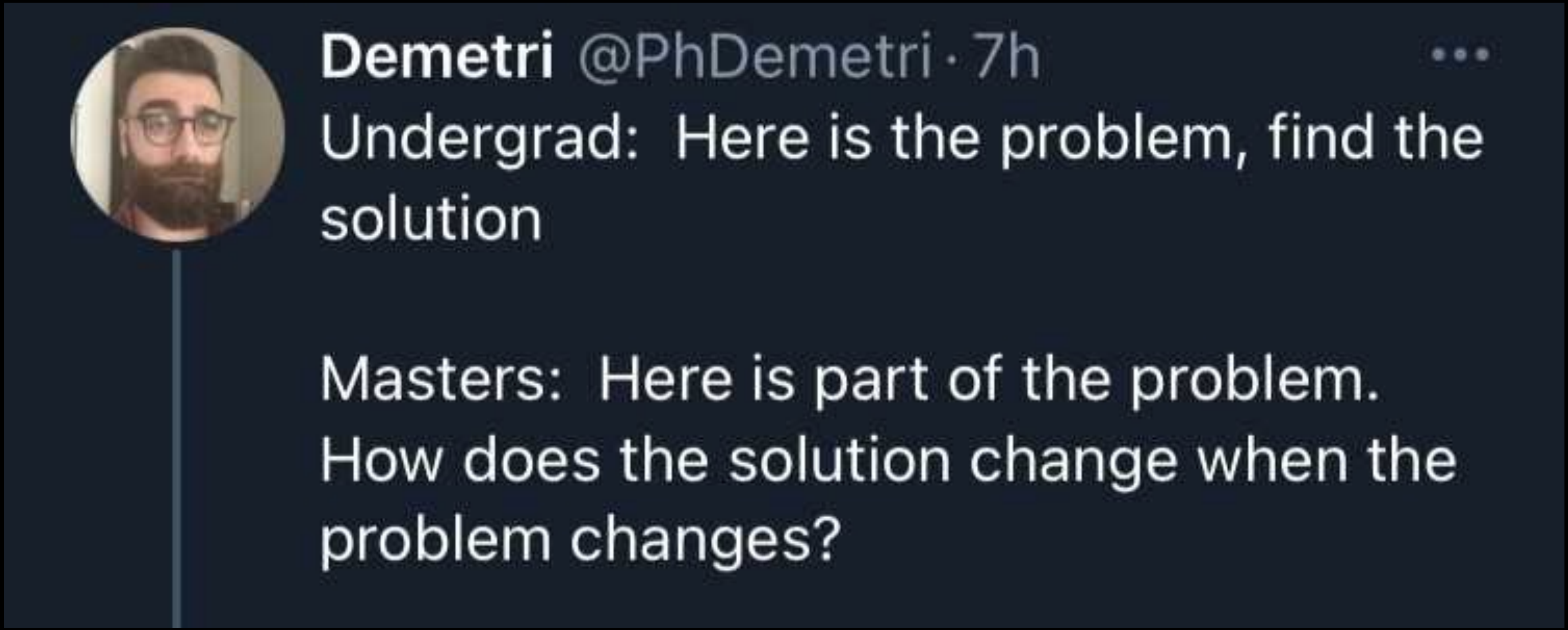
- The lack of credible empirical evaluation and validation.
- The split between (software) industry and academia.

Jacobson, Ng, McMahon, Spence, and Lidman *The Essence of Software Engineering*, Addison-Wesley, 2013.



Demetri @PhDemetri · 7h

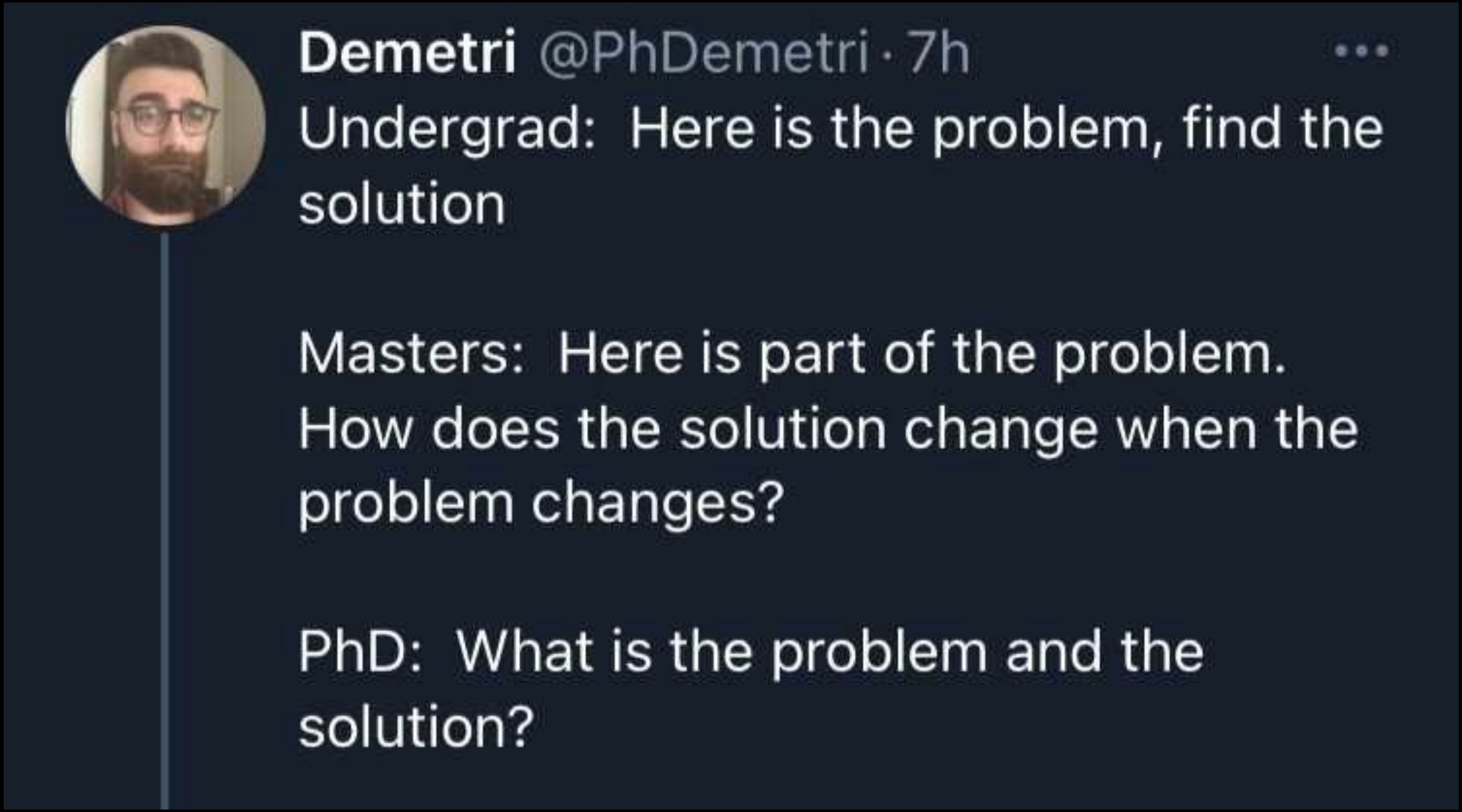
Undergrad: Here is the problem, find the solution



Demetri @PhDemetri · 7h

Undergrad: Here is the problem, find the solution

Masters: Here is part of the problem. How does the solution change when the problem changes?

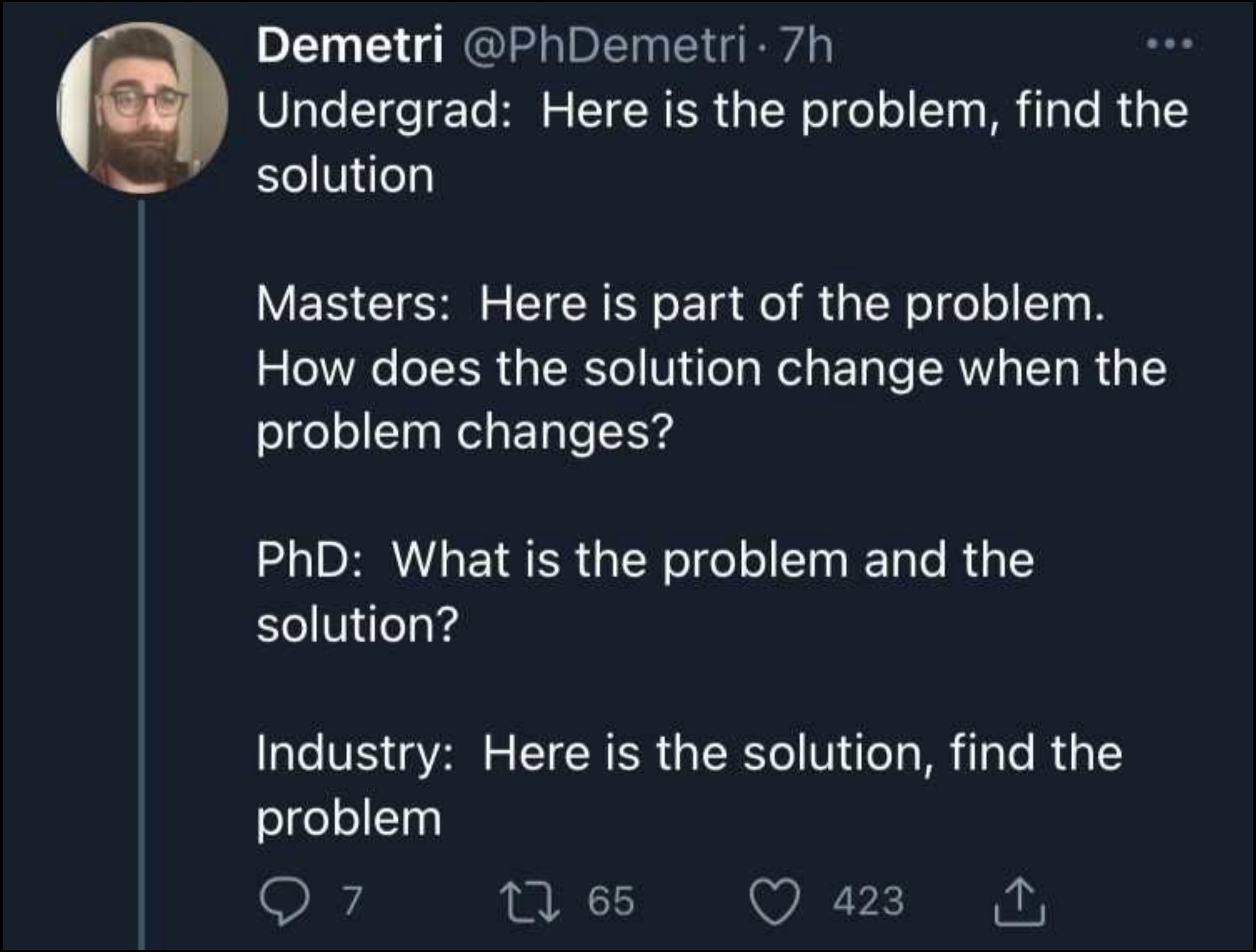


Demetri @PhDemetri · 7h

Undergrad: Here is the problem, find the solution

Masters: Here is part of the problem. How does the solution change when the problem changes?

PhD: What is the problem and the solution?



Demetri @PhDemetri · 7h

Undergrad: Here is the problem, find the solution

Masters: Here is part of the problem. How does the solution change when the problem changes?

PhD: What is the problem and the solution?

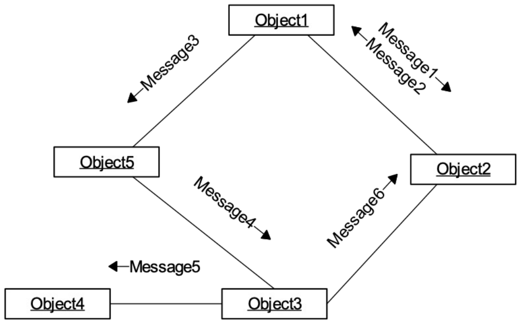
Industry: Here is the solution, find the problem

Delegated vs. centralized control – expert opinions

- **The Delegated Control Style:**
  - *Rebecca Wirfs-Brock*: A delegated control style ideally has clusters of well defined responsibilities distributed among a number of objects. To me, a delegated control architecture feels like object design at its best…
  - *Alistair Cockburn*: [The delegated coffee-machine design] is, I am happy to see, robust with respect to change, and it is a much more reasonable "model of the world."”
- **The Centralized Control Style:**
  - *Rebecca Wirfs-Brock*: A centralized control style is characterized by single points of control interacting with many simple objects. To me, centralized control feels like a "procedural solution" cloaked in objects…
  - *Alistair Cockburn*: Any oversight in the “mainframe” object (even a typo!) [in the centralized coffee-machine design] means potential damage to many modules, with endless testing and unpredictable bugs.

Responsibility Driven Design

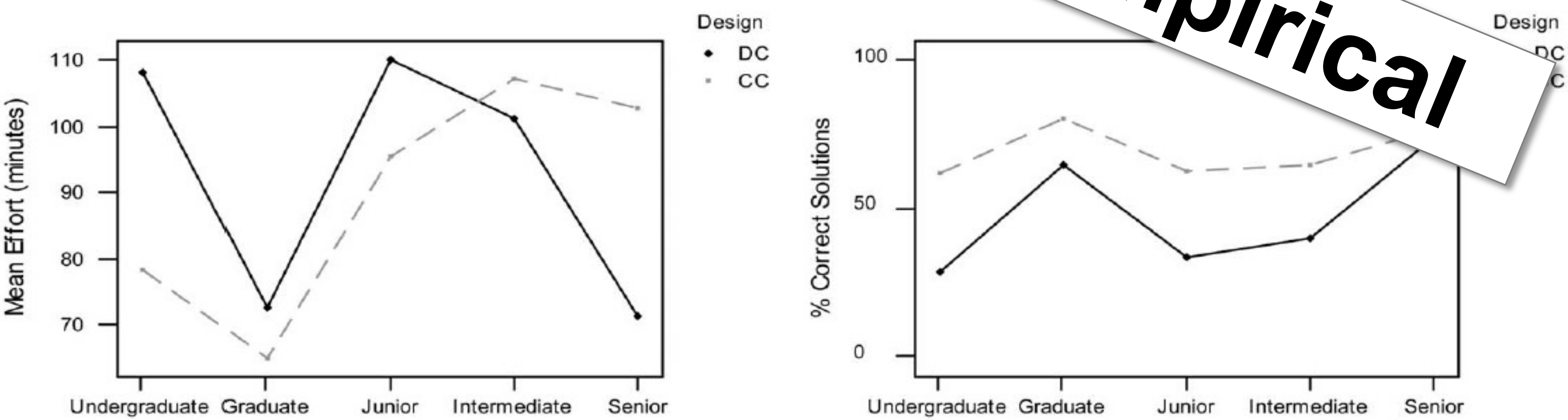
Role Modelling



Analytical

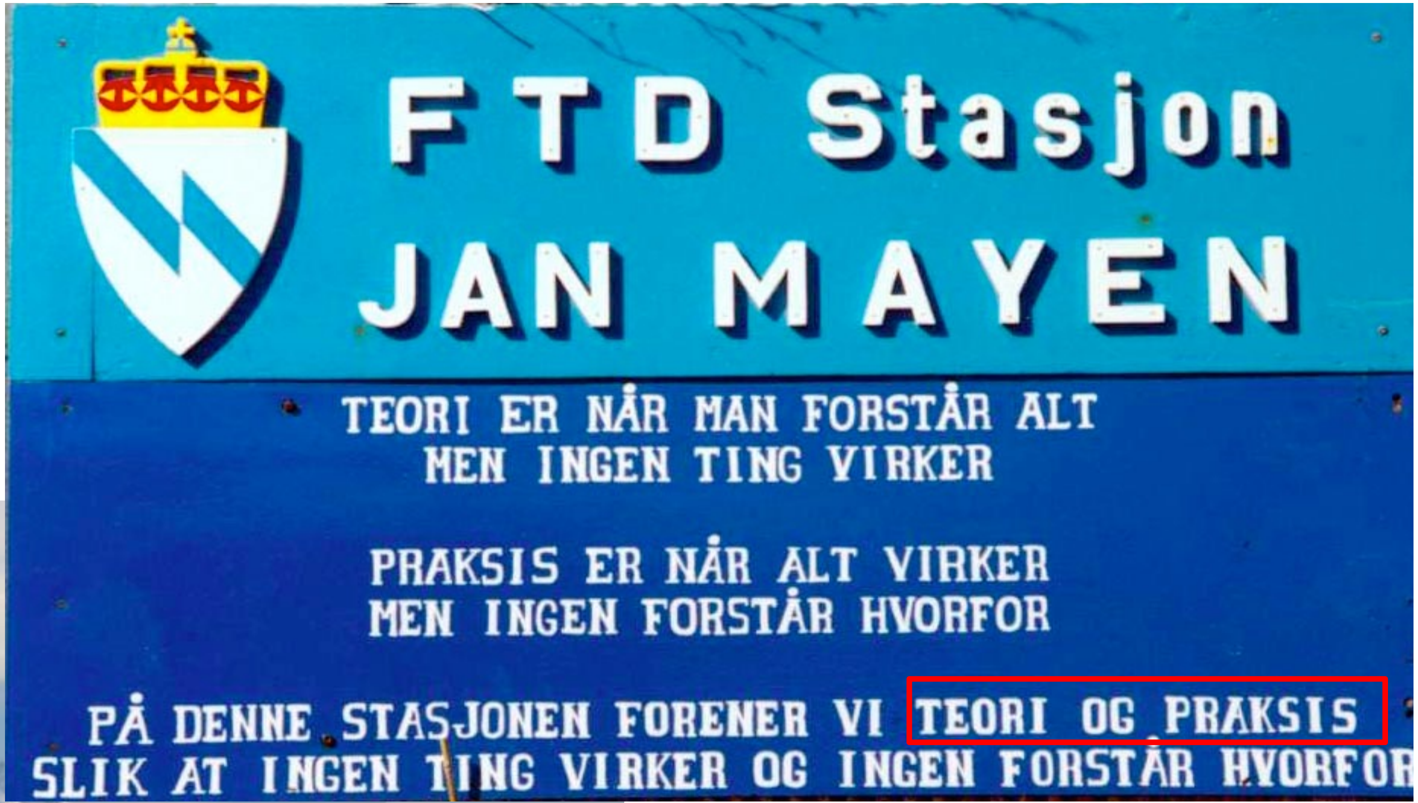
Evaluating the effect of a delegated vs. centralized control style on the maintainability of object-oriented software

Empirical



“Assuming that it is not only highly skilled experts who are going to maintain an object-oriented system, a viable conclusion from the controlled experiment reported in this paper is that a design with a centralized control style may be more maintainable than is a design with a delegated control style.”

Erik Arisholm and Dag Sjøberg, *IEEE Transactions on Software Engineering*, vol. 30, no. 8, August 2004, pp. 521-534.



FTD Stasjon
JAN MAYEN

TEORI ER NÅR MAN FORSTÅR ALT
MEN INGEN TING VIRKER

PRAKSIS ER NÅR ALT VIRKER
MEN INGEN FORSTÅR HVORFOR

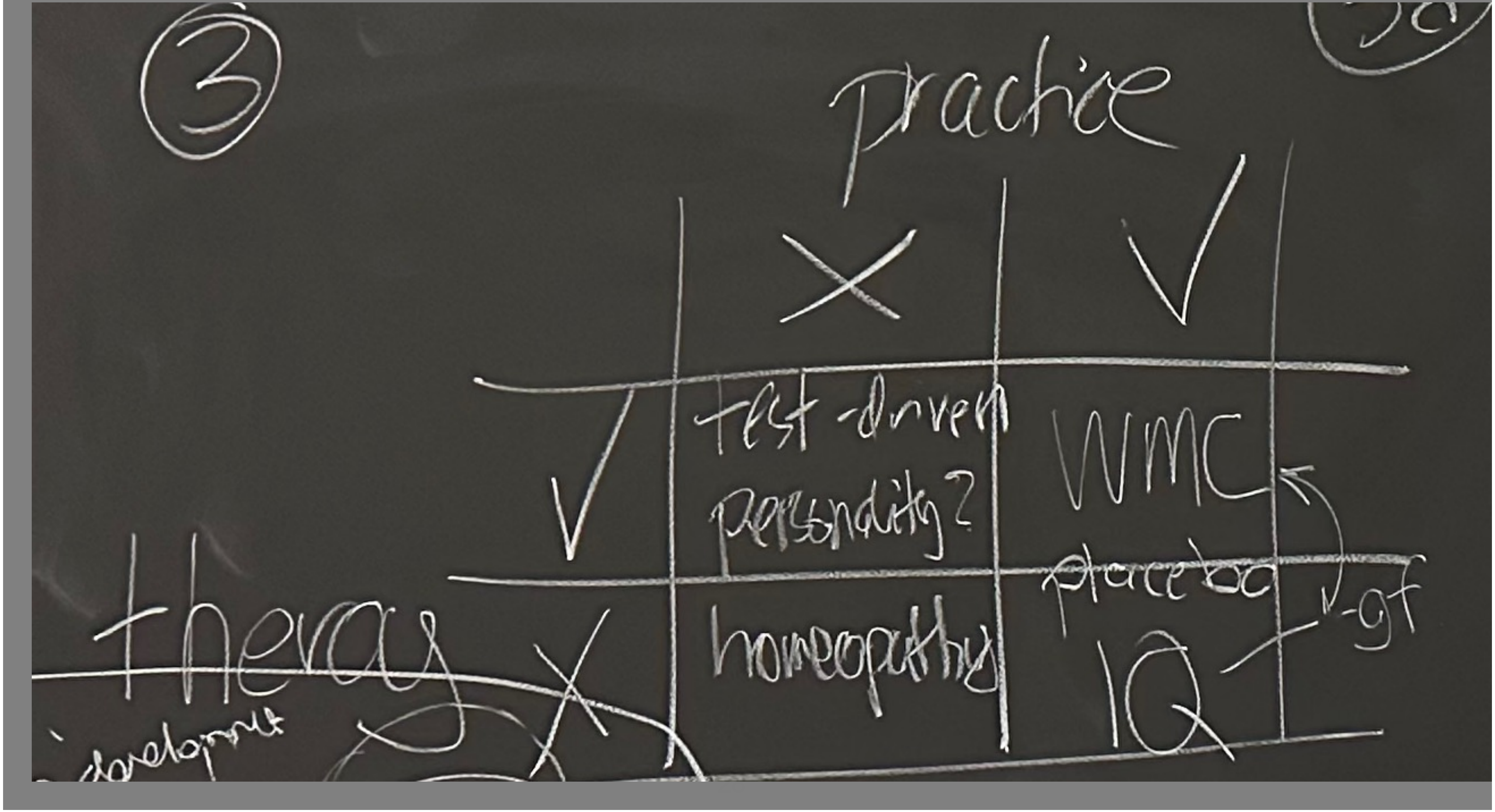
PÅ DENNE STASJONEN FORENER VI TEORI OG PRAKSIS
SLIK AT INGEN TING VIRKER OG INGEN FORSTÅR HVORFOR

Theory is when one understands everything, but nothing works

Practice is when everything works, but no one understand why

At [Jan Mayen] we unite theory and practice so that nothing works and no one understands why
(my translation)

Theory and practice



	✗	✓
✓	test-driven personality?	WMC
✗	homeopathy	placebo IQ

”The research I have available claims the opposite … ”

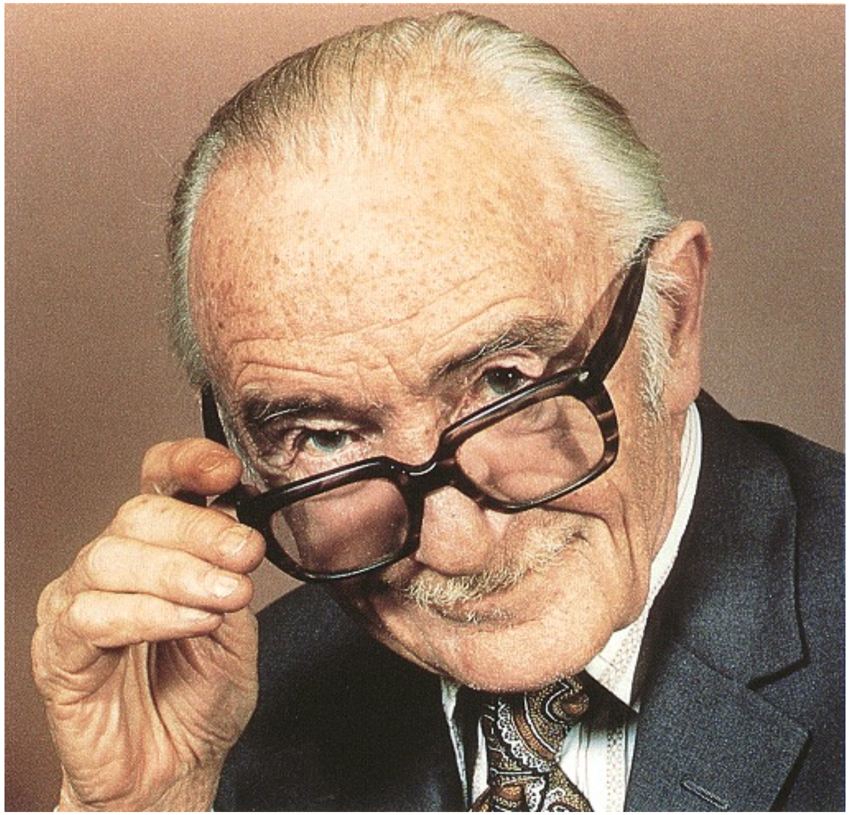
NRK Folkeopplysningen 3:7

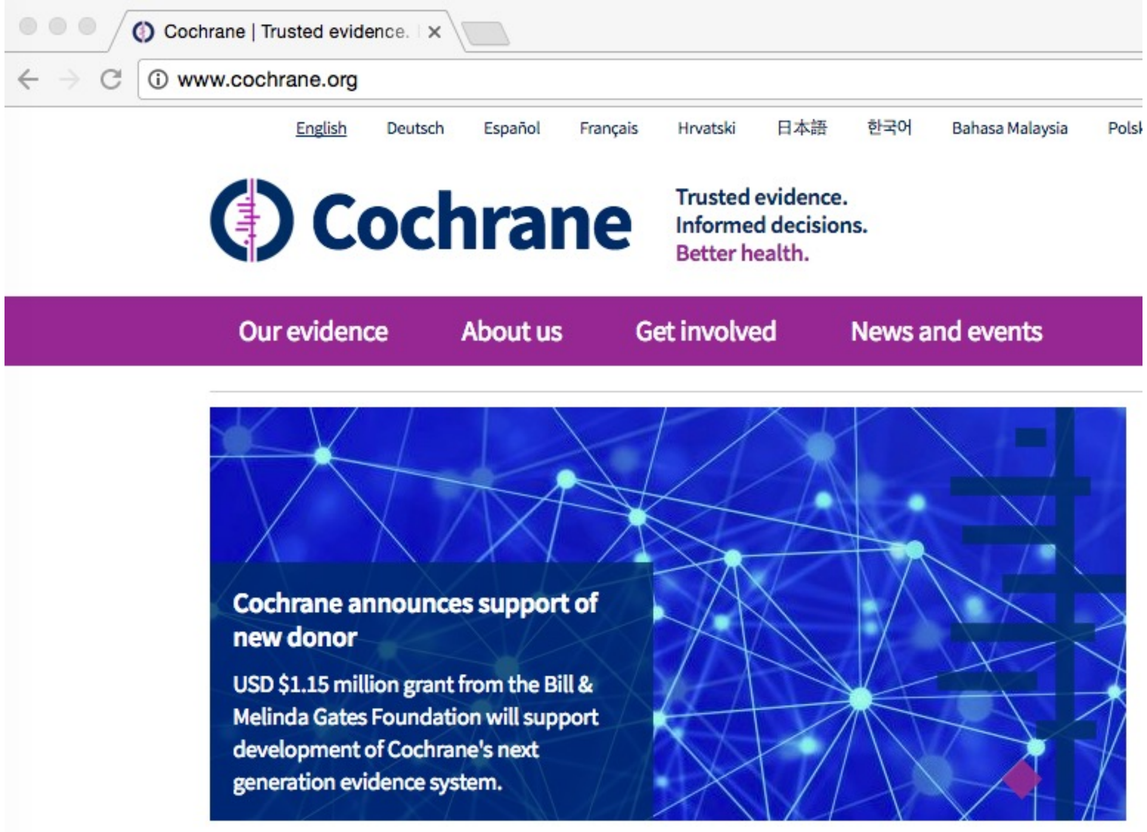
Structure

- Empirical research
- Evidence-based software engineering
  - The steps of EBSE
  - Research synthesis
- Empirical research methods
  - Controlled experiments
  - Case studies
  - Surveys
  - Action research
  - Validity
- The importance of context

Evidence-based software engineering

- Adapted from Evidence-Based Medicine
  - To provide the means by which current best evidence from research can be integrated with practical experience and human values in the decision making process regarding the development and maintenance of software
- EBSE sets requirements on practitioners and researchers:
  - Practitioners need to track down and use best evidence in context of practice
  - Researchers need to provide best evidence





Cochrane | Trusted evidence.

www.cochrane.org

English Deutsch Español Français Hrvatski 日本語 한국어 Bahasa Malaysia Polsk

Cochrane

Trusted evidence.
Informed decisions.
Better health.

Our evidence About us Get involved News and events

Cochrane announces support of new donor

USD $1.15 million grant from the Bill & Melinda Gates Foundation will support development of Cochrane's next generation evidence system.

Archie Cochrane

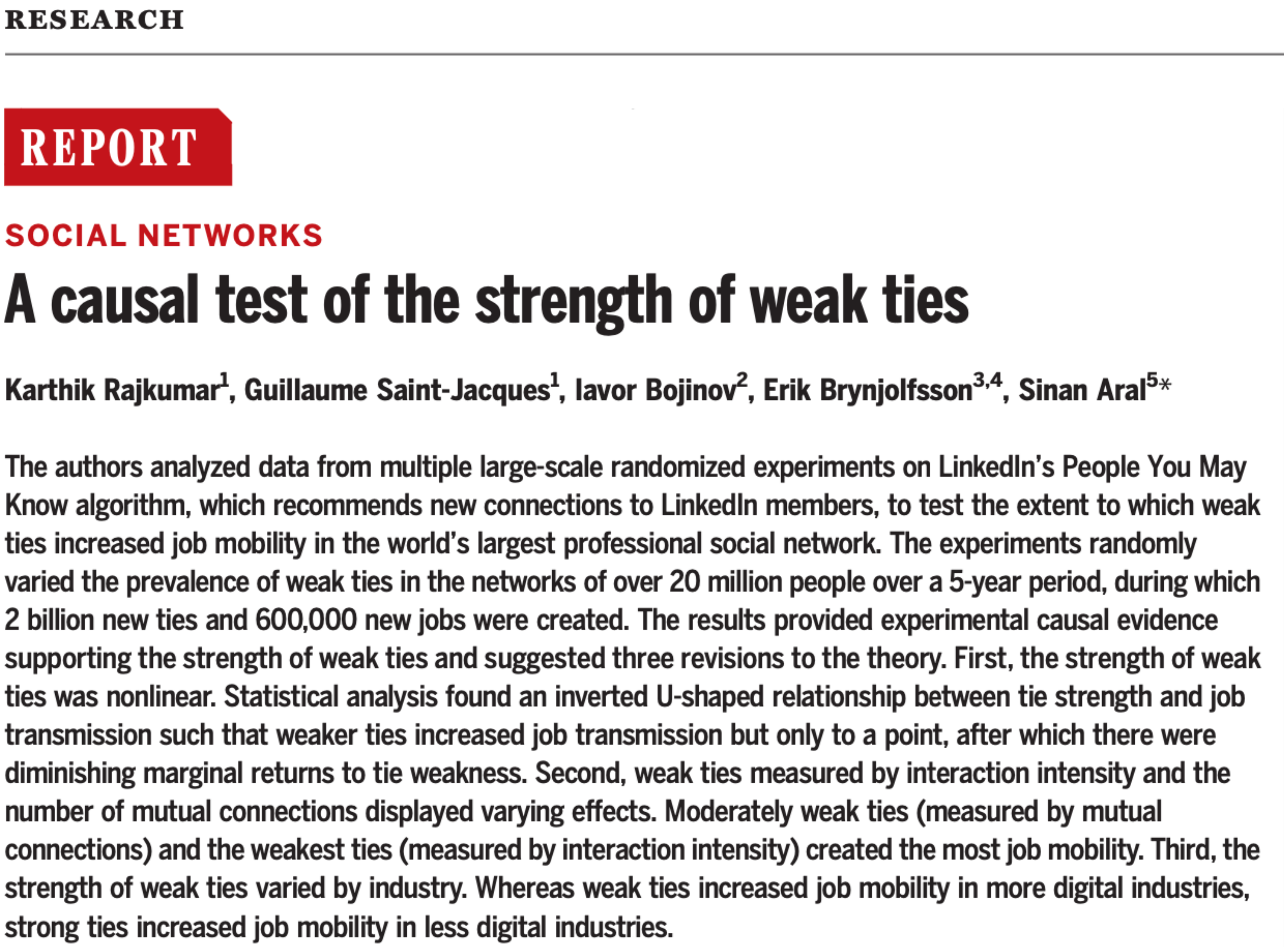
“It is surely a great criticism of our profession that we have not organised a critical summary, by specialty or subspecialty, adapted periodically, of all relevant randomised controlled trials.”

Software engineering challenges

- No comparable (to medicine) research infrastructure.
- No agreed standards for empirical studies
- Few software engineering guidelines based on empirical evidence.
- Challenges in addressing software engineering specifics
  - The skill factor
  - The lifecycle issue
  - The context dependences

The five steps of EBSE:

1. Converting a relevant problem or information need into an answerable question.
2. Searching the literature for the best available evidence to answer the question.
3. Critically appraising the evidence for its validity, impact, and applicability.
4. Integrating the appraised evidence with practical experience and the values and circumstances of the customer to make decisions about practice.
5. Evaluating performance and seeking ways to improve it.



RESEARCH

REPORT

SOCIAL NETWORKS

A causal test of the strength of weak ties

Karthik Rajkumar[1], Guillaume Saint-Jacques[1], Iavor Bojinov[2], Erik Brynjolfsson[3,4], Sinan Aral[5]*

The authors analyzed data from multiple large-scale randomized experiments on LinkedIn's People You May Know algorithm, which recommends new connections to LinkedIn members, to test the extent to which weak ties increased job mobility in the world's largest professional social network. The experiments randomly varied the prevalence of weak ties in the networks of over 20 million people over a 5-year period, during which 2 billion new ties and 600,000 new jobs were created. The results provided experimental causal evidence supporting the strength of weak ties and suggested three revisions to the theory. First, the strength of weak ties was nonlinear. Statistical analysis found an inverted U-shaped relationship between tie strength and job transmission such that weaker ties increased job transmission but only to a point, after which there were diminishing marginal returns to tie weakness. Second, weak ties measured by interaction intensity and the number of mutual connections displayed varying effects. Moderately weak ties (measured by mutual connections) and the weakest ties (measured by interaction intensity) created the most job mobility. Third, the strength of weak ties varied by industry. Whereas weak ties increased job mobility in more digital industries, strong ties increased job mobility in less digital industries.

Science 2022, vol. 377, Issue 6612, pp. 1304–1310, https://doi.org/10.1126/science.abl4476

Step 1: Asking an answerable question

- The first step in EBSE is to convert a relevant problem or information need into an answerable question.
- Typical questions ask for specific knowledge about how to appraise and apply methods, tools, and techniques in practice.
- Well formulated questions usually have three components:
  - The main intervention or action we are interested in.
  - The context or specific situations of interest.
  - The main outcomes or effects of interest.
- Example:
  - "Does the use of pair programming lead to improved code quality when practiced by professional software developers?"

Step 2: Finding the best evidence

- Finding an answer to our question includes selecting an appropriate information resource and executing a search strategy.
- The main source of research-based evidence is articles published in scientific journals. Examples of databases that index published articles include:
  - IEEE *Xplore*, http://ieeexplore.ieee.org
  - ACM Digital Library, http://www.acm.org/dl
  - ISI Web of Science, http://isiknowledge.com
  - Google scholar
- Often, reading important magazines such as the *Communications of the ACM*, *IEEE Computer*, *IEEE Software*, and *IT Professional* would probably be enough to get a general overview of the latest developments within software engineering.

Step 3: Critically appraising the evidence

- Unfortunately, published research isn't always of good quality; the problem under study might be unrelated to practice or the research method could have weaknesses so that the results cannot be trusted.
- To assess whether research is of good quality and can be applied to practice, we must be able to critically appraise the evidence.
  - Is there any vested interest?
  - Is the evidence valid?
  - Is the evidence important?
  - Can the evidence be used in practice?
  - Is the evidence in this study consistent with the evidence in other available studies?

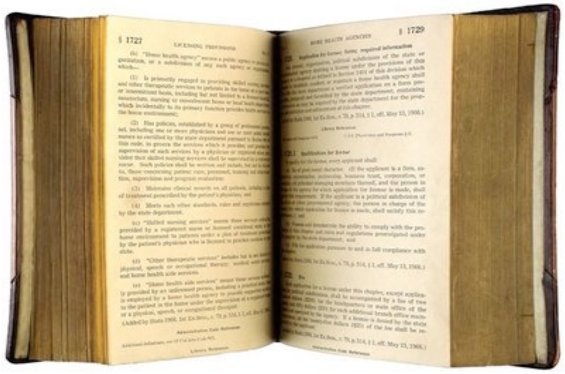
Step 4: Applying the evidence

- Active use of new knowledge is characterized by applying or adapting specific evidence to a specific situation in practice.
- Therefore, in order to practice EBSE, the individual software developer must commit him or herself to actively engage in a learning process, combining the externally transmitted evidence with prior knowledge and experience.
- Thus, it is at this point that EBSE needs to be integrated with process improvement.
- EBSE should provide the scientific basis for undertaking specific process changes while SPI should manage the process of introducing a new technology.

Step 5: Evaluating performance

- We need to consider how well we perform each step in the EBSE process and how we might improve our use of EBSE.
  - In particular, we should ask ourselves how well we are integrating evidence with practical experience, customer requirements, and our knowledge of the specific circumstances.
- Following SPI practice, we also need to assess whether process change has been effective.
  - This might include After Action Reviews, Postmortem Analyses, and organization-wide measurement programs.

What is research synthesis?

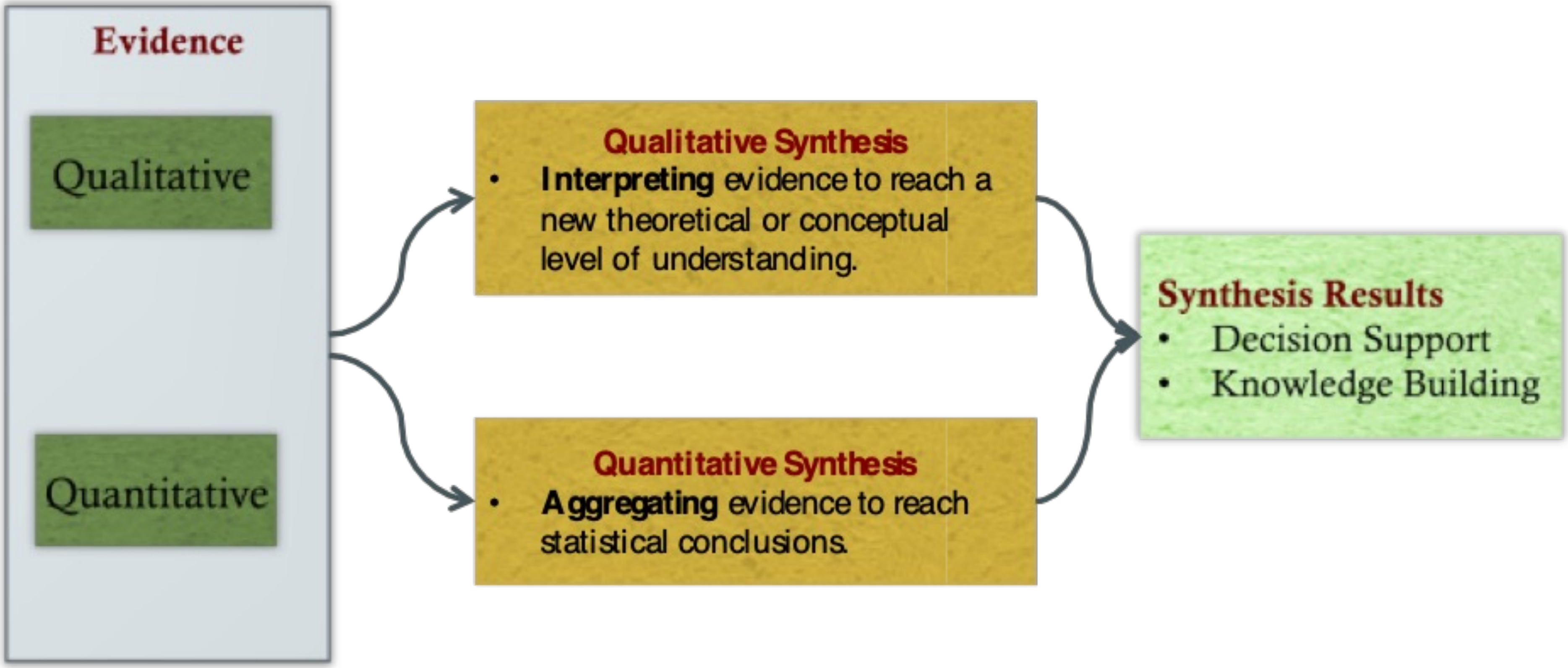


- Collective term for a family of methods for summarizing, integrating, combining, and comparing the findings of different studies on a topic or research question.
- Embodies the idea that individual studies or pieces of evidence are combined to produce a coherent whole, in the form of an argument, theory, or conclusions.
- It can provide conclusions with increased accuracy and less uncertainty compared to individual studies.
- A guiding principle is to be as rigorous and as transparent as possible.

Confidence in research synthesis depends on body of evidence strength and quality in primary studies and synthesis

- The confidence we can place in the conclusions and recommendations arising from a research synthesis depends on three issues:
  - The quality of the primary studies
  - The quality of the synthesis itself
  - The strength of the total body of evidence

Synthesis Types



Qualitative Synthesis

- **Interpreting** evidence to reach a new theoretical or conceptual level of understanding.

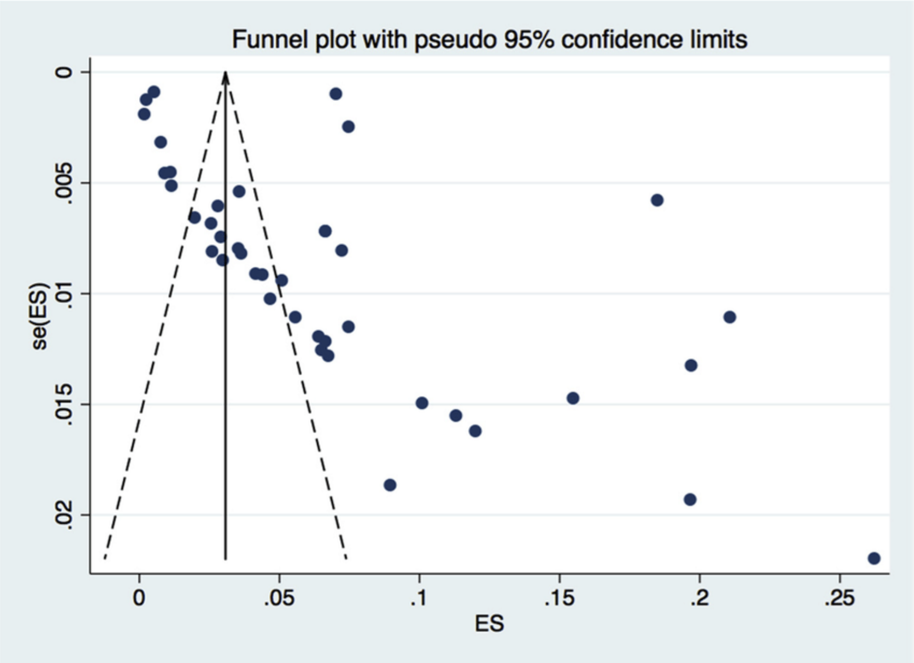
Quantitative Synthesis

- **Aggregating** evidence to reach statistical conclusions.

Synthesis Results

- Decision Support
- Knowledge Building

What if we have weak evidence in?



Weak Evidence

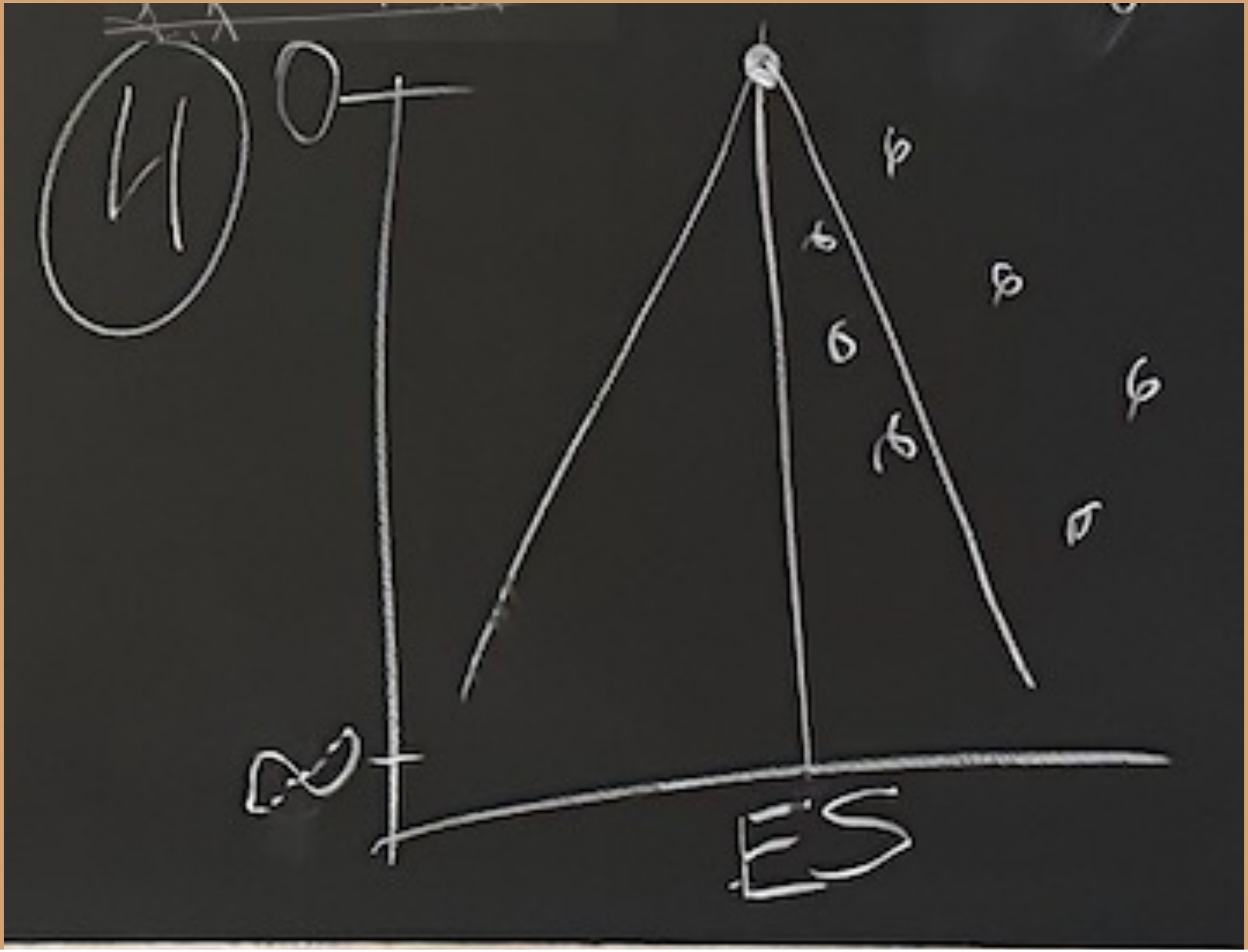
THIS IS GARBAGE. DON'T RECYCLE IT.

Research Synthesis

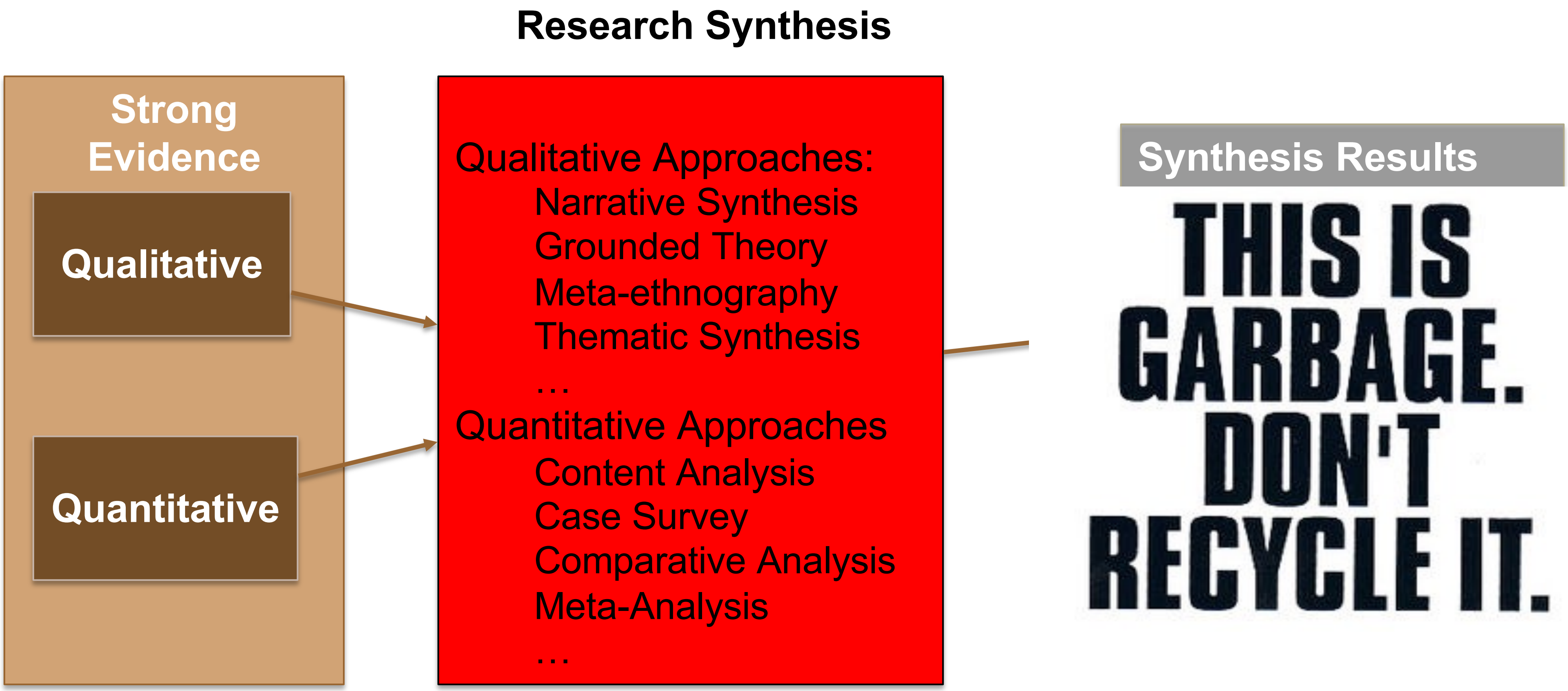
Synthesis Results for:

- Practitioners
- Researchers

Systematic bias in publications – funnel plot (previous slide)



What if we have a weak process of synthesis?



Research Synthesis

Strong Evidence

Qualitative

Quantitative

Qualitative Approaches:
- Narrative Synthesis
- Grounded Theory
- Meta-ethnography
- Thematic Synthesis
- …

Quantitative Approaches
- Content Analysis
- Case Survey
- Comparative Analysis
- Meta-Analysis
- …

Synthesis Results

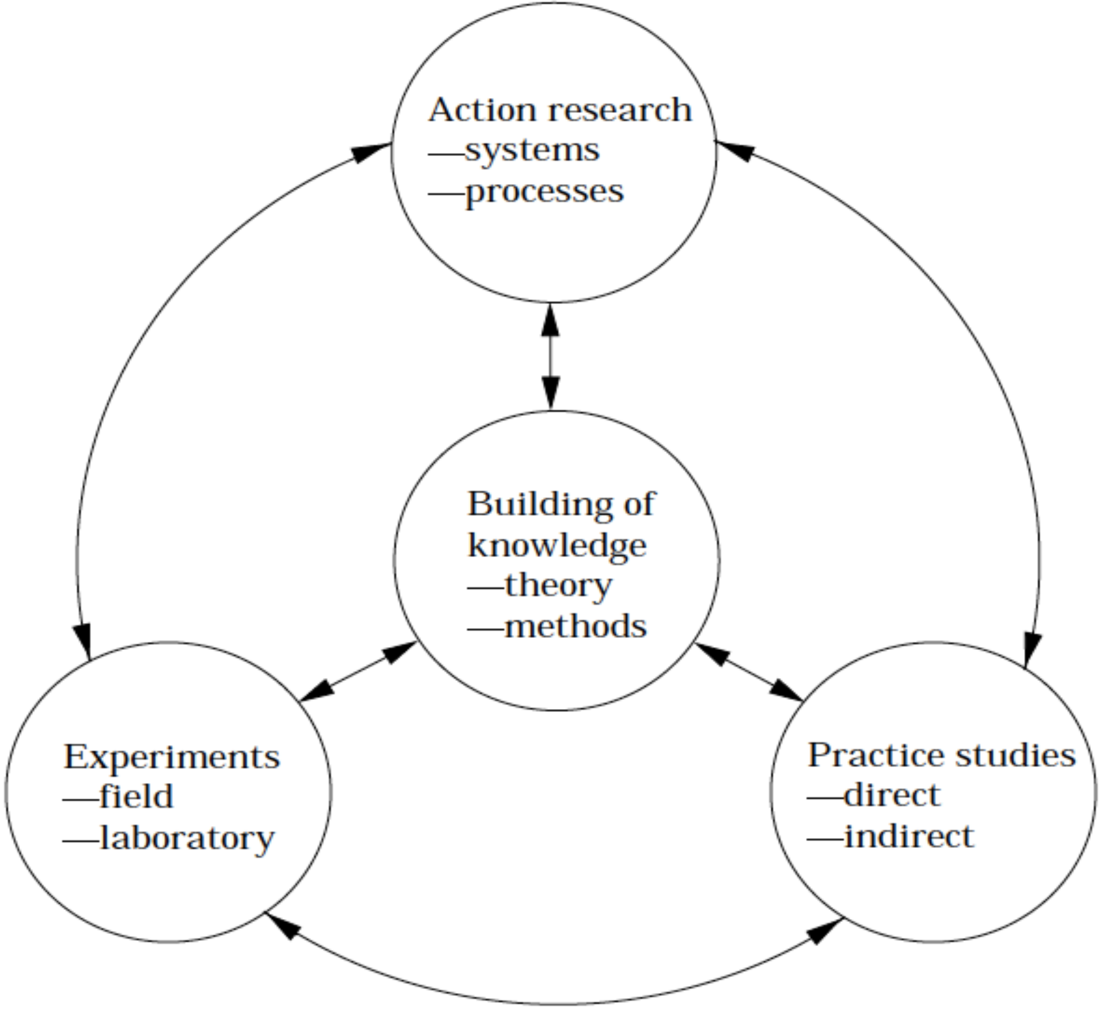
Empirical Research Methods

Bent Hamer: “Kitchen Stories,” 2003. Trailer

Structure

- Empirical research
- Evidence-based software engineering
  - The steps of EBSE
  - Research synthesis
- Empirical research methods
  - Controlled experiments
  - Case studies
  - Surveys
  - Action research
  - Validity
- The importance of context

An alternative, supporting approach to study SE practice



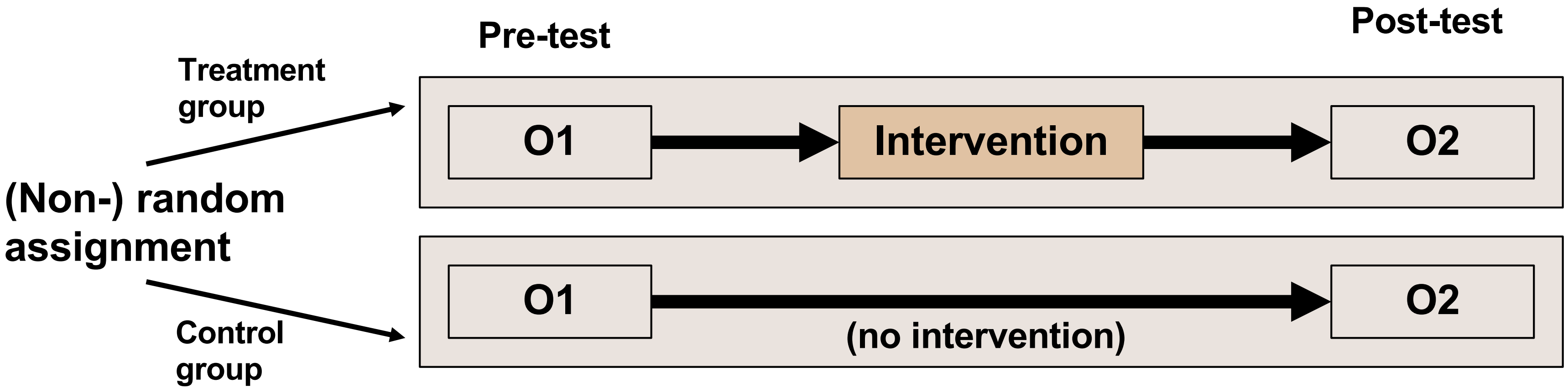
Lars Mathiassen, Collaborative Practice Research, *Information Technology & People*, Vol. 15 No. 4, 2002, pp. 321-345.

Controlled experiments

- An experiment is a study in which an intervention is deliberately introduced to observe its effects:
  - The identification of causal relations provides an explanation of *why* a phenomenon occurred.
  - The identification of casual processes yields an account of *how* a phenomenon occurred.
- Experiments are conducted when the investigator wants control over the situation, with direct, precise, and systematic manipulation of the behavior of the phenomenon to be studied.
- All experiments involve at least a treatment, an outcome measure, units of assignment, and some comparison from which change can be inferred and (hopefully) attributed to the treatment.

Classical experimental design

- **Randomized experiment**
  - An experiment in which units are assigned to receive the treatment or an alternative condition by a random process
- **Quasi-experiment**
  - An experiment in which units are not assigned to conditions randomly



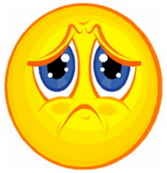
O1: Pre-intervention data collection point
O2: Post-intervention data collection point

Advantages

- They are a well established strategy, seen by many as the ‘scientific’ and therefore most acceptable approach
- They are the only research strategy that can prove causal relationships
- Laboratory experiments permit high levels of precision in measuring outcomes and in analyzing data

Disadvantages

- Laboratory experiments (e.g., with students at the university) often create artificial situations, which are not comparable with real-world situations
- It is often difficult or impossible to control all the relevant variables
- It is often difficult to recruit a representative sample of participants
- It may be necessary to conceal from the participants the purpose of the research so they do not skew the results



Experiment – example

What?

Research Question:

- What is best – Pair Programming or Solo Programming?

Why?

Many studies with contradicting results – mostly conducted with students (not with professional developers).

Source:

E. Arisholm, H. Gallis, T. Dybå, and D. Sjøberg, “Evaluating Pair Programming with Respect to System Complexity and Programmer Expertise,” *IEEE Transactions on Software Engineering*, 2007, 33(2): 65-86.

Who, where and when?

- 295 junior, intermediate and senior professional Java consultants from 29 companies were paid to participate (one work day)
- Norway, Sweden, UK; 2001-2005
- 99 individuals, 98 pairs
- The pairs and individuals performed the same Java maintenance tasks on either:
  - a ”simple” system (centralized control style), or
  - a ”complex” system (delegated control style)
- We measured:
  - duration (elapsed time)
  - effort (cost)
  - quality (correctness) of their solutions

Case study research

Case study research is an empirical inquiry that:

- Investigates a contemporary phenomenon within its real-life context, especially when
- the boundaries between phenomenon and context are not clearly evident.

Types of case studies:

- Singlecase, multicase
- Exploratory, descriptive, explanatory
- Holistic, embedded
- Qualitative, quantitative
- Positivist, interpretative, critical

Advantages

- It can deal with complex situations where it is difficult to study a single factor in isolation
- It is appropriate for situations where the researcher has little or no control over events
- It is suitable for both theory building and theory testing
- It allows the researcher to show complexities and to explore alternative meanings and explanations
- It produces data that is close to people’s experience

Disadvantages

- It is sometimes seen as lacking rigor and leading to generalizations with poor credibility
- It can be difficult and time-consuming to negotiate access to the necessary settings, people and documents
- The presence of the researcher can affect how people behave
- There aren’t really any rules to follow

Case study – example

What?

Research Question:

- What are the challenges of shared decision-making in agile software development teams?

Why?

Agile software development changes the nature of collaboration, coordination, and communication in software projects

Source:

N.B. Moe, A. Aurum, and T. Dybå, “Challenges of Shared Decision-Making: A Multiple Case Study of Agile Software Development,” Information and Software Technology, 2010, 54(8): 853-865.

Who, where and when?

- Multiple case study of four projects in two software product companies
- Norway; 2007-2010
- Both companies recently adopted Scrum
  - One company introduced Scrum in the middle of two 3-year projects
  - One company introduced Scrum at the beginning of two 9-12 month projects
- We collected data in semi-structured interviews, through participant observations, and from process artifacts
- Data collected over a period of 11-12 months in all four projects

Action Research

- Simultaneously contribute to the practical concerns in a concrete situation and to the goals of science.
- Dual commitment to study a system and concurrently to collaborate with members of the system in changing it.
- Active collaboration between researchers and practitioners underlines the importance of co-learning as a primary aspect of the research process.

Rule 3
Observe without interference

Action Research attempts to provide

practical value

to the client organization while simultaneously contributing

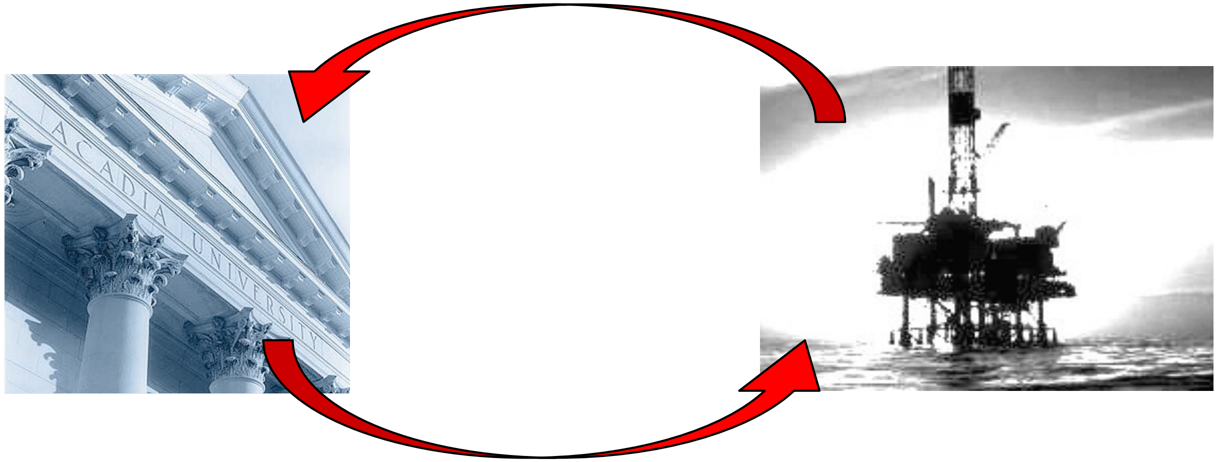
to the acquisition of

new theoretical knowledge

Criticisms of Action Research

- Action Research has been criticized for:
  - its lack of methodological rigor
  - its lack of distinction from consulting, and
  - its tendency to produce either

'research with little action or action with little research'



Action research – example

What?

Research Question:

- What benefits and challenges can arise from introducing knowledge redundancy interventions based on job rotation in software development?

Why?

Establish a formalized support service and contribute to improved flexibility in project staffing by knowledge redundancy

Source:

T.E. Fægri, T. Dybå, and T. Dingsøyr (2010) "Introducing Knowledge Redundancy Practice in a Small Software Organization: Experiences with Job Rotation in Support Work," *Information and Software Technology*, 52(10): 1118-1132.

Who, where and when?

- Action research in one company to integrate organizational change with scientific inquiry.
- Norway; 2008
- The practical objectives were:
  - to establish customer support as a legitimate organizational function that would shield developers from support enquiries, and
  - to contribute to improved flexibility in project staffing by enabling overlapping product experience among developers.
- During a period of 18 weeks, nine developers rotated to customer support.
- We collected data in meetings, from comprehensive interviews, and from customer support work logs.

Survey research

- A survey is useful for studying a large number of variables using a large sample size and rigorous statistical analysis.
- They are used when:
  - control of the independent and dependent variables is not possible or not desirable,
  - when the phenomena of interest must be studied in their natural setting, and
  - when the phenomena of interest occur in current time or the recent past.

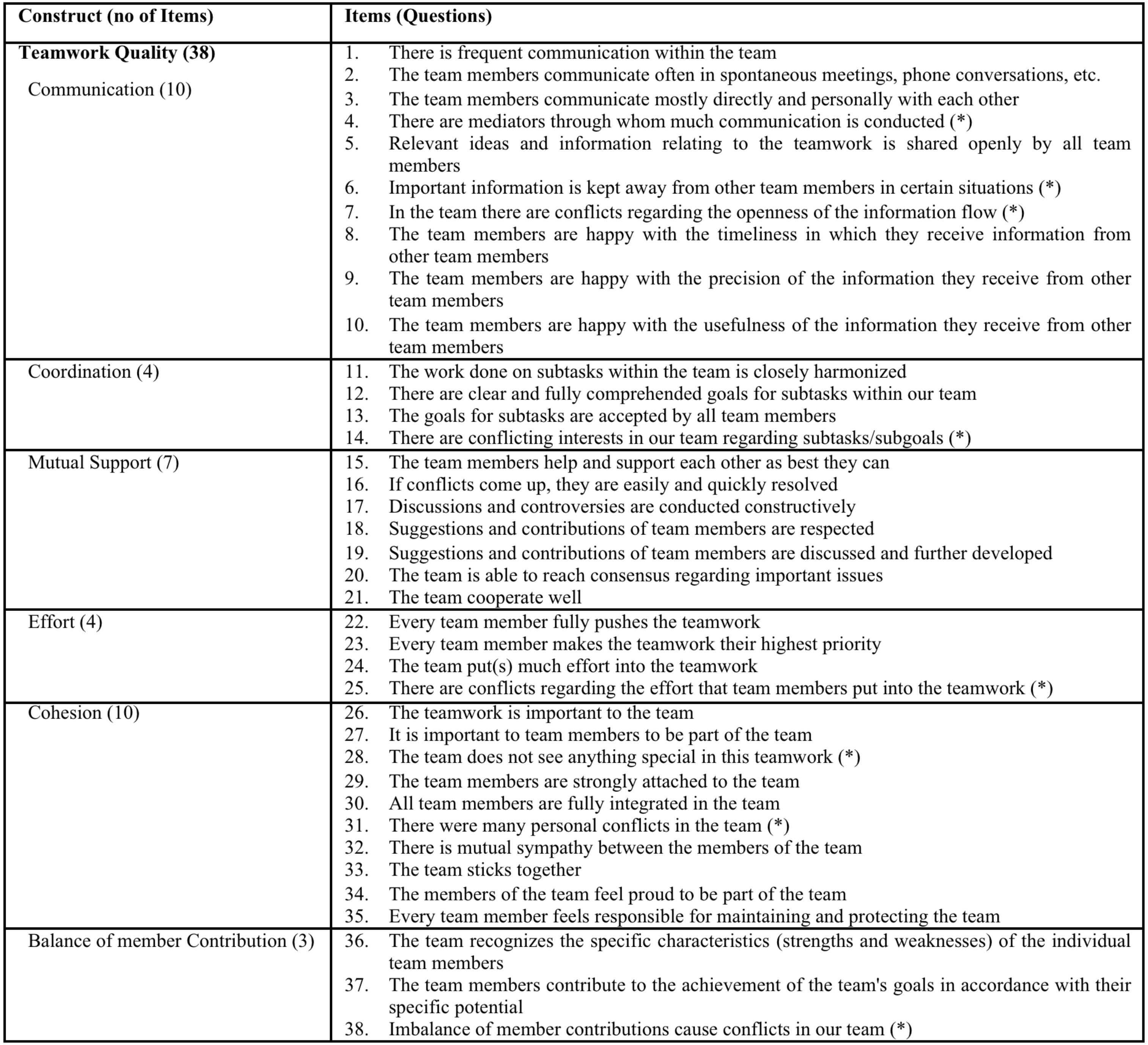
Example

Teamwork Quality and Project Success in Agile Software Development: A Survey of Agile Development Teams

Yngve Lindsjørn[a], Dag I.K Sjøberg[a,b], Torgeir Dingsøyr[b], Gunnar R. Bergersen[a], Tore Dybå[b,a]

[a] Department of Informatics, University of Oslo, Norway {ynglin, dagsj, gunnab} @ifi.uio.no
[b] SINTEF, Trondheim, Norway {torgeir.dingsoyr, tore.dyba}@sintef.no

Table 9 - Items in Questionnaire



Construct (no of Items)	Items (Questions)
Teamwork Quality (38) Communication (10)	1. There is frequent communication within the team 2. The team members communicate often in spontaneous meetings, phone conversations, etc. 3. The team members communicate mostly directly and personally with each other 4. There are mediators through whom much communication is conducted (*) 5. Relevant ideas and information relating to the teamwork is shared openly by all team members 6. Important information is kept away from other team members in certain situations (*) 7. In the team there are conflicts regarding the openness of the information flow (*) 8. The team members are happy with the timeliness in which they receive information from other team members 9. The team members are happy with the precision of the information they receive from other team members 10. The team members are happy with the usefulness of the information they receive from other team members
Coordination (4)	11. The work done on subtasks within the team is closely harmonized 12. There are clear and fully comprehended goals for subtasks within our team 13. The goals for subtasks are accepted by all team members 14. There are conflicting interests in our team regarding subtasks/subgoals (*)
Mutual Support (7)	15. The team members help and support each other as best they can 16. If conflicts come up, they are easily and quickly resolved 17. Discussions and controversies are conducted constructively 18. Suggestions and contributions of team members are respected 19. Suggestions and contributions of team members are discussed and further developed 20. The team is able to reach consensus regarding important issues 21. The team cooperate well
Effort (4)	22. Every team member fully pushes the teamwork 23. Every team member makes the teamwork their highest priority 24. The team put(s) much effort into the teamwork 25. There are conflicts regarding the effort that team members put into the teamwork (*)
Cohesion (10)	26. The teamwork is important to the team 27. It is important to team members to be part of the team 28. The team does not see anything special in this teamwork (*) 29. The team members are strongly attached to the team 30. All team members are fully integrated in the team 31. There were many personal conflicts in the team (*) 32. There is mutual sympathy between the members of the team 33. The team sticks together 34. The members of the team feel proud to be part of the team 35. Every team member feels responsible for maintaining and protecting the team
Balance of member Contribution (3)	36. The team recognizes the specific characteristics (strengths and weaknesses) of the individual team members 37. The team members contribute to the achievement of the team's goals in accordance with their specific potential 38. Imbalance of member contributions cause conflicts in our team (*)

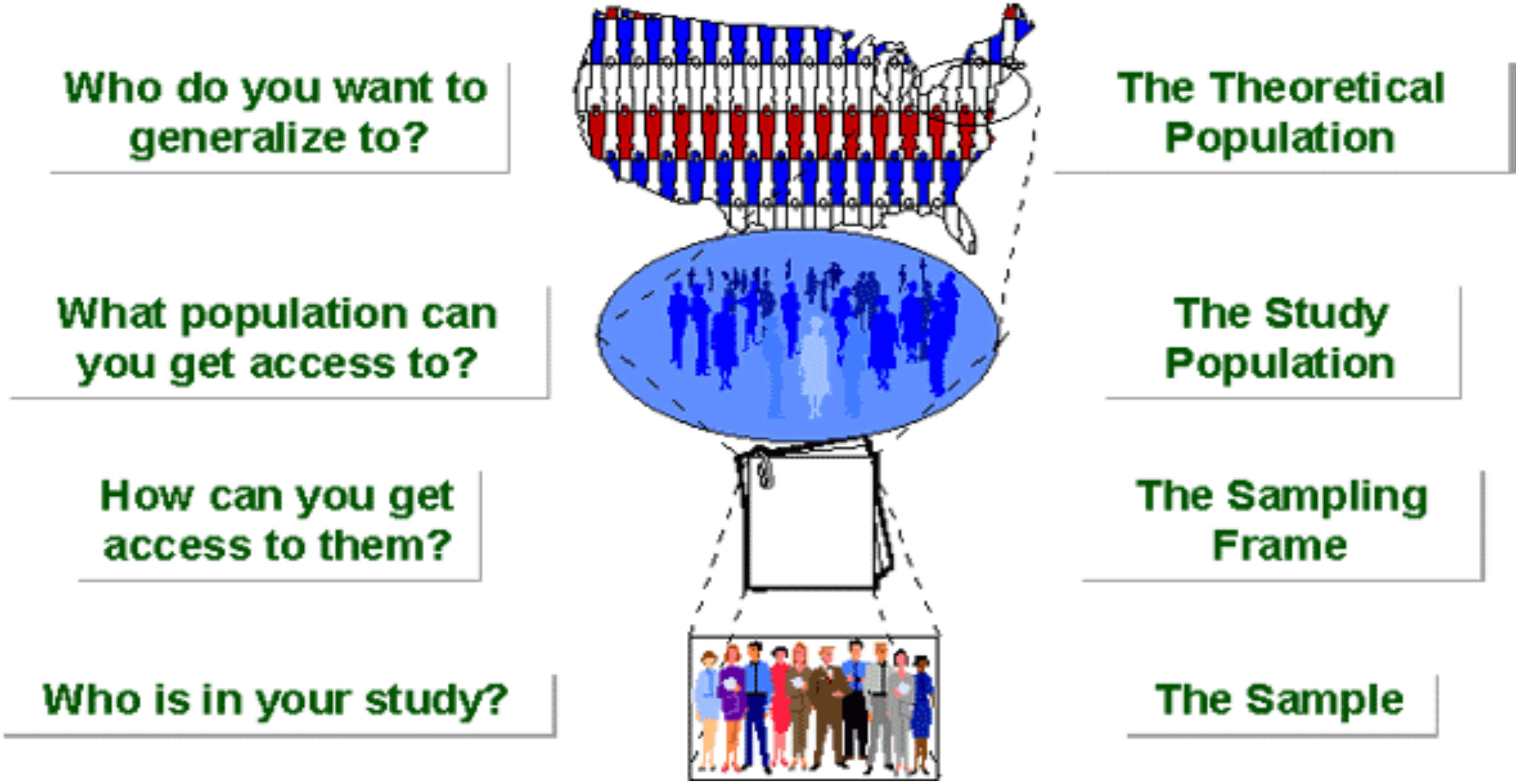
Common in society

- Requires relatively few resources to include many people
- Create statistics and test hypotheses over characteristics of the target group (the population being investigated)
- Obtain information about people's *opinion* about what, how much, how many, how and why or what people *say* they do
  - As opposed to experiments, one does *not control* independent and dependent variables
  - As opposed case studies and ethnography, one does *not observe*

Types of surveys

- **Cross-sectional surveys** are used to gather information on a population at a single point in time.
- **Longitudinal surveys** gather data over a period of time. The researcher may then analyze changes in the population and attempt to describe and/or explain them.
  - *Trend studies* focus on a particular population, which is sampled and scrutinized repeatedly. While samples are of the same population, they are typically not composed of the same people.
  - *Cohort studies* also focus on a particular population, sampled and studied more than once. A cohort study would sample the same group of people, every time.
  - *Panel studies* allow the researcher to find out why changes in the population are occurring, since they use the same sample of people every time.

Sampling and generalization



Who do you want to generalize to? — The Theoretical Population

What population can you get access to? — The Study Population

How can you get access to them? — The Sampling Frame

Who is in your study? — The Sample

W.M. Trochim, The Research Methods Knowledge Base, http://www.socialresearchmethods.net/kb/

Types of questions

- All researchers must make two basic decisions when designing a survey – they must decide:
  1. whether they are going to employ an oral, written, or electronic method, and
  2. whether they are going to choose questions that are open or close-ended.
- We will focus on **written** and **close-ended** methods.

The importance of wording …

Two catholic priests wondered if one is allowed to smoke when one prays? They both sent a letter to the Pope:

P1: “Is it allowed to smoke when one prays?”

Answer: **NO** – the pray should get full attention

P2: “Is it allowed to pray when one smokes?”

Answer: **YES** – it is always a good thing to pray

Question formats

- Classification of objects or individuals.

  “Are you: Male___ Female ___ Other ___?”

- Ranking of items in order to reflect the relative ordering of phenomena.

  “Please rank the following factors in order of importance (1-4)”

- Pairwise comparison

  “Which do you prefer”

- Rating of characteristics
  - Simple, single-item scales, e.g.,

    “Programming is a terrific course (check one)”

    “Strongly agree__, agree__, neither__, disagree__, strongly disagree__”

Likert type scales

- **Evaluation-type**

  Example:

  – “Familiarity with and comprehension of the software development environment”

  □ Little
  □ Unsatisfactory
  □ Neutral
  □ Satisfactory
  □ Excellent

- **Frequency-type**

  Example:

  –“Customers provide information to the project team about the requirements”

  □ Never
  □ Rarely
  □ Neutral
  □ Occasionally
  □ Most of the time

- **Agreement-type**

  Example:

  –“The tasks supported by the software at the customer site change frequently”

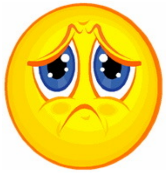
  □ Strongly Agree
  □ Agree
  □ Neutral
  □ Disagree
  □ Strongly Disagree

Advantages

- They provide a wide an inclusive coverage of people or events
- They can be administered from remote locations using mail, email or telephone
- They can provide a lot of data in a short time at a reasonable cost
- They lend themselves to quantitative analysis
- They can be replicated
- Usually, high reliability is easy to obtain

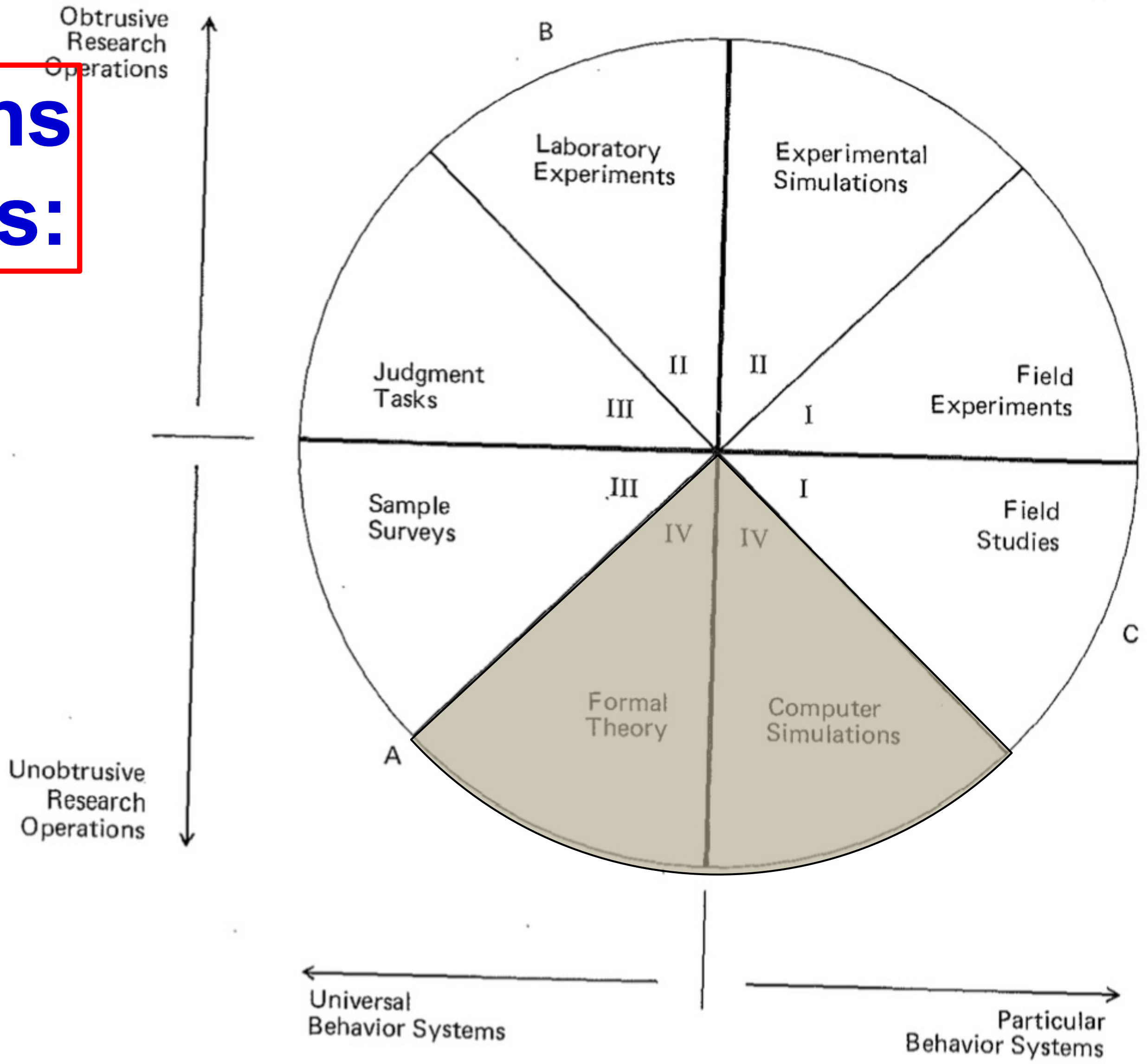
Disadvantages

- They lack depth
- They tend to focus on what can be counted or measured
- They do not establish cause and effect
- They cannot judge the accuracy or honesty of people's responses by observing their body language



Important dimensions of empirical methods:

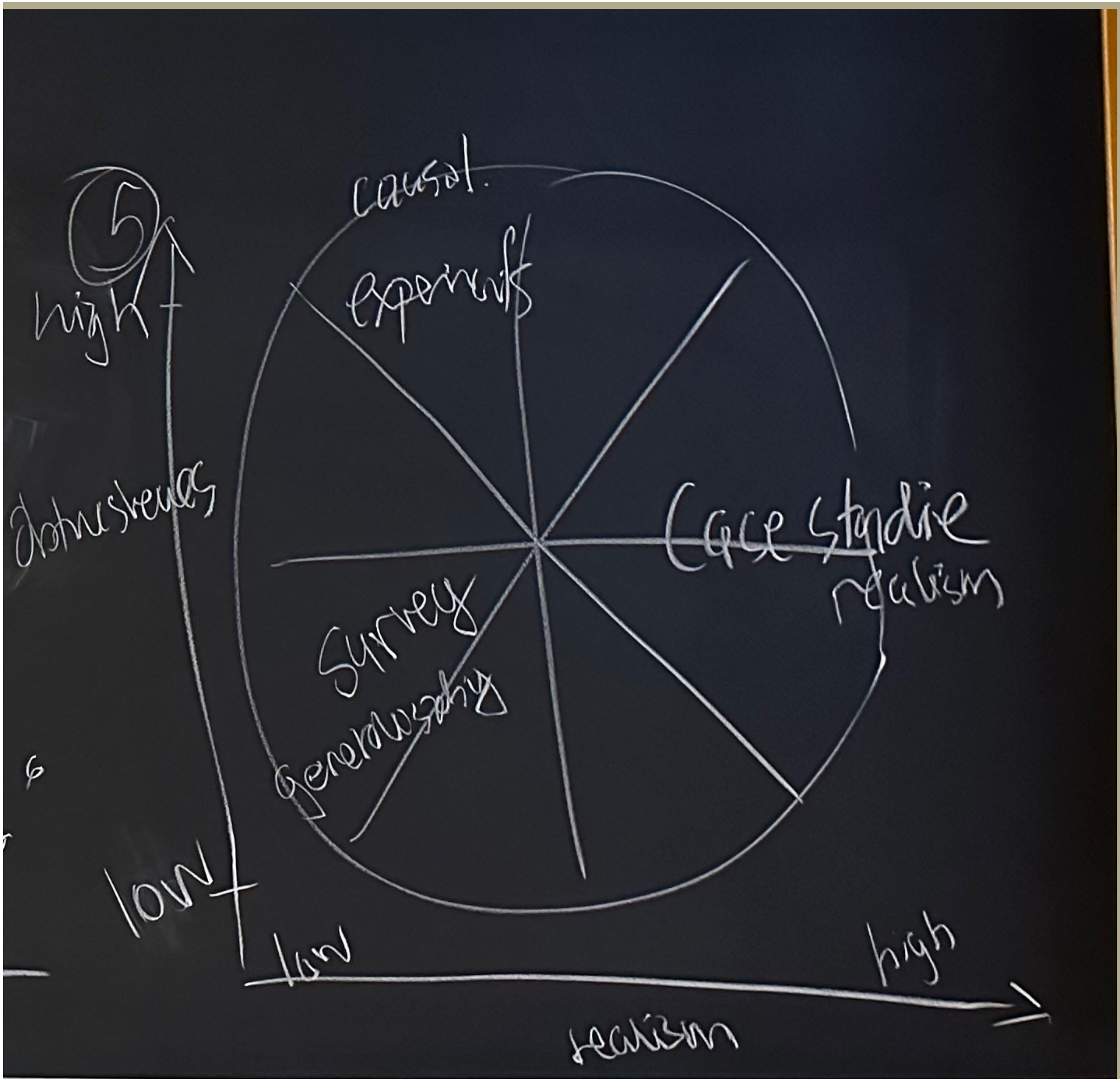
- –obtrusiveness,
- –generality,
- –artificiality, and
- –point of maximum concern



I. Settings in natural systems.
II. Contrived and created settings.
III. Behavior not setting dependent.
IV. No observation of behavior required.

A. Point of maximum concern with generality over actors.
B. Point of maximum concern with precision of measurement of behavior.
C. Point of maximum concern with system character of context.

Philip J. Runkel & Joseph E. McGrath, Research on human behavior: Systematic guide to method. New York: Holt, Rinehart and Wilson, 1972, p. 85.

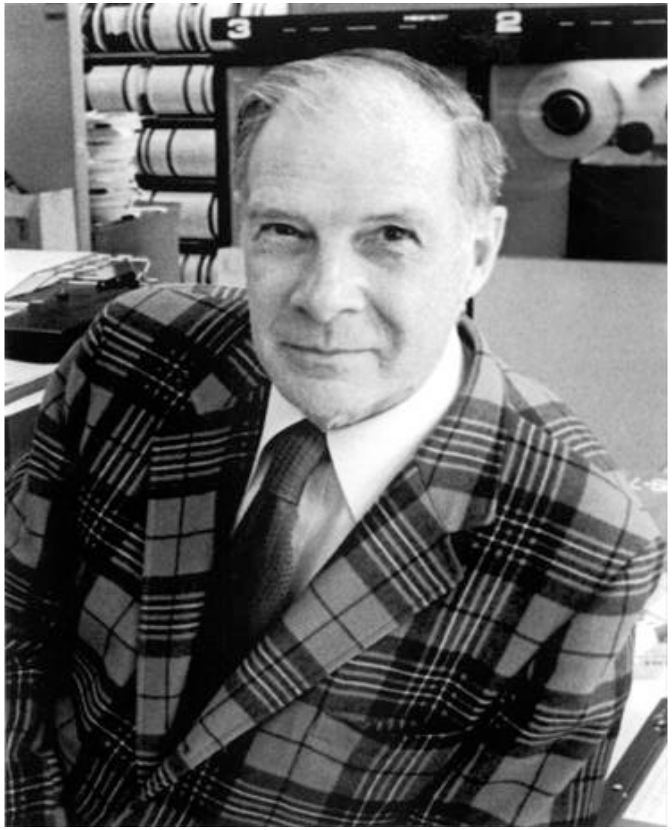

5
high
obtrusiveness
low
causal.
experiments
Case studie
realism
Survey
Generalisability
6
low
high
realism

Selecting the research method

Research question	Controlled experiment	Longitudinal survey	Cross-sectional survey	Case study Action research
Effectiveness: Does it work? Does method A work better than method B?	++	+	-	--
Explanation: How does it work? Why does it work?	--	-	+	++
Context: In what circumstances does it work, for whom?	--	-	+	++
Safety: Will it do more good than harm?	++	+	+	+
Acceptability: Will the target group accept the new method of working?	--	-	+	++
Prevalence: How often is this method/ technique applied/implemented?	--	--	++	--
Appropriateness: Is this the right process/method for this target group?	--	-	+	++

Adapted from cebma.org

“The purpose of computing is insight, not numbers.”



Richard Hamming

Structure

- Empirical research
- Evidence-based software engineering
  - The steps of EBSE
  - Research synthesis
- Empirical research methods
  - Controlled experiments
  - Case studies
  - Surveys
  - Action research
  - Validity
- The importance of context

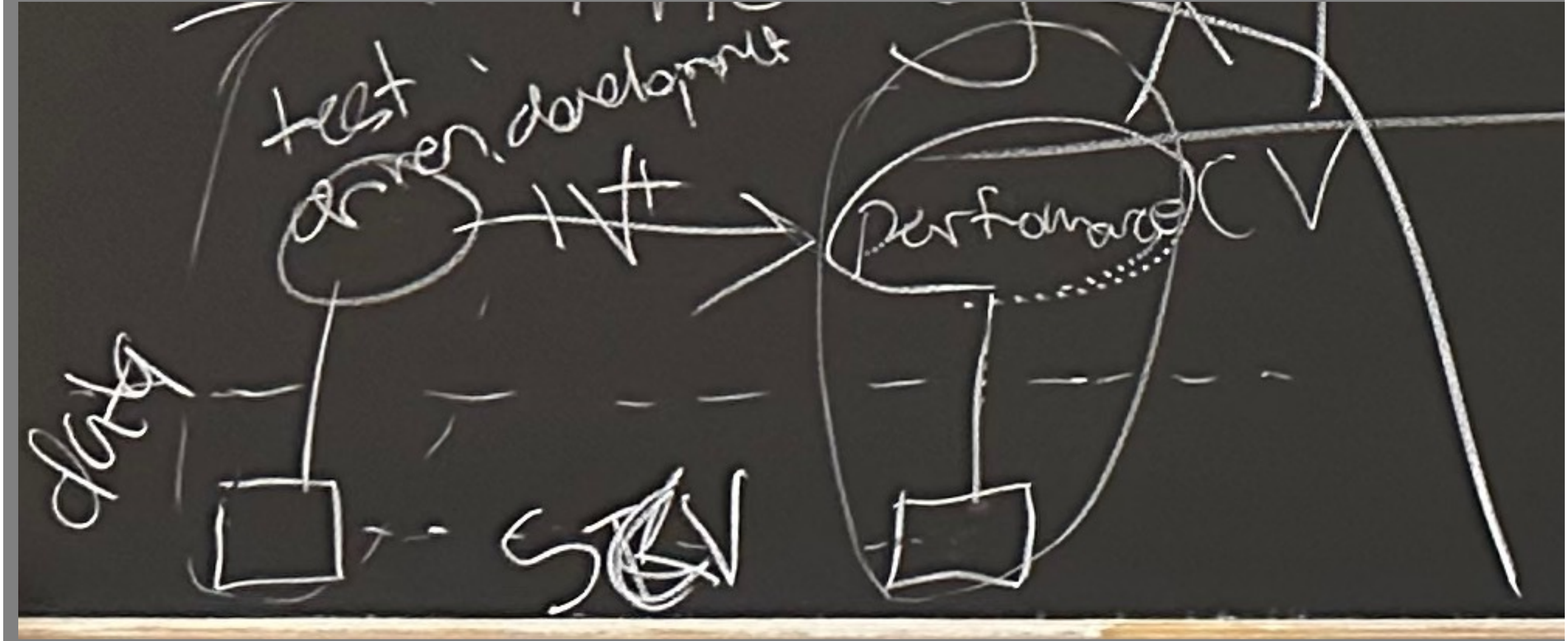
Reliability and validity of empirical studies

- **Reliability**
  - Can the study can be repeated (i.e., by other researchers) and yield the same results?

- **Statistical conclusion validity**
  - Is the statistical inference valid?
- **Internal validity**
  - Does the observed covariation between *A* (the presumed treatment) and *B* (the presumed outcome) reflects a causal relationship from *A* to *B?*
- **Construct validity**
  - Do the measures in the study represent the (abstract, possibly theoretical) constructs they are intended to measure?
- **External validity**
  - Does the cause–effect relationship hold over variations in persons, settings, treatment variables, and measurement variables?

The four validities

See Sjøberg & Bergersen (2023) - Construct validity in SE, figure 4, for a better picture
https://ieeexplore.ieee.org/document/9780058

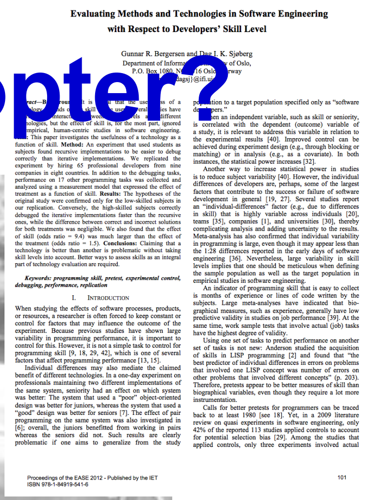


The quality of empirical studies

Three methodological features have been shown to influence the results of primary studies:

- **Randomization** can avoid selection bias by making sure that each subject in the study has an equal chance of getting into each treatment group.
- **Blinding** of study participants and personnel may reduce the risk that knowledge of which treatment was received, rather than the treatment itself, affects outcomes and outcome measurements.
- **Missing outcome data**, due to attrition (withdrawal and dropout) during the study or exclusions from the analysis, raise the possibility that the observed effect estimate is biased.

Context: What is best? Bicycle or helicopter?



Evaluating Methods and Technologies in Software Engineering with Respect to Developers' Skill Level

Gunnar R. Bergersen and Dag I. K. Sjøberg
Department of Informa[illegible] of Oslo,
P.O. Box 1080, N[illegible]16 Oslo, [illegible]rway
[illegible]dagsj}@ifi.ui[illegible]

[illegible]ract—B[illegible] It is [illegible] that the use[illegible]ss of a [illegible]ology [illegible]nds o[illegible] skill [illegible] use[illegible]eral [illegible]ies have [illegible] interact[illegible]ween [illegible]vels a[illegible] different [illegible]nologies, but the effect of skill is, for the most part, ignored [illegible]mpirical, human-centric studies in software engineering. [illegible]: This paper investigates the usefulness of a technology as a function of skill. **Method:** An experiment that used students as subjects found recursive implementations to be easier to debug correctly than iterative implementations. We replicated the experiment by hiring 65 professional developers from nine companies in eight countries. In addition to the debugging tasks, performance on 17 other programming tasks was collected and analyzed using a measurement model that expressed the effect of treatment as a function of skill. **Results:** The hypotheses of the original study were confirmed only for the low-skilled subjects in our replication. Conversely, the high-skilled subjects correctly debugged the iterative implementations faster than the recursive ones, while the difference between correct and incorrect solutions for both treatments was negligible. We also found that the effect of skill (odds ratio = 9.4) was much larger than the effect of the treatment (odds ratio = 1.5). **Conclusions:** Claiming that a technology is better than another is problematic without taking skill levels into account. Better ways to assess skills as an integral part of technology evaluation are required.

Keywords: programming skill, pretest, experimental control, debugging, performance, replication

I. Introduction

When studying the effects of software processes, products, or resources, a researcher is often forced to keep constant or control for factors that may influence the outcome of the experiment. Because previous studies have shown large variability in programming performance, it is important to control for this. However, it is not a simple task to control for programming skill [9, 18, 29, 42], which is one of several factors that affect programming performance [13, 15].

Individual differences may also mediate the claimed benefit of different technologies. In a one-day experiment on professionals maintaining two different implementations of the same system, seniority had an effect on which system was better: The system that used a "poor" object-oriented design was better for juniors, whereas the system that used a "good" design was better for seniors [7]. The effect of pair programming on the same system was also investigated in [6]; overall, the juniors benefitted from working in pairs whereas the seniors did not. Such results are clearly problematic if one aims to generalize from the study po[illegible]ation to a target population specified only as "software de[illegible]pers."

[illegible]hen an independent variable, such as skill or seniority, is correlated with the dependent (outcome) variable of a study, it is relevant to address this variable in relation to the experimental results [40]. Improved control can be achieved during experiment design (e.g., through blocking or matching) or in analysis (e.g., as a covariate). In both instances, the statistical power increases [32].

Another way to increase statistical power in studies is to reduce subject variability [40]. However, the individual differences of developers are, perhaps, some of the largest factors that contribute to the success or failure of software development in general [19, 27]. Several studies report an "individual-differences" factor (e.g., due to differences in skill) that is highly variable across individuals [20], teams [35], companies [1], and universities [30], thereby complicating analysis and adding uncertainty to the results. Meta-analysis has also confirmed that individual variability in programming is large, even though it may appear less than the 1:28 differences reported in the early days of software engineering [36]. Nevertheless, large variability in skill levels implies that one should be meticulous when defining the sample population as well as the target population in empirical studies in software engineering.

An indicator of programming skill that is easy to collect is months of experience or lines of code written by the subjects. Large meta-analyses have indicated that biographical measures, such as experience, generally have low predictive validity in studies on job performance [39]. At the same time, work sample tests that involve actual (job) tasks have the highest degree of validity.

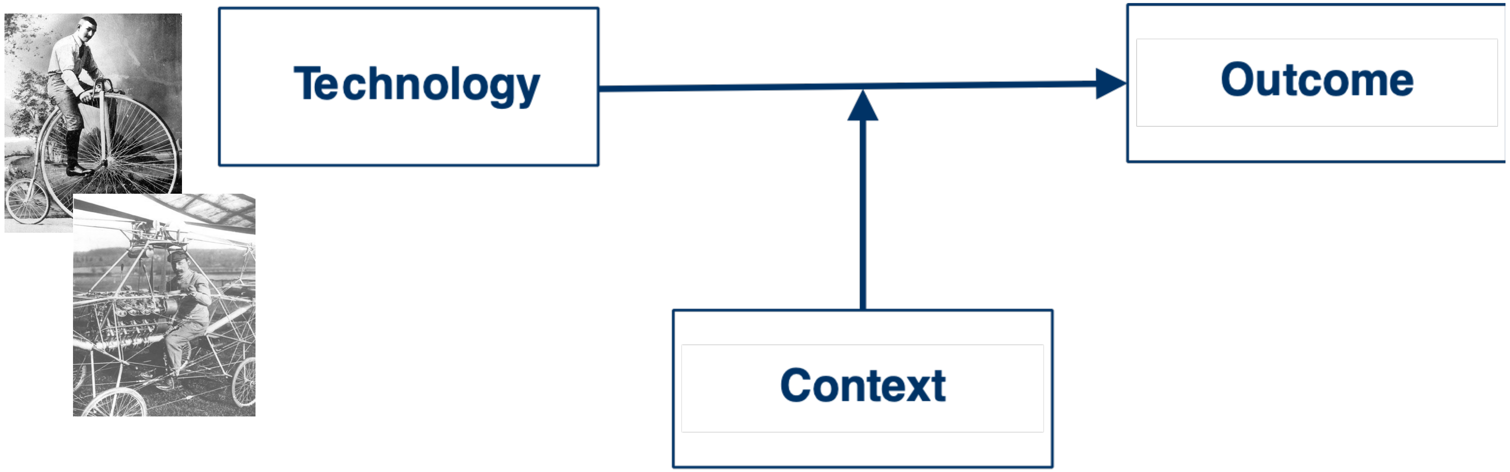
Using one set of tasks to predict performance on another set of tasks is not new: Anderson studied the acquisition of skills in LISP programming [2] and found that "the best predictor of individual differences in errors on problems that involved one LISP concept was number of errors on other problems that involved different concepts" (p. 203). Therefore, pretests appear to be better measures of skill than biographical variables, even though they require a lot more instrumentation.

Calls for better pretests for programmers can be traced back to at least 1980 [see 18]. Yet, in a 2009 literature review on quasi experiments in software engineering, only 42% of the reported 113 studies applied controls to account for potential selection bias [29]. Among the studies that applied controls, only three experiments involved actual

Proceedings of the EASE 2012 - Published by the IET
ISBN 978-1-84919-541-6

Structure

- Empirical research
- Evidence-based software engineering
  - The steps of EBSE
  - Research synthesis
- Empirical research methods
  - Controlled experiments
  - Case studies
  - Surveys
  - Action research
  - Validity
- The importance of context



Understand which technologies that cause which outcomes in which situations, e.g.:

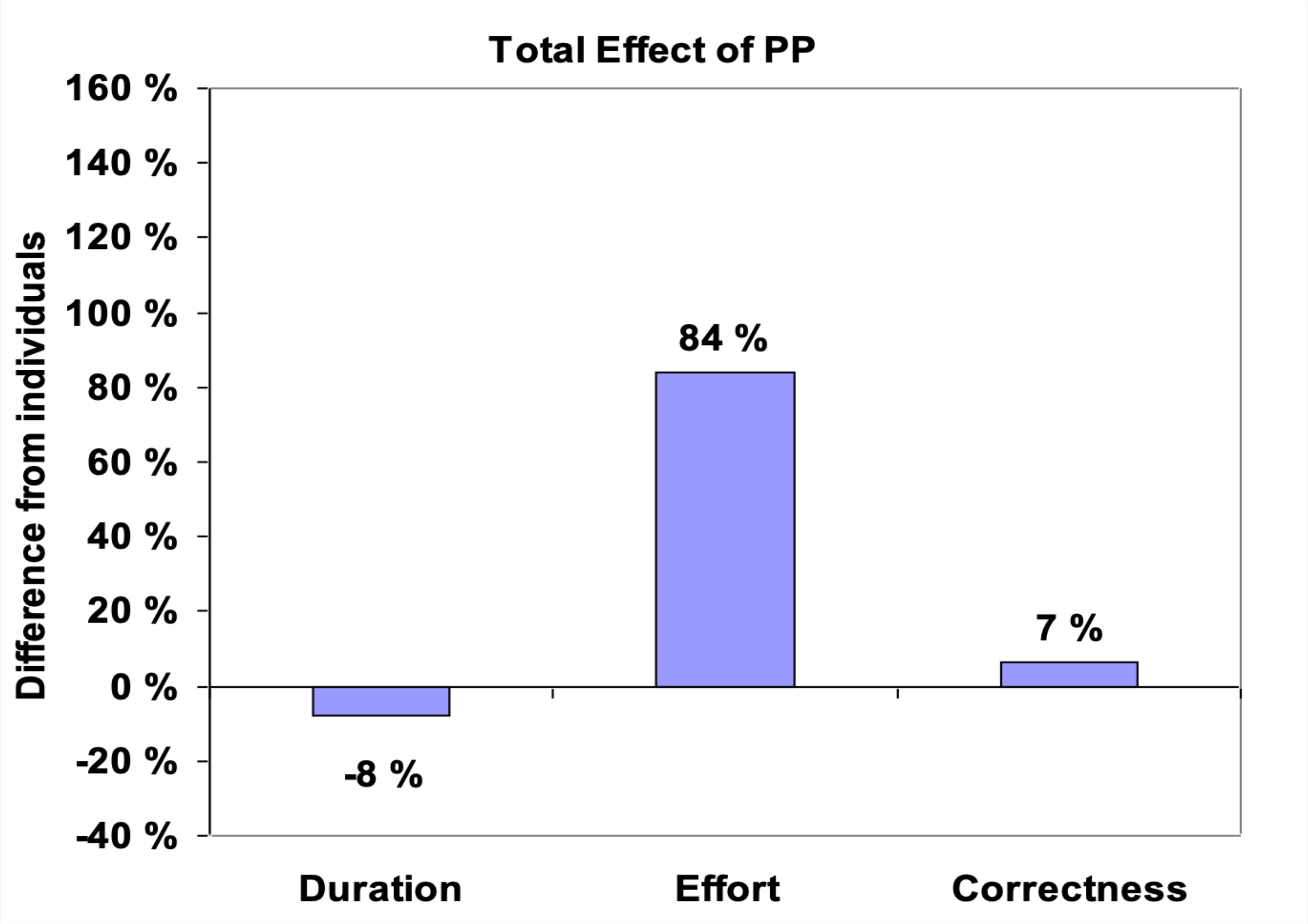
- When is technique X more efficient than technique Y?
- What resources are needed to use method X in a given situation?
- How to tailor process X to the actual situation?

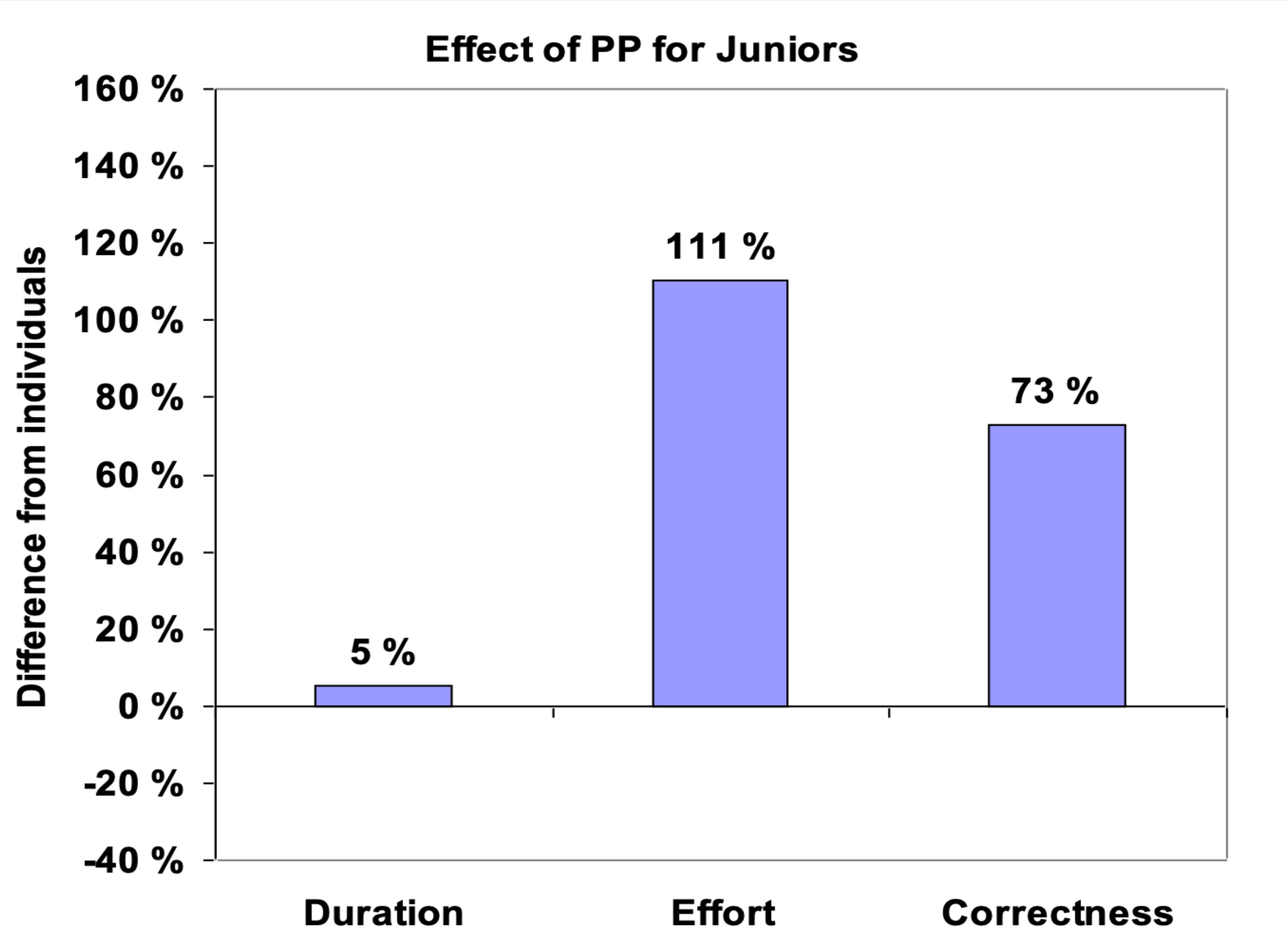
What is best?

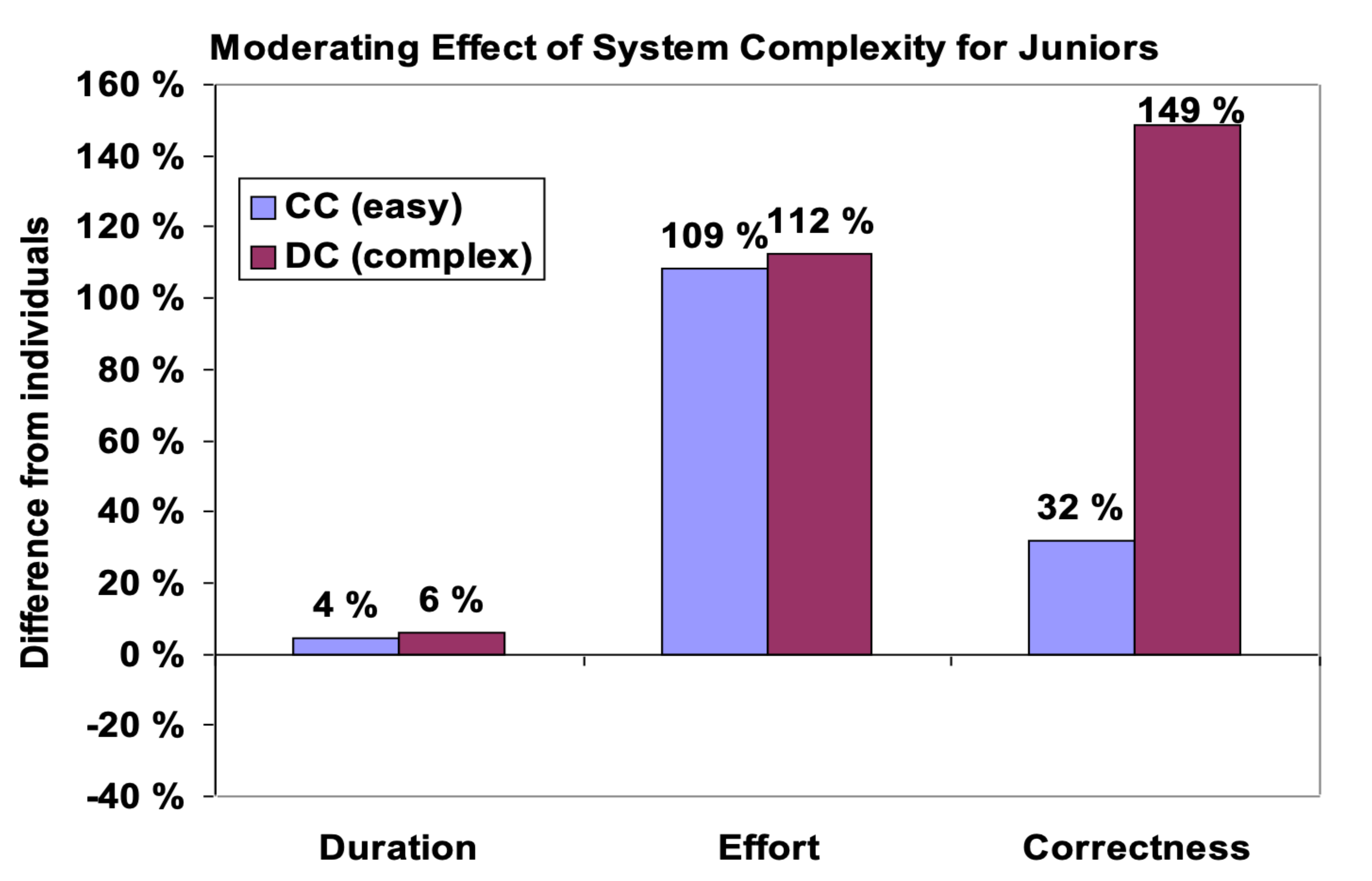
Pair programming or solo programming*

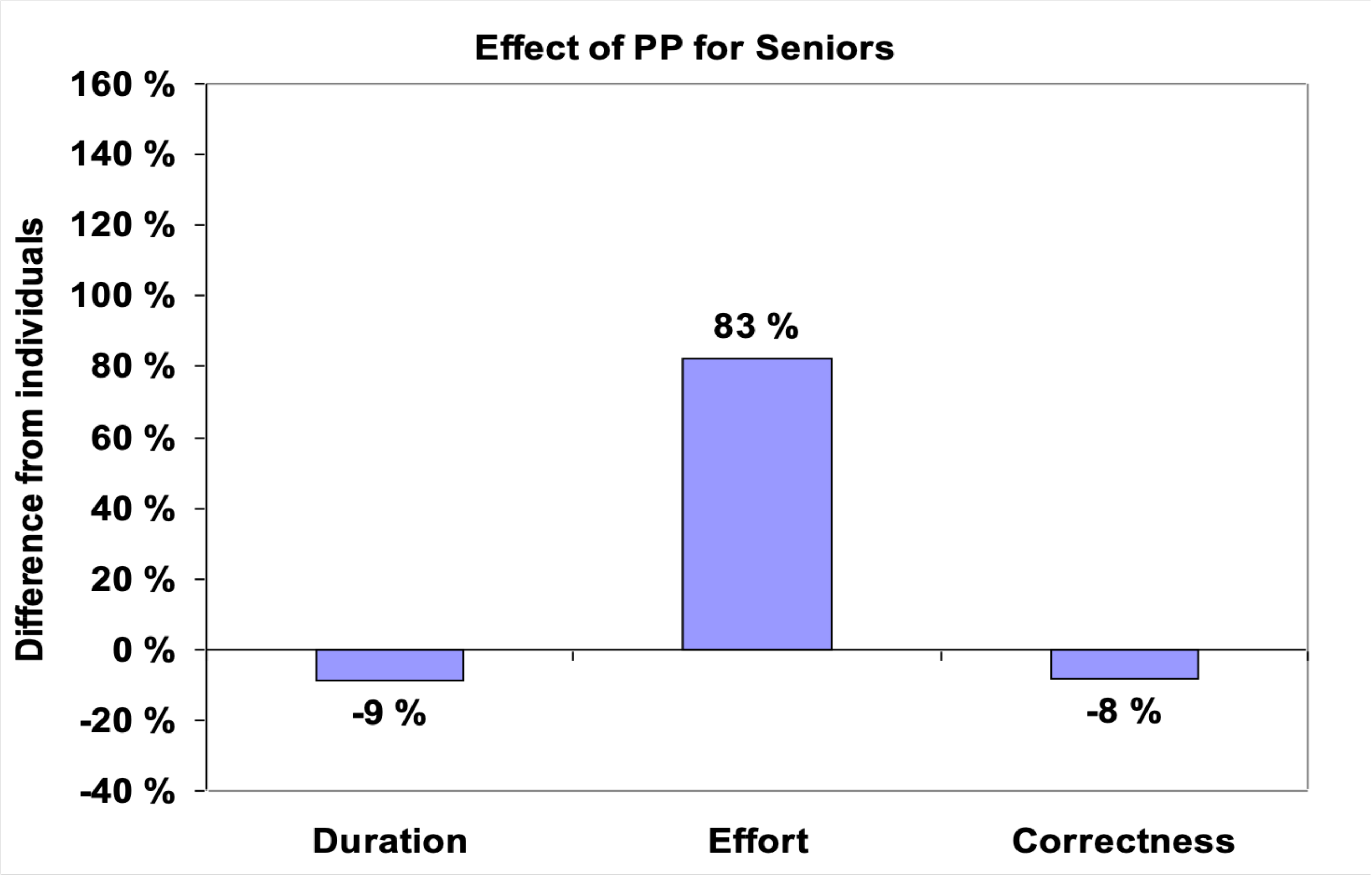
- 295 junior, intermediate and senior professional Java consultants from 29 companies were paid to participate (one work day)
- 99 individuals; 98 pairs
- The pairs and individuals performed the same Java maintenance tasks on either:
  - a ”simple” system (centralized control style), or
  - a ”complex” system (delegated control style)
- We measured:
  - duration (elapsed time)
  - effort (cost)
  - quality (correctness) of their solutions

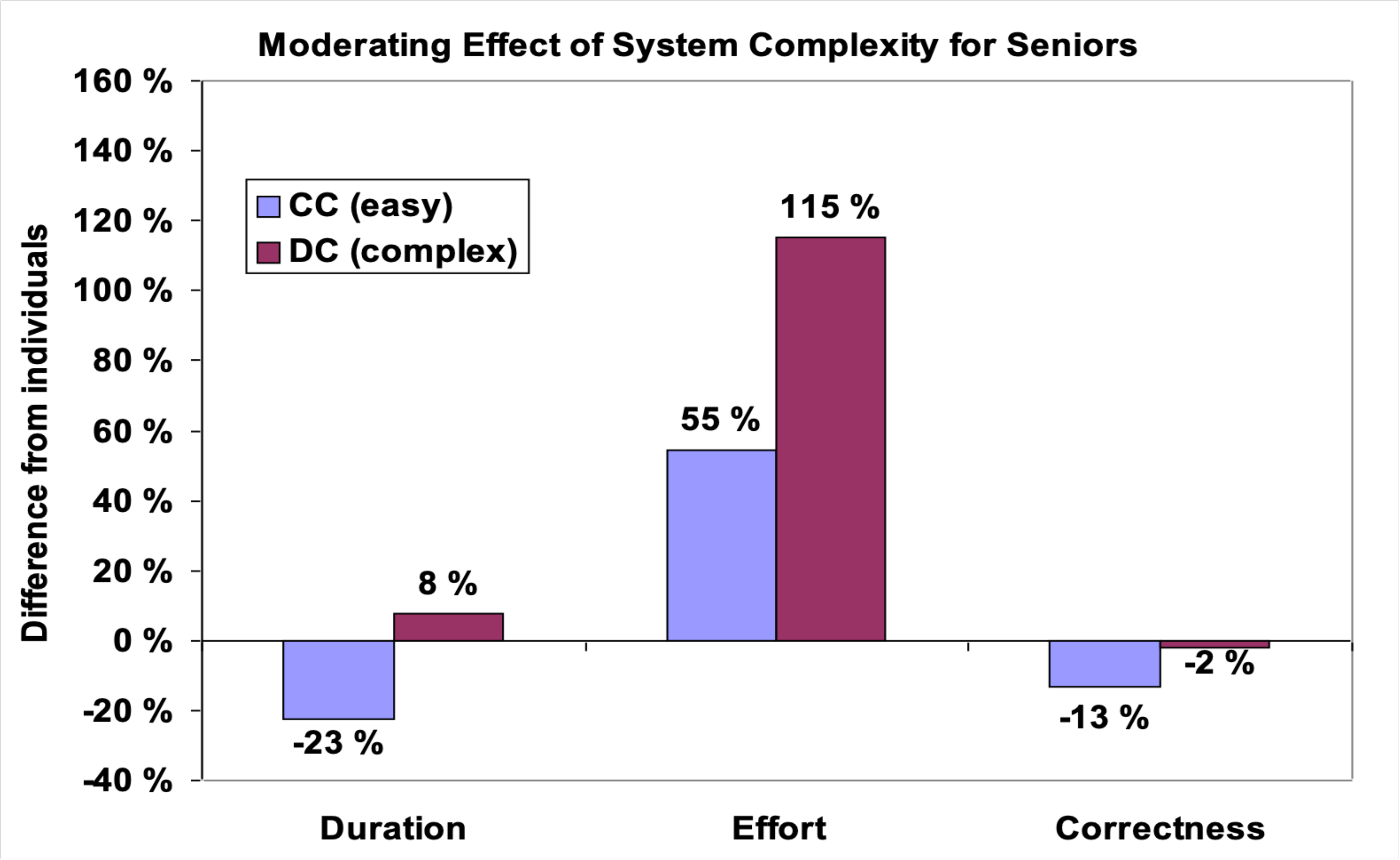
*E. Arisholm, H. Gallis, T. Dybå, and D. Sjøberg, “Evaluating Pair Programming with Respect to System Complexity and Programmer Expertise,” *IEEE Transactions on Software Engineering*, 2007, 33(2): 65-86.


Total Effect of PP
Difference from individuals
160 %
140 %
120 %
100 %
80 %
60 %
40 %
20 %
0 %
-20 %
-40 %
-8 %
84 %
7 %
Duration
Effort
Correctness


Effect of PP for Juniors
Difference from individuals
160 %
140 %
120 %
100 %
80 %
60 %
40 %
20 %
0 %
-20 %
-40 %
5 %
111 %
73 %
Duration
Effort
Correctness


Moderating Effect of System Complexity for Juniors
160 %
140 %
120 %
100 %
80 %
60 %
40 %
20 %
0 %
-20 %
-40 %
Difference from individuals
CC (easy)
DC (complex)
4 %
6 %
109 %
112 %
32 %
149 %
Duration
Effort
Correctness


Effect of PP for Seniors
Difference from individuals
160 %
140 %
120 %
100 %
80 %
60 %
40 %
20 %
0 %
-20 %
-40 %
83 %
-9 %
-8 %
Duration
Effort
Correctness

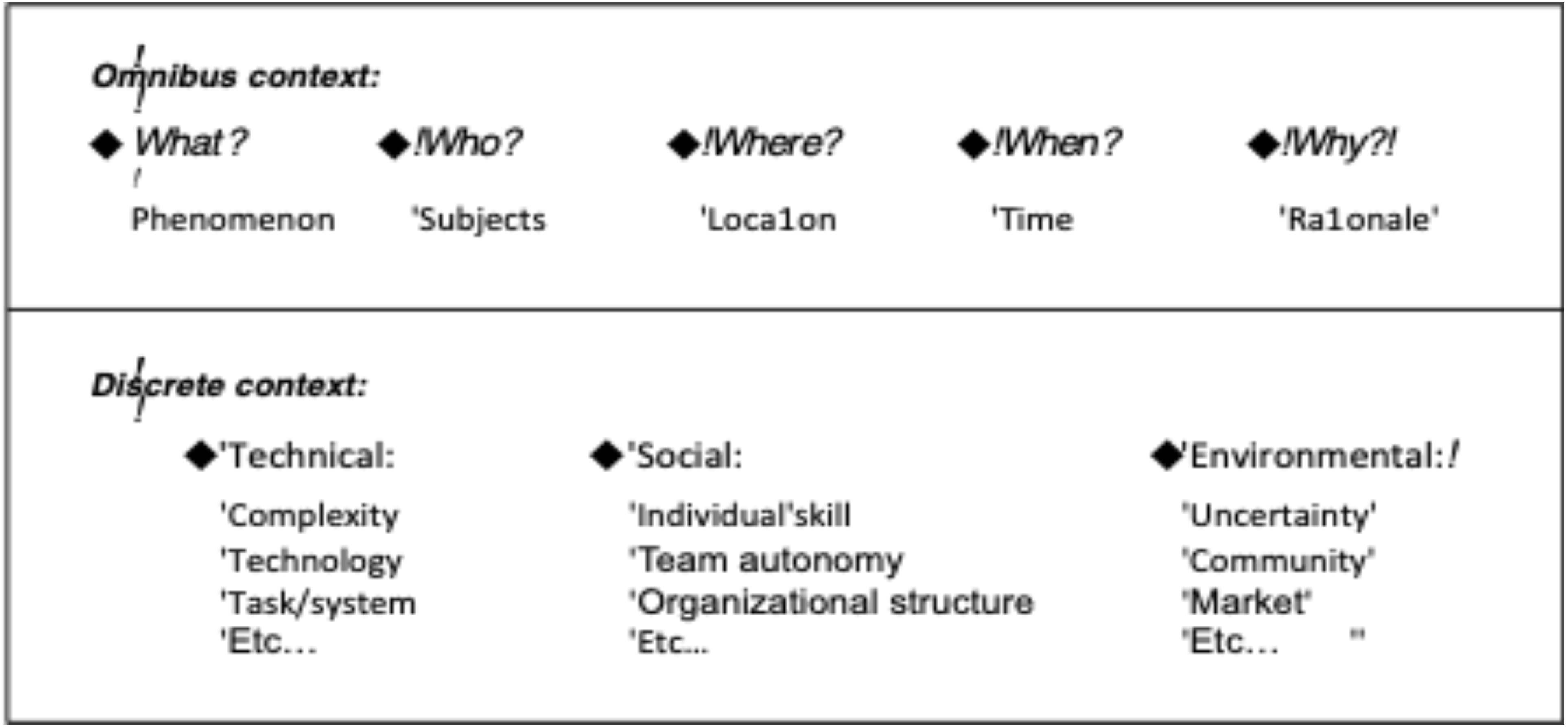

Moderating Effect of System Complexity for Seniors
Difference from individuals
160 %
140 %
120 %
100 %
80 %
60 %
40 %
20 %
0 %
-20 %
-40 %
CC (easy)
DC (complex)
-23 %
8 %
55 %
115 %
-13 %
-2 %
Duration
Effort
Correctness

So, when should we use PP?

Programmer Expertise	Task Complexity	Use PP?	Comments
Junior	Easy	Yes	Provided that increased quality is the main goal
	Complex	Yes	Provided that increased quality is the main goal
Intermediate	Easy	No	
	Complex	Yes	Provided that increased quality is the main goal
Expert	Easy	No	
	Complex	No	Unless you are sure that the task is too complex to be solved satisfactorily even by solo seniors

The question of whether PP is beneficial, or not, is meaningless!

Important dimensions of SE context



Omnibus context:

◆ *What ?* — Phenomenon

◆*!Who?* — 'Subjects

◆*!Where?* — 'Loca1on

◆*!When?* — 'Time

◆*!Why?!* — 'Ra1onale'

Discrete context:

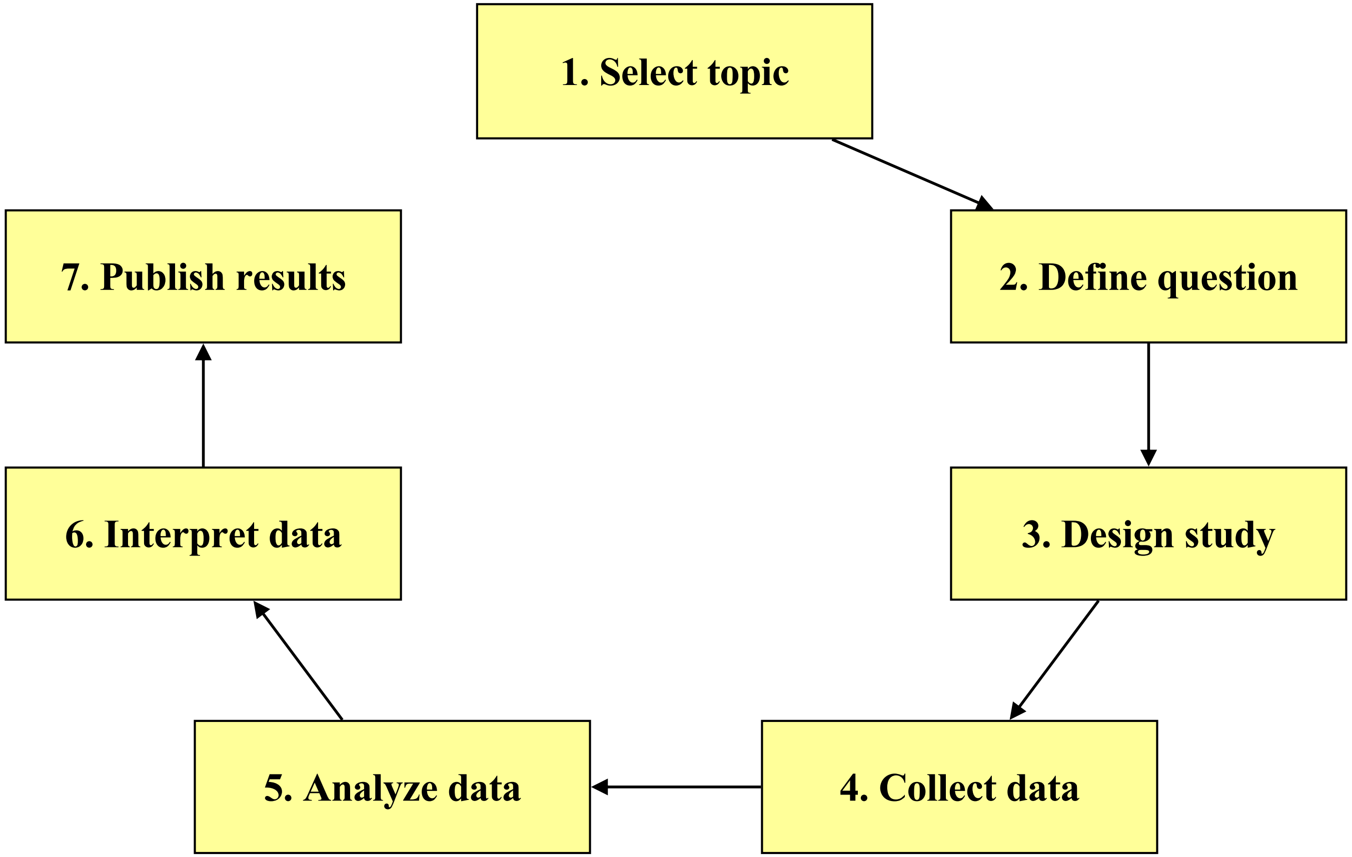
◆'Technical:
'Complexity
'Technology
'Task/system
'Etc…

◆'Social:
'Individual'skill
'Team autonomy
'Organizational structure
'Etc...

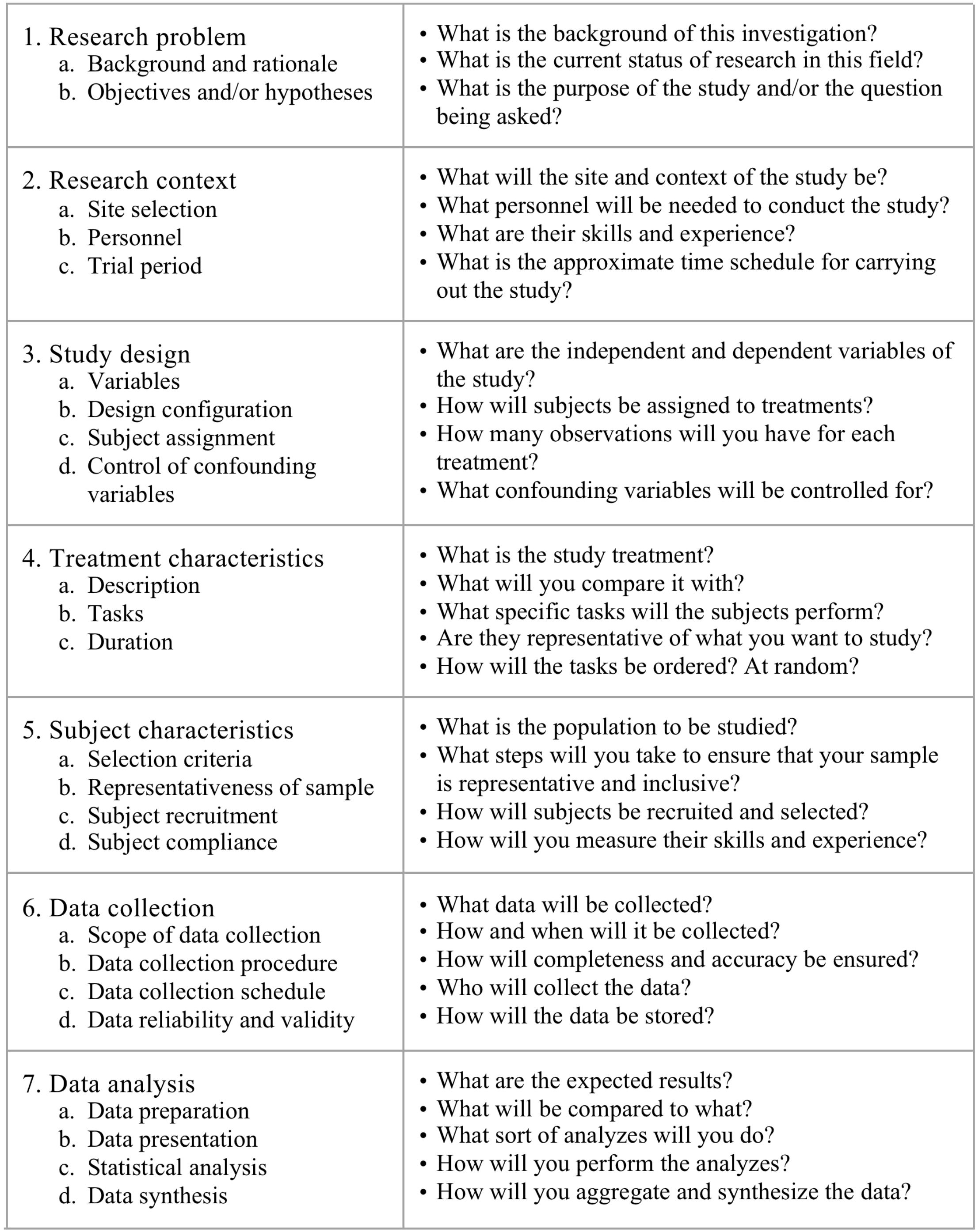
◆'Environmental:!
'Uncertainty'
'Community'
'Market'
'Etc… "

Dybå, T., Sjøberg, D.I.K., and Cruzes, D.S. (2012) "What Works for Whom, Where, When, and Why? On the Role of Context in Empirical Software Engineering," *Proceedings, ESEM 2012*, pp. 19-28.

The research process



1. Select topic
2. Define question
3. Design study
4. Collect data
5. Analyze data
6. Interpret data
7. Publish results



1. Research problem a. Background and rationale b. Objectives and/or hypotheses	• What is the background of this investigation? • What is the current status of research in this field? • What is the purpose of the study and/or the question being asked?
2. Research context a. Site selection b. Personnel c. Trial period	• What will the site and context of the study be? • What personnel will be needed to conduct the study? • What are their skills and experience? • What is the approximate time schedule for carrying out the study?
3. Study design a. Variables b. Design configuration c. Subject assignment d. Control of confounding variables	• What are the independent and dependent variables of the study? • How will subjects be assigned to treatments? • How many observations will you have for each treatment? • What confounding variables will be controlled for?
4. Treatment characteristics a. Description b. Tasks c. Duration	• What is the study treatment? • What will you compare it with? • What specific tasks will the subjects perform? • Are they representative of what you want to study? • How will the tasks be ordered? At random?
5. Subject characteristics a. Selection criteria b. Representativeness of sample c. Subject recruitment d. Subject compliance	• What is the population to be studied? • What steps will you take to ensure that your sample is representative and inclusive? • How will subjects be recruited and selected? • How will you measure their skills and experience?
6. Data collection a. Scope of data collection b. Data collection procedure c. Data collection schedule d. Data reliability and validity	• What data will be collected? • How and when will it be collected? • How will completeness and accuracy be ensured? • Who will collect the data? • How will the data be stored?
7. Data analysis a. Data preparation b. Data presentation c. Statistical analysis d. Data synthesis	• What are the expected results? • What will be compared to what? • What sort of analyzes will you do? • How will you perform the analyzes? • How will you aggregate and synthesize the data?

A research protocol is a detailed description of how and why the research will be carried out.

Selecting the research method

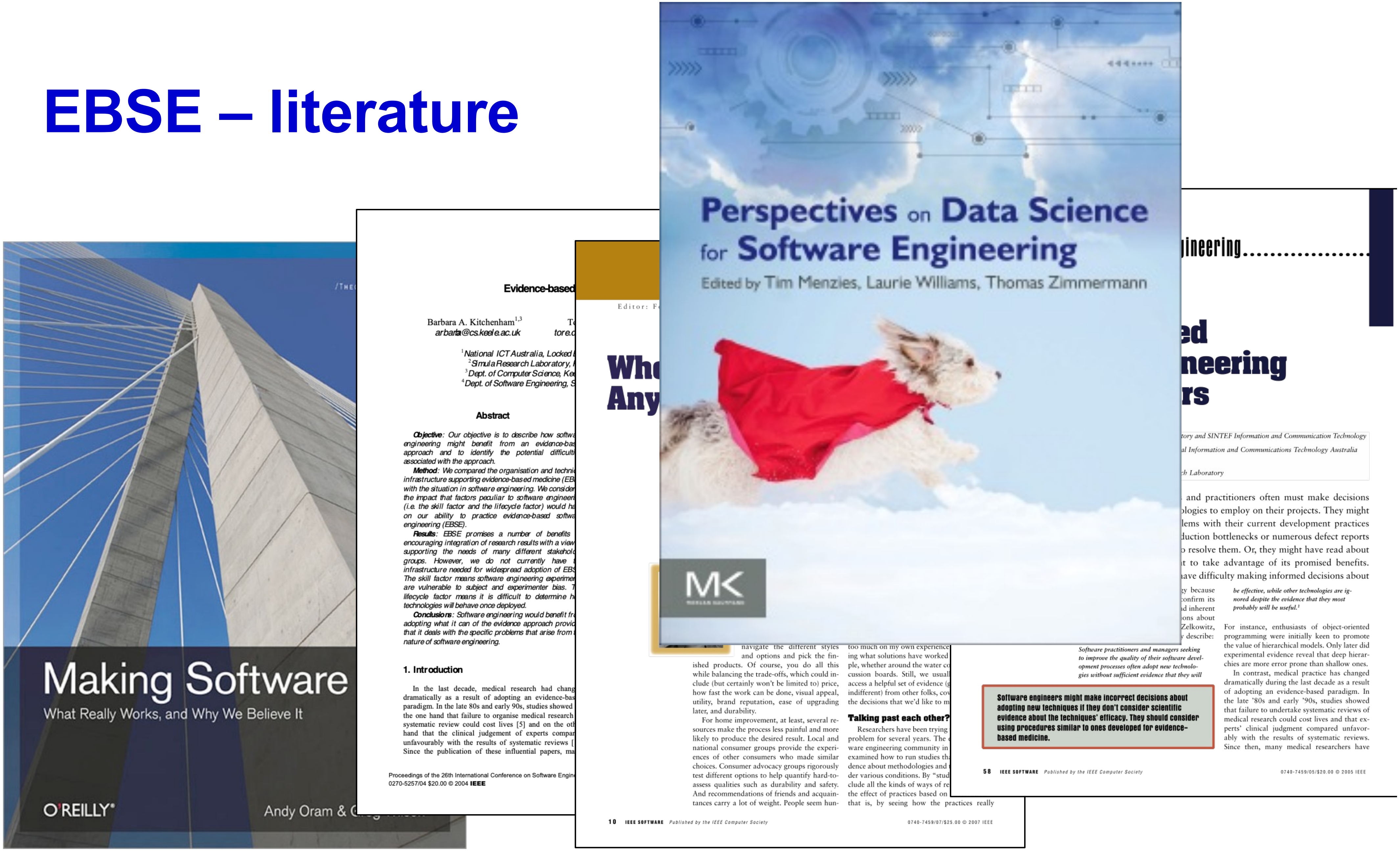
- The choice of method depends among other things on:
  - Suitable study subject (e.g., do participants have enough experience?)
  - Possibility to control the environment
  - The size/scale/cost of the study
  - The need for generality in the results
  - Availability of information/data and other resources
  - What is the purpose of the study? (exploration, prediction, understanding of cause-effect relations, applicability of results in industry, ...)
- Difficult to provide general recommendation with respect to choice of method, however …

Summary: empirical research methods

- Empirical research is a foundation for modern society
- Synthesis of evidence-based approaches (i.e., good studies) is best
- Different research methods can provide such evidence, albeit with different strengths and weaknesses

Combined, separating ”what works” from ”what doesn’t work” is a goal of both industry and academia.

EBSE – literature



/THE

Making Software

What Really Works, and Why We Believe It

O'REILLY®

Andy Oram & Greg Wilson

Evidence-based

Barbara A. Kitchenham[1,3]
arbarla@cs.keele.ac.uk

T
tore.c

[1] National ICT Australia, Locked
[2] Simula Research Laboratory,
[3] Dept. of Computer Science, Ke
[4] Dept. of Software Engineering, S

Abstract

Objective: *Our objective is to describe how softwa*
engineering might benefit from an evidence-bas
approach and to identify the potential difficulti
associated with the approach.

Method: *We compared the organisation and techni*
infrastructure supporting evidence-based medicine (EB
with the situation in software engineering. We conside
the impact that factors peculiar to software engineeri
(i.e. the skill factor and the lifecycle factor) would ha
on our ability to practice evidence-based softwa
engineering (EBSE).

Results: *EBSE promises a number of benefits*
encouraging integration of research results with a view
supporting the needs of many different stakehol
groups. However, we do not currently have t
infrastructure needed for widespread adoption of EBS
The skill factor means software engineering experime
are vulnerable to subject and experimenter bias. T
lifecycle factor means it is difficult to determine h
technologies will behave once deployed.

Conclusions: *Software engineering would benefit fr*
adopting what it can of the evidence approach provid
that it deals with the specific problems that arise from t
nature of software engineering.

1. Introduction

In the last decade, medical research had chang
dramatically as a result of adopting an evidence-bas
paradigm. In the late 80s and early 90s, studies showed
the one hand that failure to organise medical research
systematic review could cost lives [5] and on the oth
hand that the clinical judgement of experts compar
unfavourably with the results of systematic reviews [
Since the publication of these influential papers, ma

Proceedings of the 26th International Conference on Software Engin
0270-5257/04 $20.00 © 2004 **IEEE**

Editor: F

Wh
Any

navigate the different styles
and options and pick the finished products. Of course, you do all this while balancing the trade-offs, which could include (but certainly won't be limited to) price, how fast the work can be done, visual appeal, utility, brand reputation, ease of upgrading later, and durability.

For home improvement, at least, several resources make the process less painful and more likely to produce the desired result. Local and national consumer groups provide the experiences of other consumers who made similar choices. Consumer advocacy groups rigorously test different options to help quantify hard-to-assess qualities such as durability and safety. And recommendations of friends and acquaintances carry a lot of weight. People seem hun-

too much on my own experience
ing what solutions have worked
ple, whether around the water co
cussion boards. Still, we usuall
access a helpful set of evidence (g
indifferent) from other folks, cov
the decisions that we'd like to m

Talking past each other?

Researchers have been trying
problem for several years. The
ware engineering community in
examined how to run studies tha
dence about methodologies and
der various conditions. By "stud
clude all the kinds of ways of re
the effect of practices based on
that is, by seeing how the practices really

10 IEEE SOFTWARE *Published by the IEEE Computer Society* 0740-7459/07/$25.00 © 2007 IEEE

Perspectives on Data Science for Software Engineering

Edited by Tim Menzies, Laurie Williams, Thomas Zimmermann

MK
MORGAN KAUFMANN

gineering..................

ed
neering
rs

tory and SINTEF Information and Communication Technology
al Information and Communications Technology Australia
ch Laboratory

and practitioners often must make decisions
ologies to employ on their projects. They might
lems with their current development practices
duction bottlenecks or numerous defect reports
o resolve them. Or, they might have read about
it to take advantage of its promised benefits.
have difficulty making informed decisions about

gy because
confirm its
d inherent
ions about
Zelkowitz,
y describe:

> *Software practitioners and managers seeking to improve the quality of their software development processes often adopt new technologies without sufficient evidence that they will be effective, while other technologies are ignored despite the evidence that they most probably will be useful.*[1]

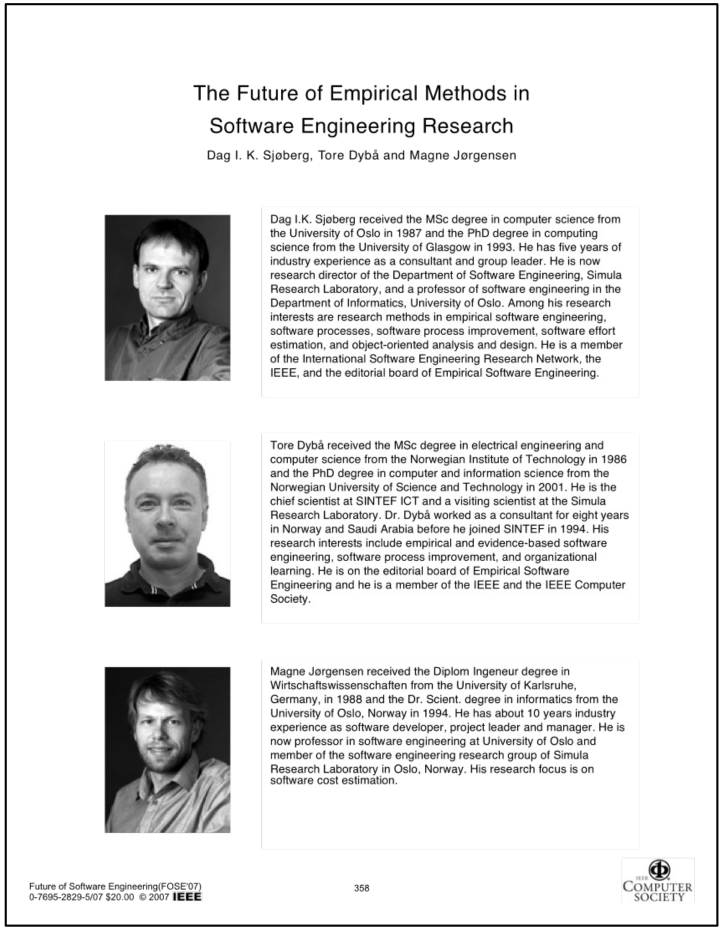
Software engineers might make incorrect decisions about adopting new techniques if they don't consider scientific evidence about the techniques' efficacy. They should consider using procedures similar to ones developed for evidence-based medicine.

For instance, enthusiasts of object-oriented programming were initially keen to promote the value of hierarchical models. Only later did experimental evidence reveal that deep hierarchies are more error prone than shallow ones.

In contrast, medical practice has changed dramatically during the last decade as a result of adopting an evidence-based paradigm. In the late '80s and early '90s, studies showed that failure to undertake systematic reviews of medical research could cost lives and that experts' clinical judgment compared unfavorably with the results of systematic reviews. Since then, many medical researchers have

58 IEEE SOFTWARE *Published by the IEEE Computer Society* 0740-7459/05/$20.00 © 2005 IEEE

Empirical research methods – literature



The Future of Empirical Methods in Software Engineering Research

Dag I. K. Sjøberg, Tore Dybå and Magne Jørgensen

Dag I.K. Sjøberg received the MSc degree in computer science from the University of Oslo in 1987 and the PhD degree in computing science from the University of Glasgow in 1993. He has five years of industry experience as a consultant and group leader. He is now research director of the Department of Software Engineering, Simula Research Laboratory, and a professor of software engineering in the Department of Informatics, University of Oslo. Among his research interests are research methods in empirical software engineering, software processes, software process improvement, software effort estimation, and object-oriented analysis and design. He is a member of the International Software Engineering Research Network, the IEEE, and the editorial board of Empirical Software Engineering.

Tore Dybå received the MSc degree in electrical engineering and computer science from the Norwegian Institute of Technology in 1986 and the PhD degree in computer and information science from the Norwegian University of Science and Technology in 2001. He is the chief scientist at SINTEF ICT and a visiting scientist at the Simula Research Laboratory. Dr. Dybå worked as a consultant for eight years in Norway and Saudi Arabia before he joined SINTEF in 1994. His research interests include empirical and evidence-based software engineering, software process improvement, and organizational learning. He is on the editorial board of Empirical Software Engineering and he is a member of the IEEE and the IEEE Computer Society.

Magne Jørgensen received the Diplom Ingeneur degree in Wirtschaftswissenschaften from the University of Karlsruhe, Germany, in 1988 and the Dr. Scient. degree in informatics from the University of Oslo, Norway in 1994. He has about 10 years industry experience as software developer, project leader and manager. He is now professor in software engineering at University of Oslo and member of the software engineering research group of Simula Research Laboratory in Oslo, Norway. His research focus is on software cost estimation.

Future of Software Engineering(FOSE'07)
0-7695-2829-5/07 $20.00 © 2007 IEEE

358

IEEE COMPUTER SOCIETY

Empir Software Eng (2011) 16:425–429
DOI 10.1007/s10664-011-9163-y

Qualitative research in software engineering

Tore Dybå · Rafael Prikladnicki · Kari Rönkkö ·
Carolyn Seaman · Jonathan Sillito

Published online: 28 May 2011
© Springer Science+Business Media, LLC 2011

Qualitative research methods were developed in the social sciences to enable researchers to study social and cultural phenomena and are designed to help researchers understand people and the social and cultural contexts within which they live (Denzin and Lincoln 2011). The goal of understanding a phenomenon from the point of view of the participants and its particular social and institutional context is largely lost when textual data are quantified. Taylor and Bogdan (1984) point out that qualitative research methods were designed mostly by educational researchers and other social scientists to study the complexities of human behavior (e.g., motivation, communication, difficulties in understanding). According to these authors, human behavior is clearly a phenomenon that, due to its complexity, requires qualitative methods to be fully understood, since much of human behavior cannot be adequately described and explained through statistics and other quantitative methods. Examples of qualitative methods are action research, case study research, ethnography, and grounded theory. Qualitative data sources include observation and participant observation (fieldwork), interviews and questionnaires, documents and texts, and the researcher's impressions and reactions.

Many in the software industry recognize that software development also presents a number of unique management and organizational issues that need to be addressed and solved in order for the field to progress. And this situation has led to studies related not only

T. Dybå
SINTEF, Trondheim, Norway

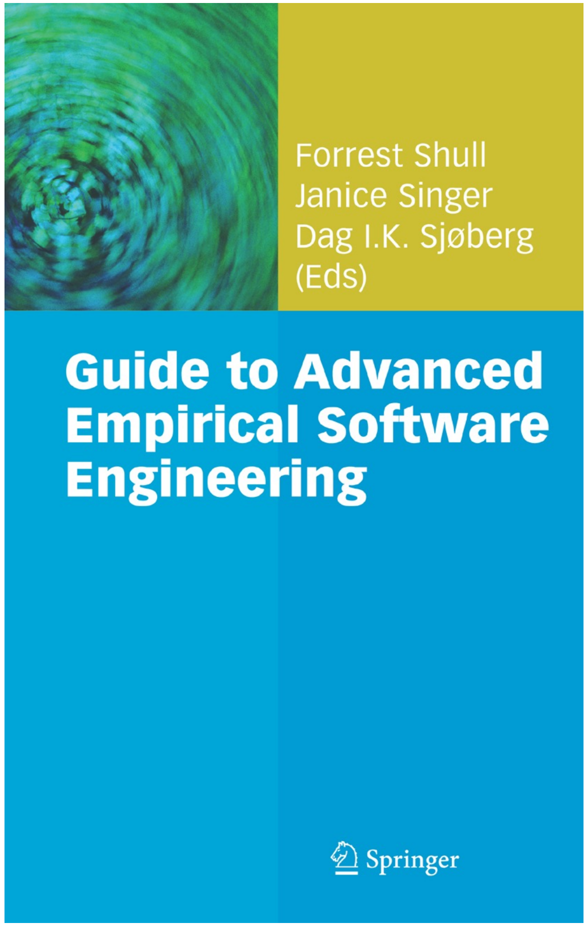
R. Prikladnicki
PUCRS, Porto Alegre, Brazil

K. Rönkkö
Blekinge Institute of Technology, Karlskrona, Sweden

C. Seaman (✉)
University of Maryland Baltimore County, Baltimore, MD, USA
e-mail: cseaman@umbc.edu

J. Sillito
University of Calgary, Calgary, Alberta, Canada

Springer



Forrest Shull
Janice Singer
Dag I.K. Sjøberg
(Eds)

Guide to Advanced Empirical Software Engineering

Springer