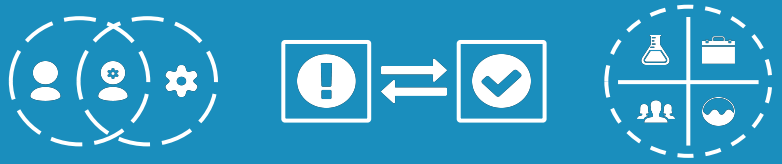


Embarking On a Journey to Conduct Disruptive Research in Software Engineering: *Who, What, How*

Margaret-Anne (Peggy) Storey

Siesta 2023

University of Victoria

Universität Zürich[UZH]

A Research Framework

- Who
- What
- How

Disruptive software development innovations

- **A Research Framework** *(some background)*
  - Who
  - What
  - How
- Disruptive software development innovations

What is empirical research?

1. : originating in or based on observation or experience. empirical data. 2. : relying on experience or observation alone often without due regard for system and theory.

26 Aug 2023

Merriam-Webster

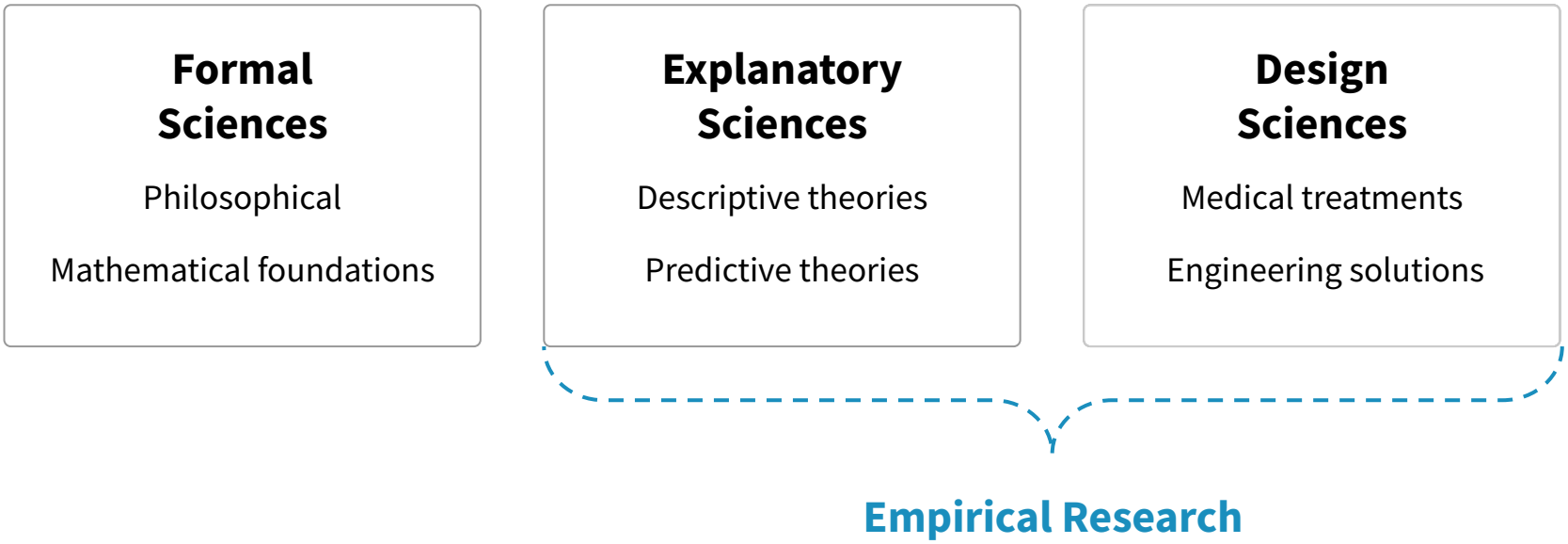
https://www.merriam-webster.com › dictionary › empi...

Empirical Definition & Meaning - Merriam-Webster

What is Empirical Evidence? Empirical evidence is the information obtained through observation and documentation of certain behavior and patterns or through an experiment. Empirical evidence is a quintessential part of the scientific method of research that is applicable in many disciplines. 11 May 2020

corporatefinanceinstitute.com

https://corporatefinanceinstitute.com › data-science › em...



6 | The Sciences

van Aken, 2004

Published: 01 November 2012

What are developers talking about? An analysis of topics and trends in Stack Overflow

Anton Barua, Stephen W. Thomas & Ahmed E. Hassan

Empirical Software Engineering **19**, 619–654 (2014) | Cite this article

Augmenting API documentation with insights from stack overflow

Authors:  Christoph Treude,  Martin P. Robillard Authors Info & Claims

ICSE '16: Proceedings of the 38th International Conference on Software Engineering • May 2016 • Pages 392–403

• https://doi.org/10.1145/2884781.2884800

Published: 14 May 2016 Publication History

Substantive domain (the *actors* we study - **who**)

Conceptual domain (the *ideas* we have about the actors we study - **what**)

Methodological domain (the *methods* we use - **how**)

McGrath - research in the behavioural sciences

Extending: Runkel & McGrath:
Research on Human Behavior: A Systematic Guide to Method, 1972

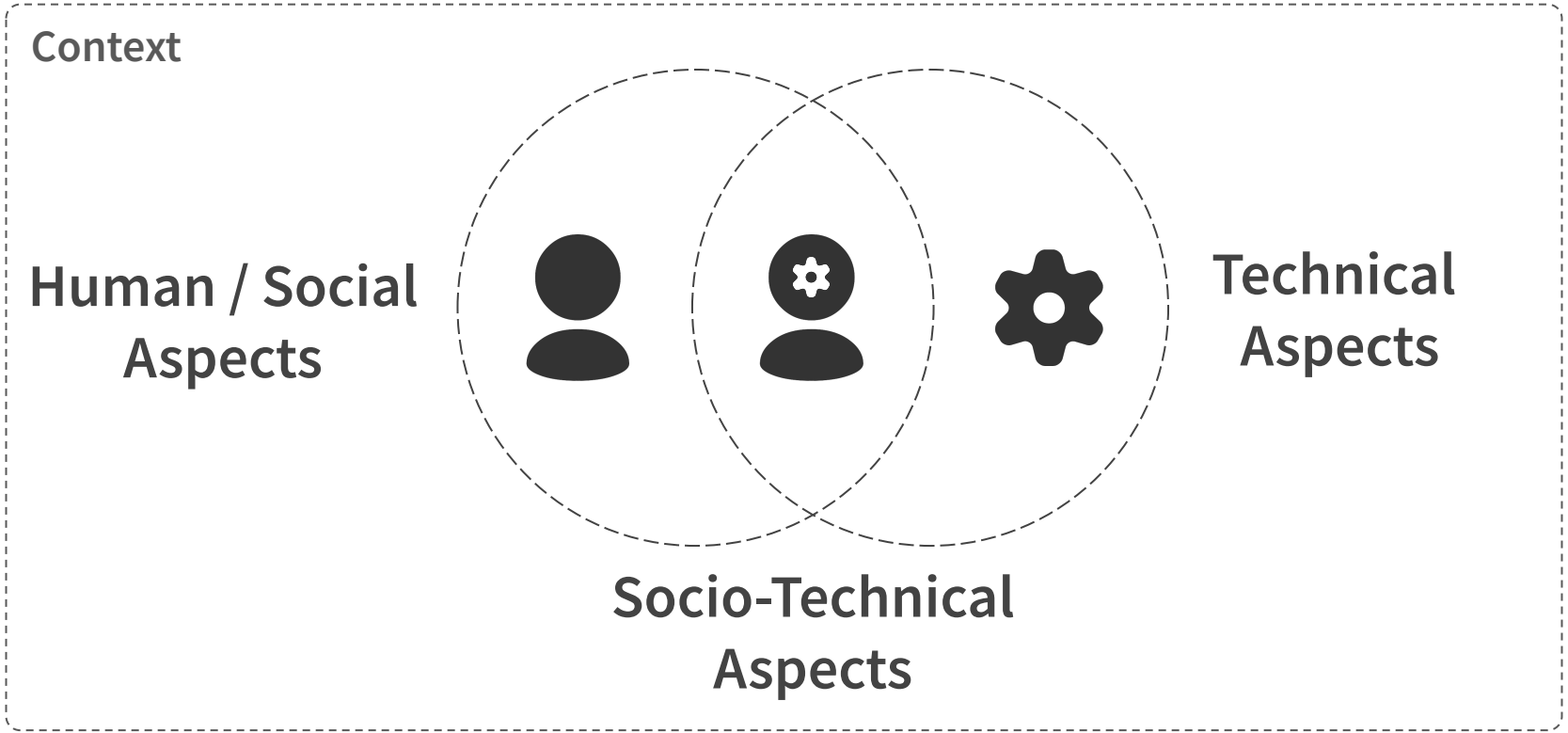
A Research Framework

Who (what *actors* do we study?)

What

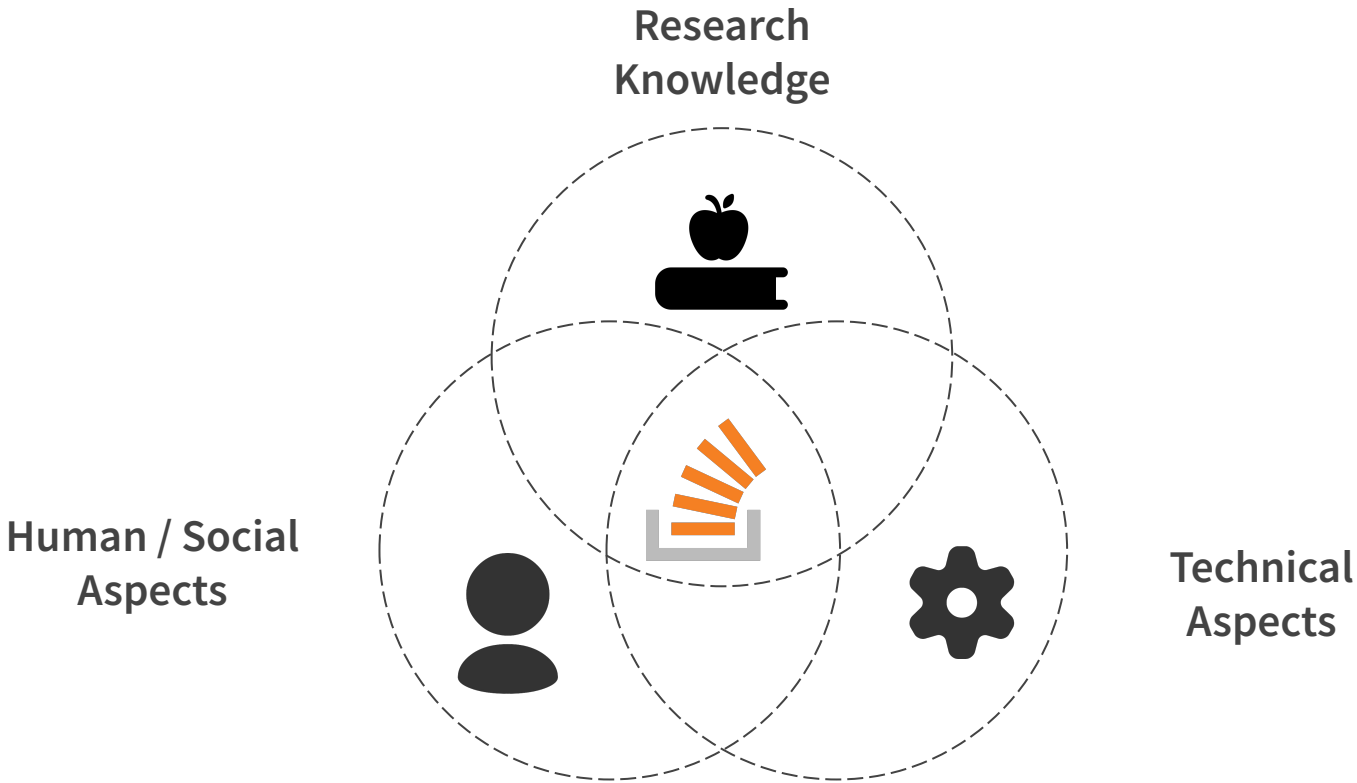
How

Disruptive software development innovations



 | Software Engineering **Design** Space (Substantive Domain)

Joint Optimization



13 | **Who** is the claimed beneficiary of our research?

A Research Framework

Who

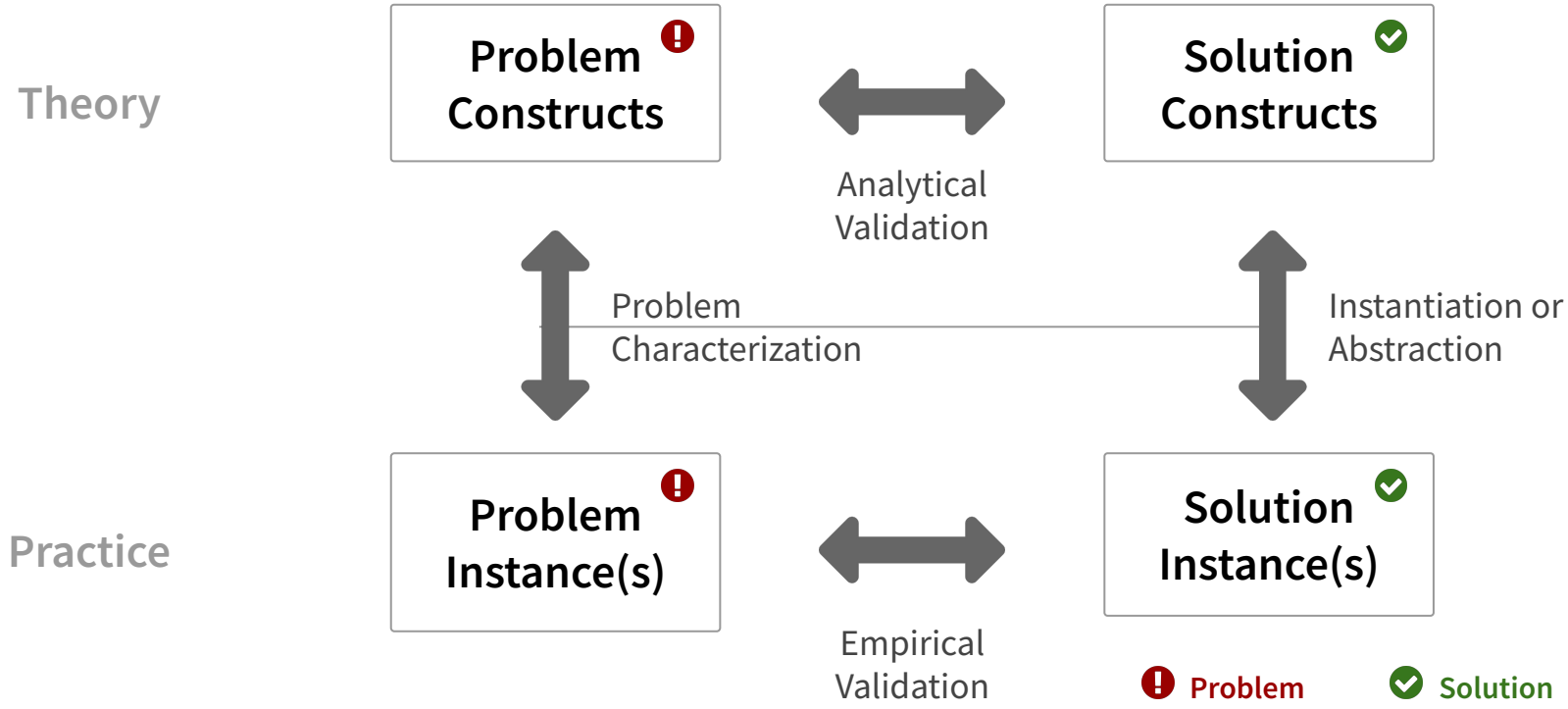
What *(which ideas and contributions do we study)*

How

Disruptive software development innovations

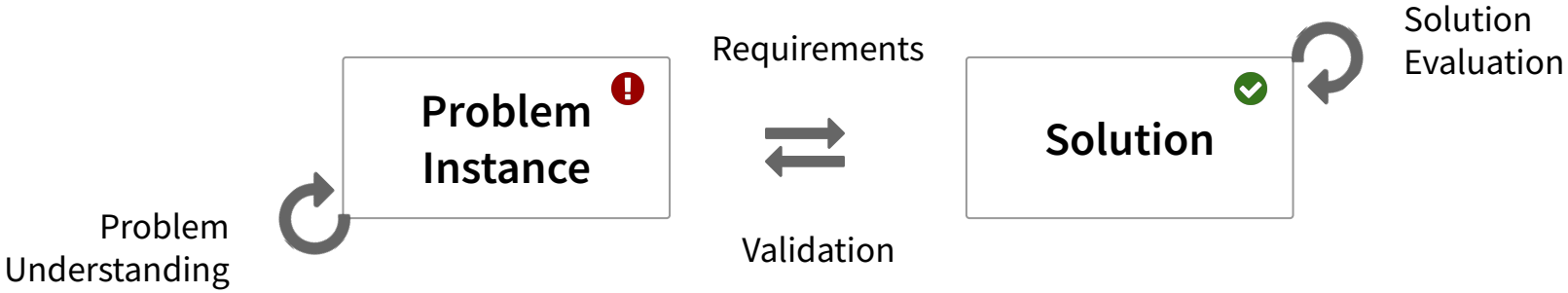
Design Science — Hevner (2007)

See also Wieringa's book



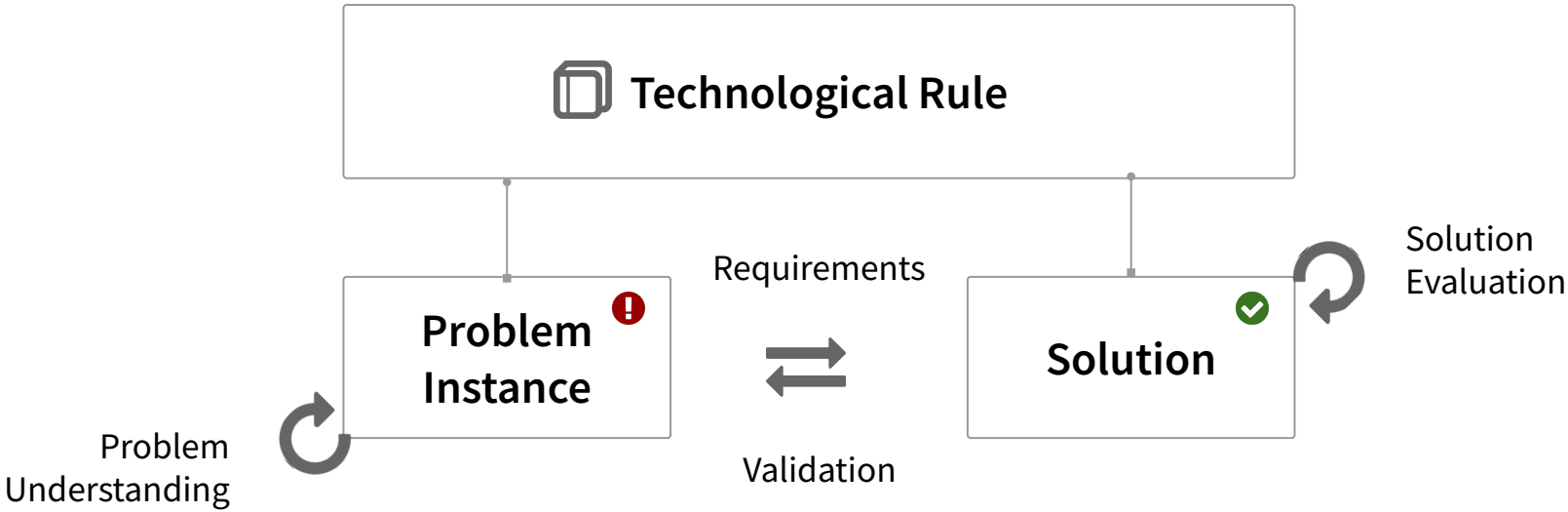
Design Science - From Theory to Practice

Engström, E., Storey, MA., Runeson, P., Host M., Baldasserra M. T., How software engineering research aligns with design science: a review. Empirical Software Eng 25, 2630–2660 (2020).



Design Science
A new view

Engström, E., Storey, MA., Runeson, P., Host M., Baldasserra M. T., How software engineering research aligns with design science: a review. Empirical Software Eng 25, 2630–2660 (2020).



van Aken, 2004

Technological Rule
(Theory Fragment)

To address the lack of API documentation

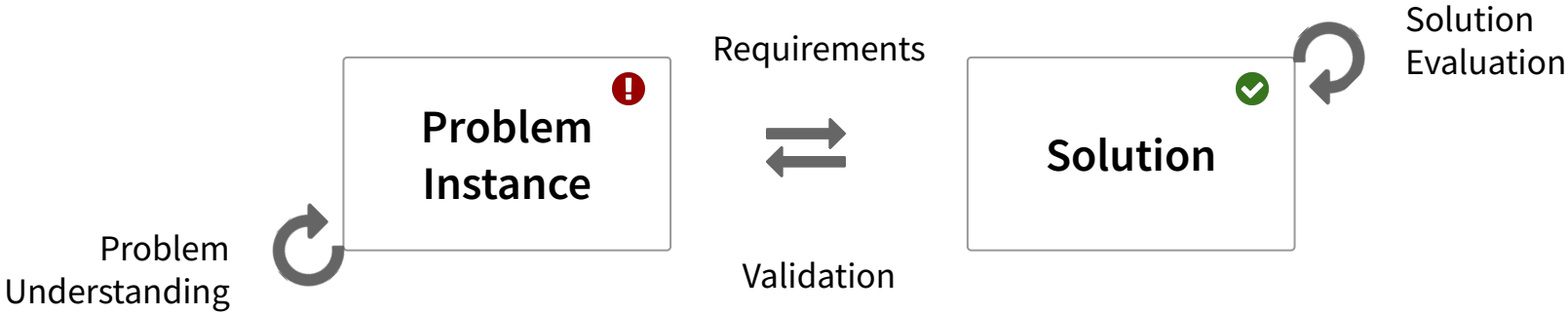
in open source projects

use insights from stack overflow

To achieve an effect in a given context **use / do intervention.**

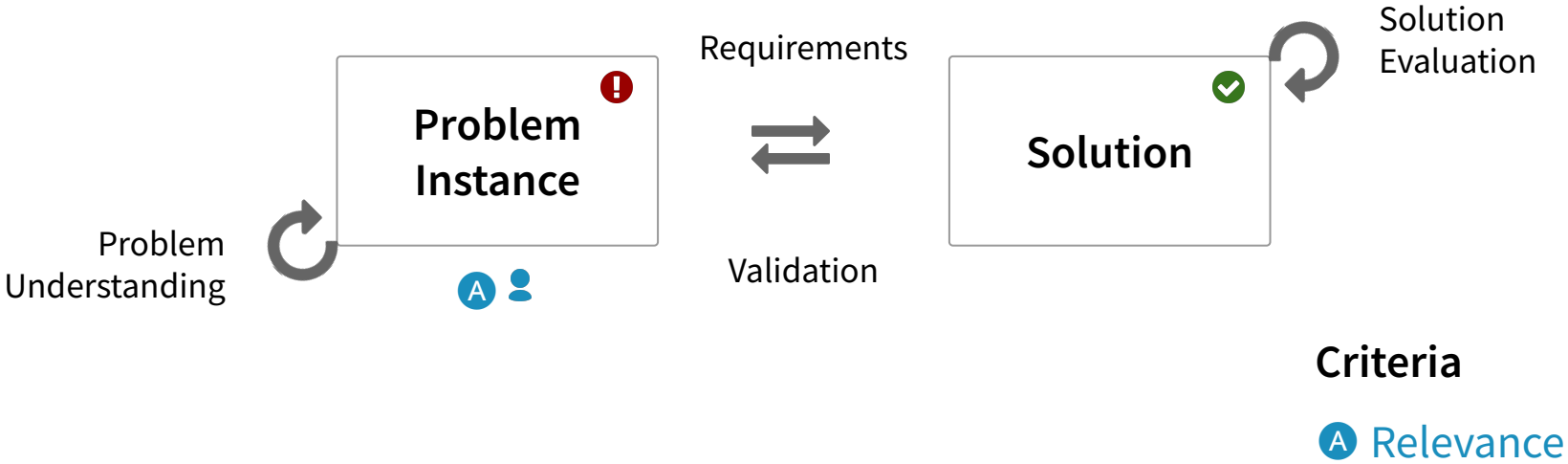
x y z

van Aken, 2004

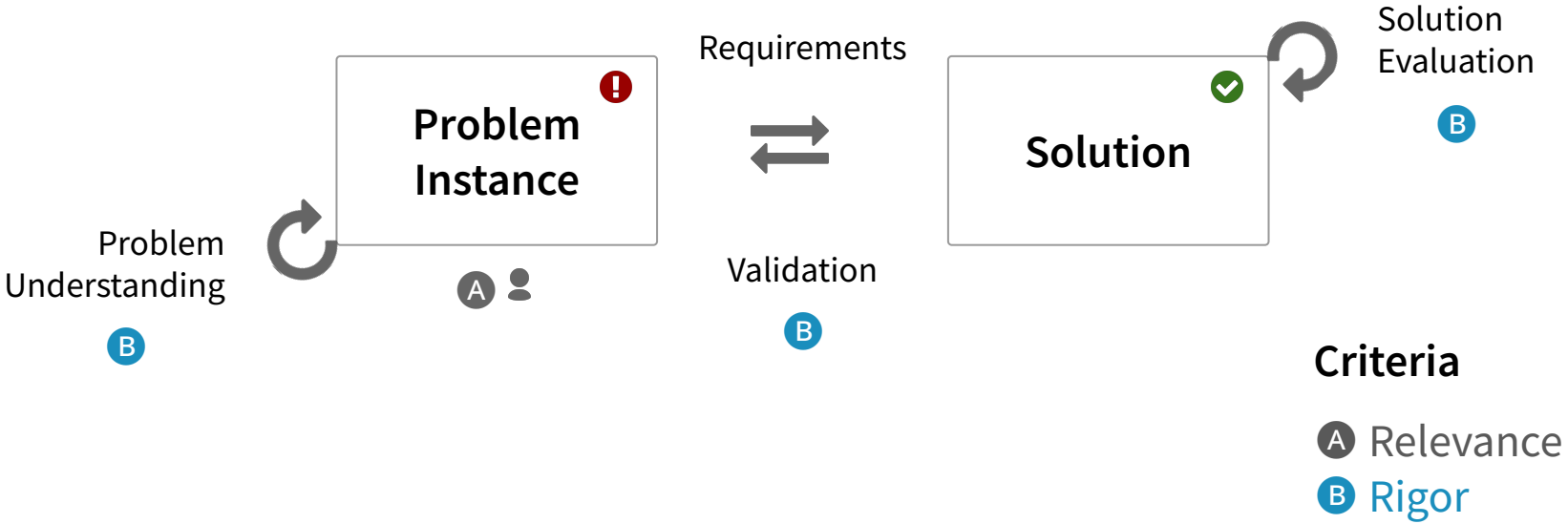


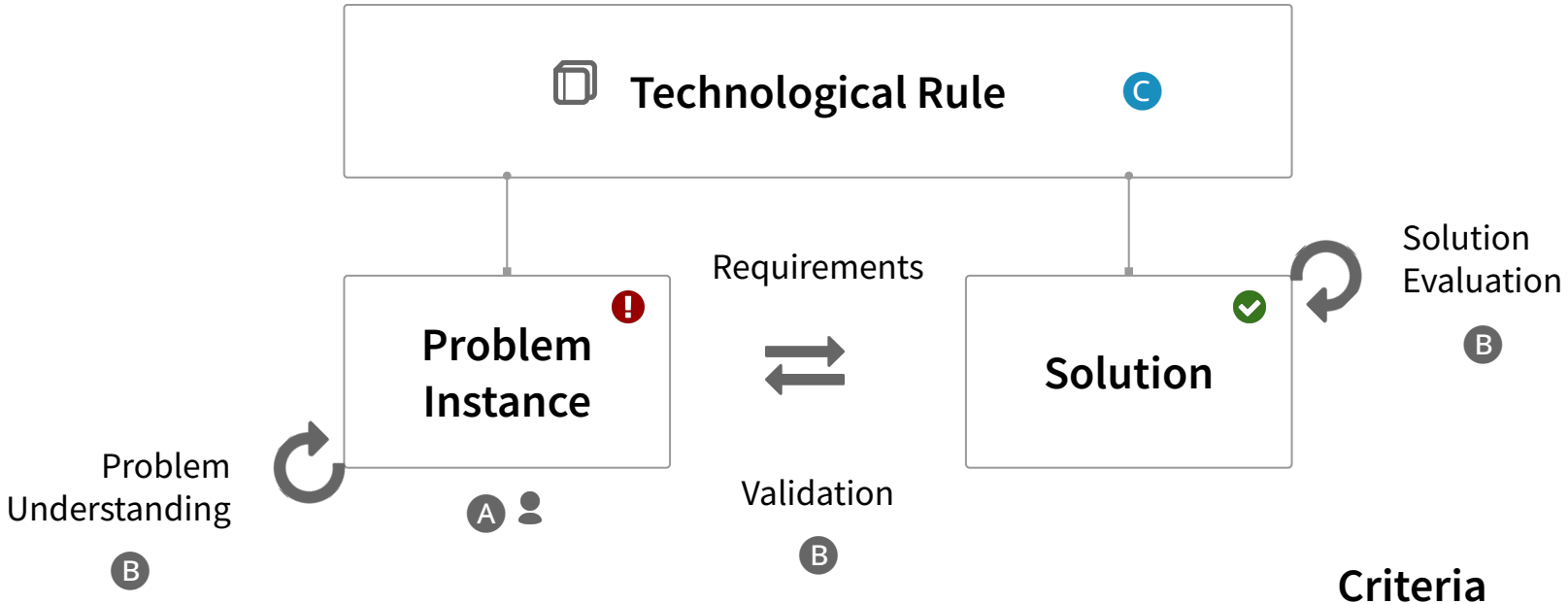
Evaluation Criteria

Storey et al., 2017



Evaluation Criteria


Problem Instance
Problem Understanding
B
A
Requirements
Validation
B
Solution
Solution Evaluation
B
Criteria
A Relevance
B Rigor



Criteria

- A Relevance
- B Rigor
- C Novelty

Evaluation Criteria

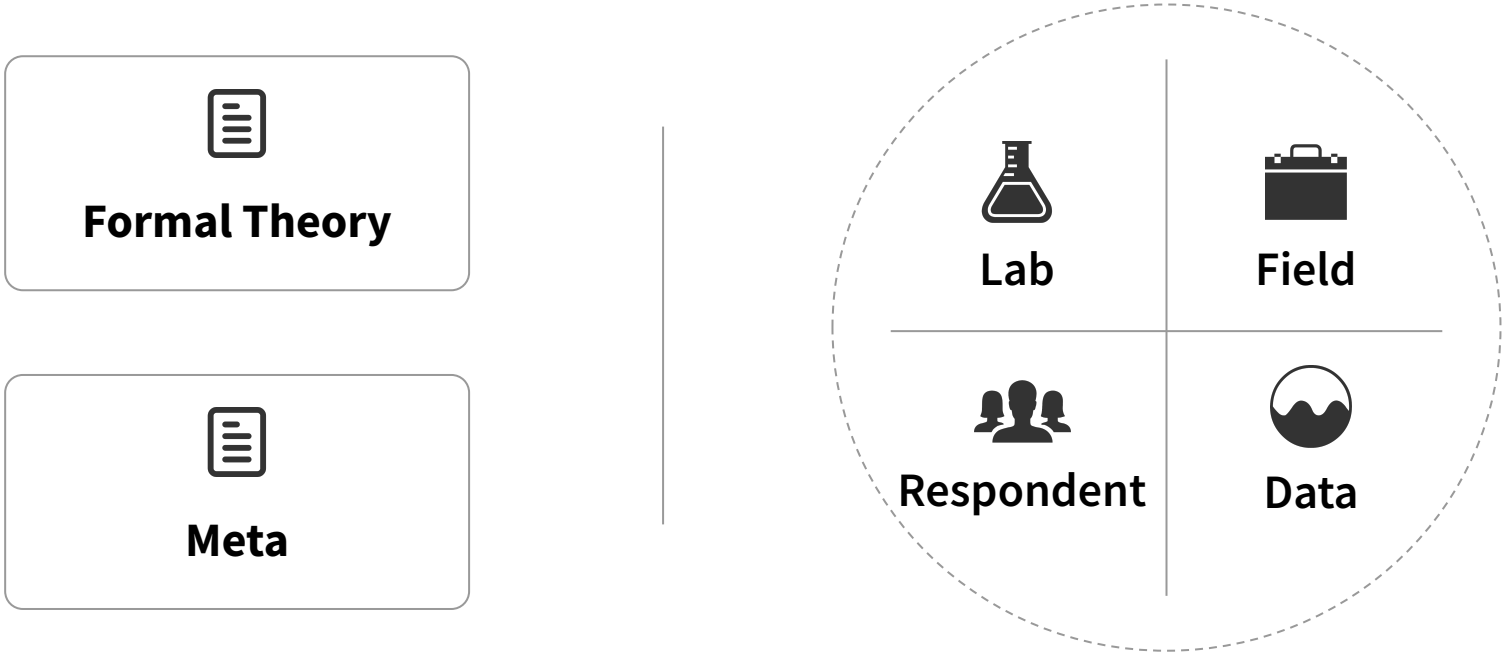
A Research Framework

- Who
- What
- **How** *(which methods do we use)*

Disruptive software development innovations

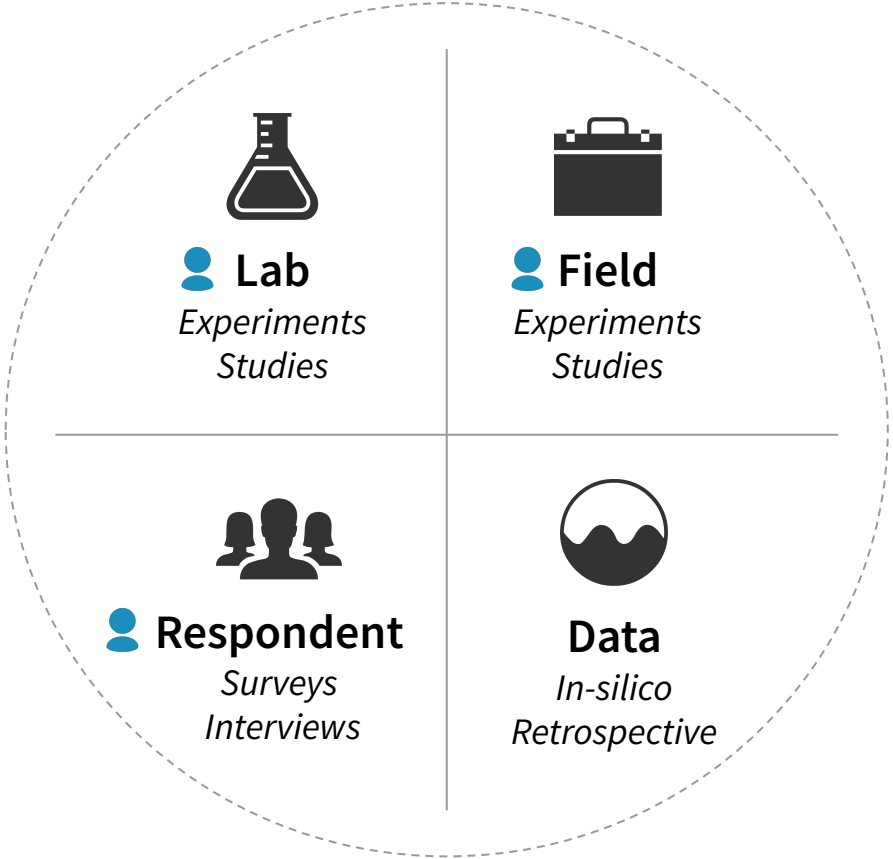
Non-Empirical

Empirical



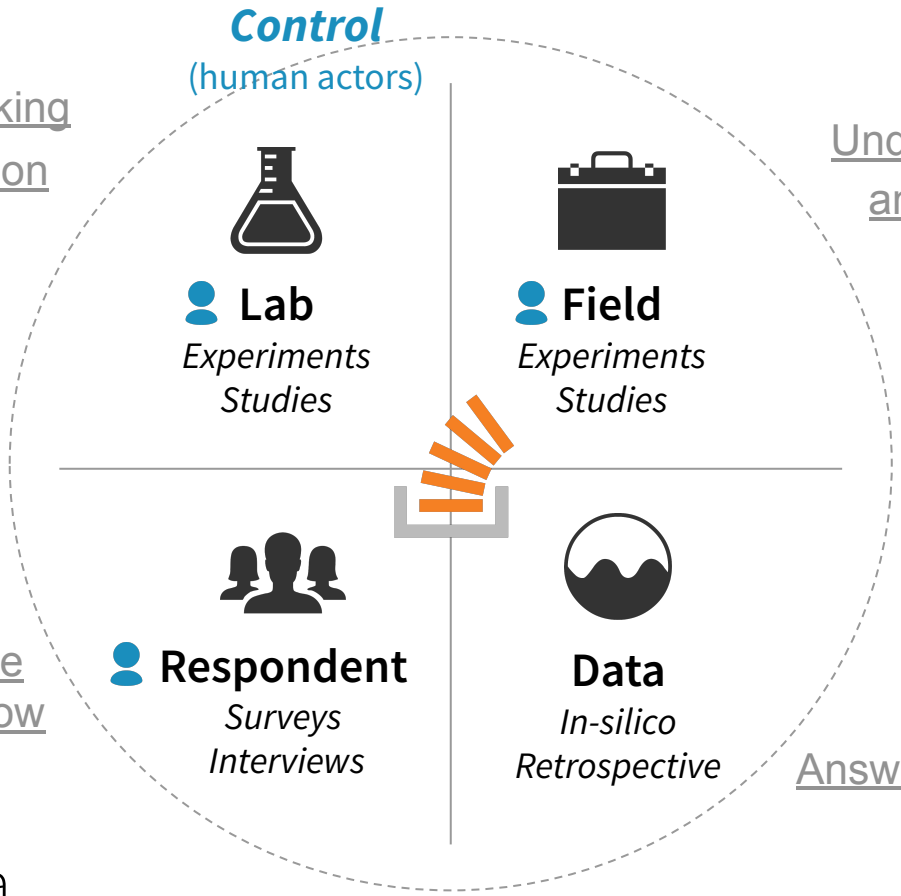
Research Strategies (methodological domain)

Extending: Runkel & McGrath:
Research on Human Behavior: A Systematic Guide to Method, 1972


Lab
Experiments
Studies
Field
Experiments
Studies
Respondent
Surveys
Interviews
Data
In-silico
Retrospective

You Get Where You're Looking for: The Impact of Information Sources on Code Security

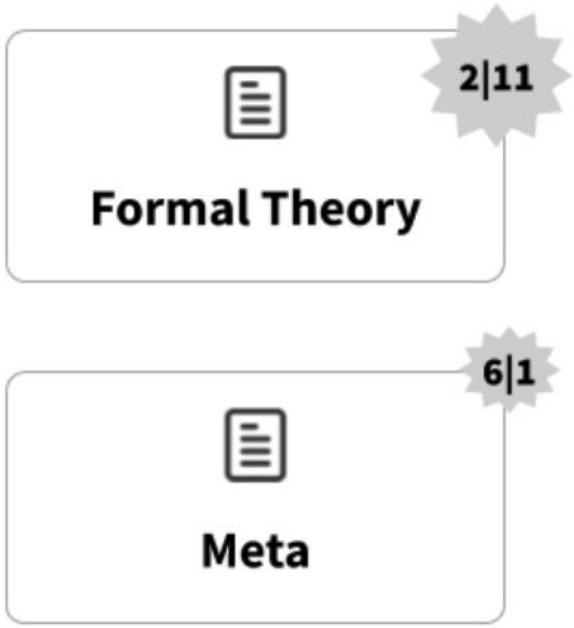
Signals Matter: Understanding Popularity and Impact of Users on Stack Overflow



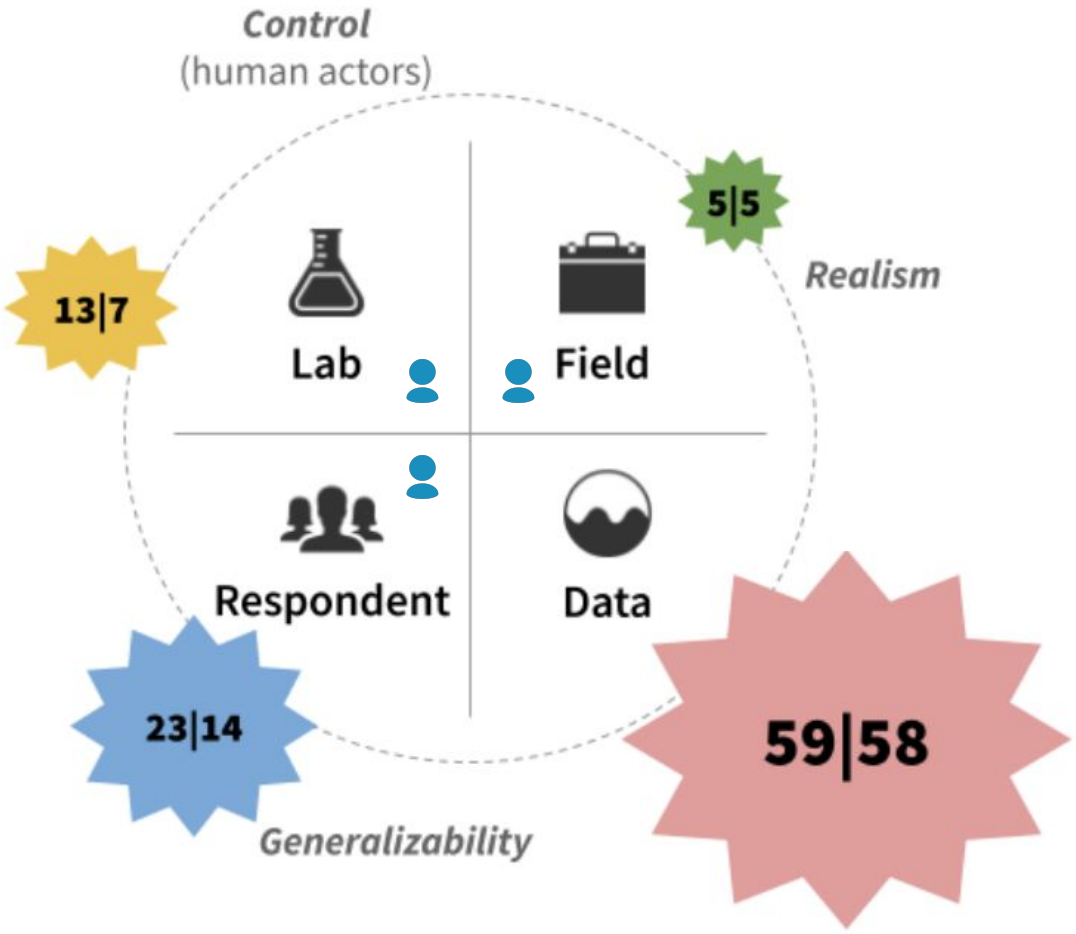
Paradise unplugged: identifying barriers for female participation on stack overflow

Mining Successful Answers in Stack Overflow

Non-Empirical Strategies:

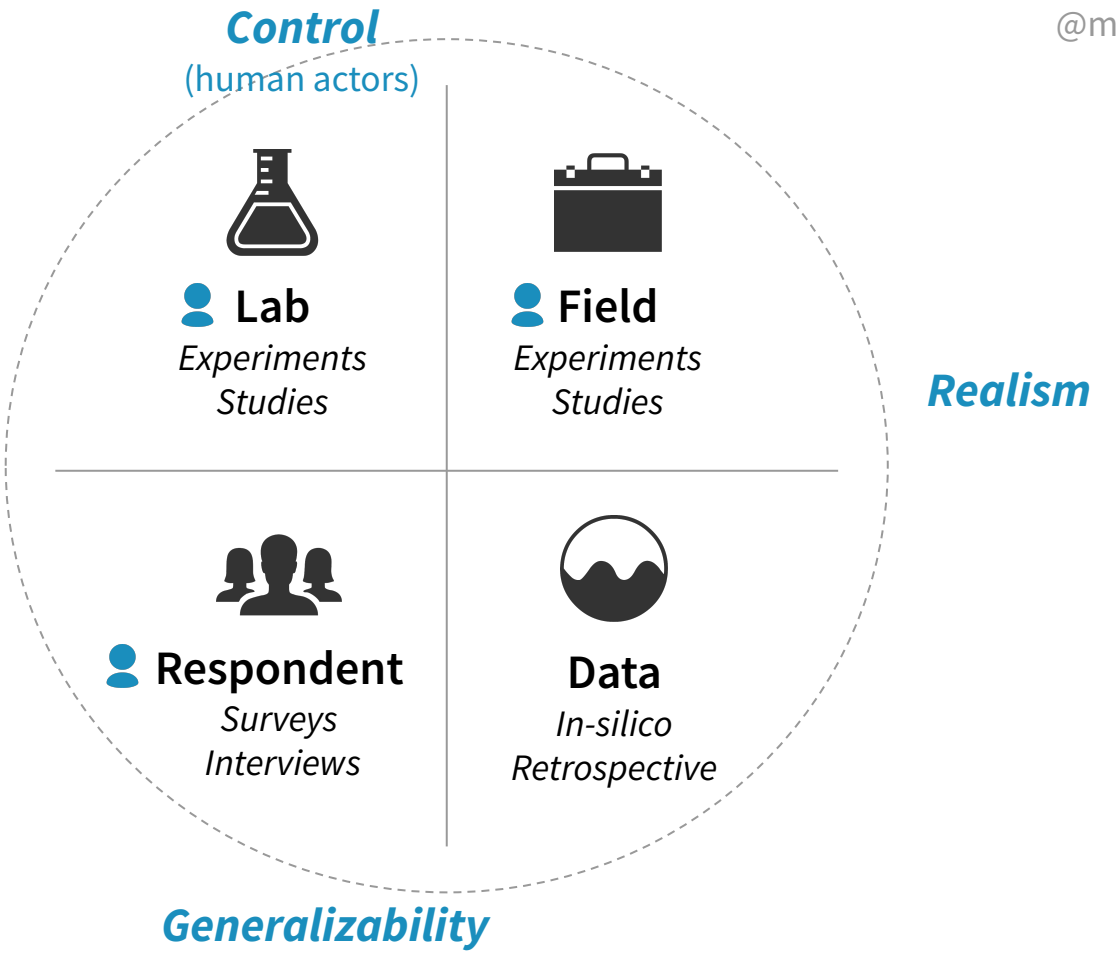


Legend: EMSE | ICSE

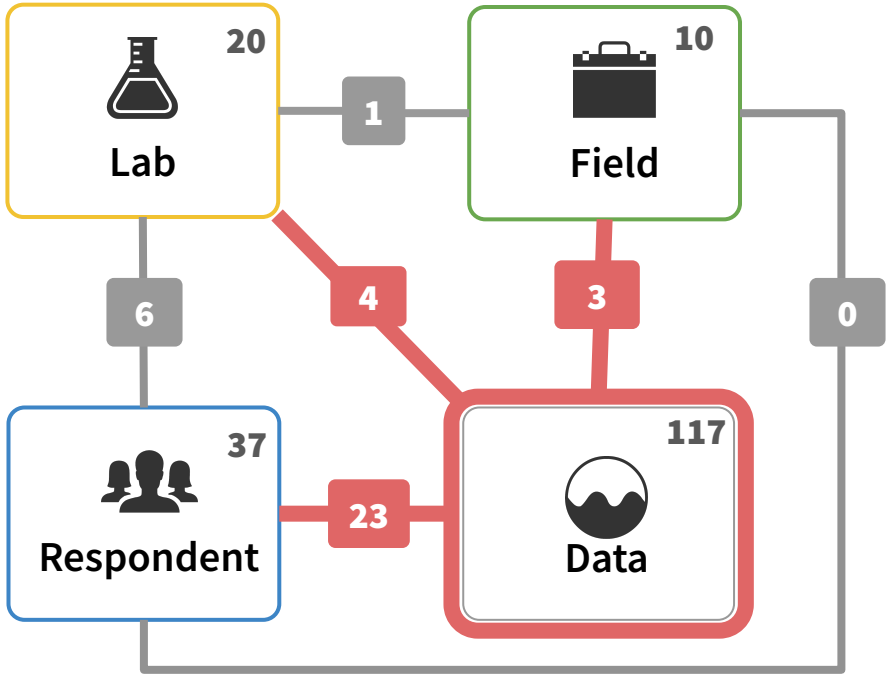


"AI does not fail people in a lab; it fails them in real life, with real consequences."

– Sloane and Moss, Nature 2019

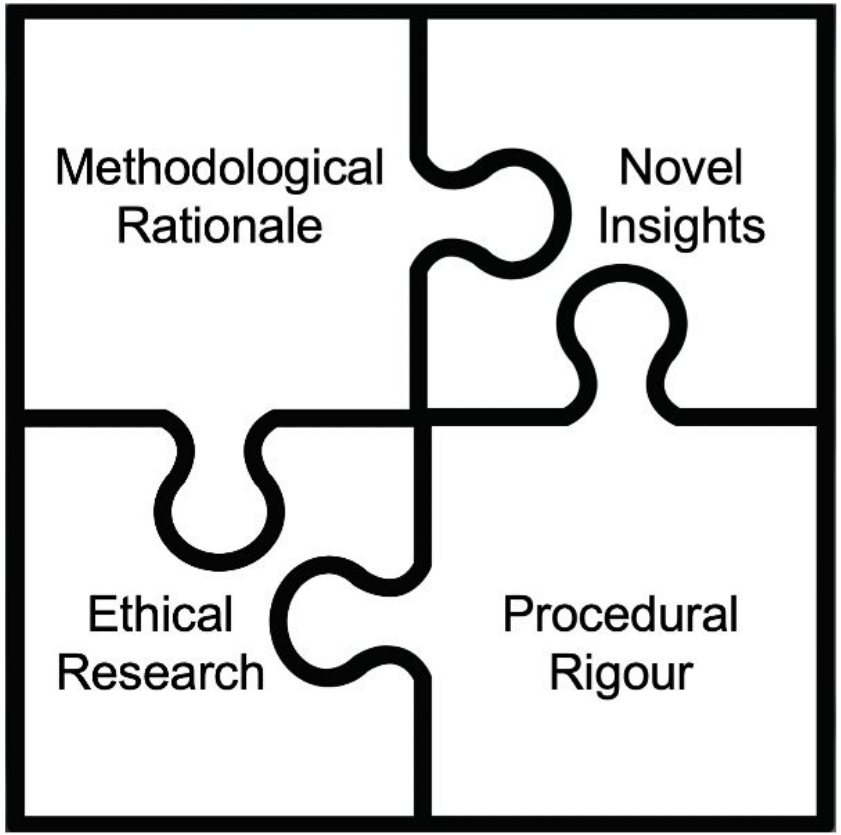


Use Mixed Strategies!



31 **Mixed Method (Triangulation)** across 151 SE papers

How to use Mixed Methods?



Mixed Method **Guidelines**

(tutorial paper in progress)

Exploratory sequential

Explanatory sequential

Convergent parallel

Multi method design

Uninvited guest: A method makes an unexpected entrance later in the paper

Smoke and mirrors: When one approach offers a token contribution

Selling your soul: Employing an additional method to appeal to reviewer

Integration failure: Poor integration of findings from all methods used

Limitation shirker: Failure to discuss limitations from all methods used

Missing the mark: Misalignment with the research question/objective

Sample contamination: Same participants used across methods

Ignoring the writing on the wall: Failing to use findings from an earlier study when developing a follow-up instrument

Sigsoft Empirical Standards

A Research Framework *(in action)*

Who

What

How

Disruptive software development innovations

Who?

beneficiary

- Human (Stakeholders)
- Technical System
  - may include tools, frameworks, and platforms being used to support development.
- Researcher

What?

contribution

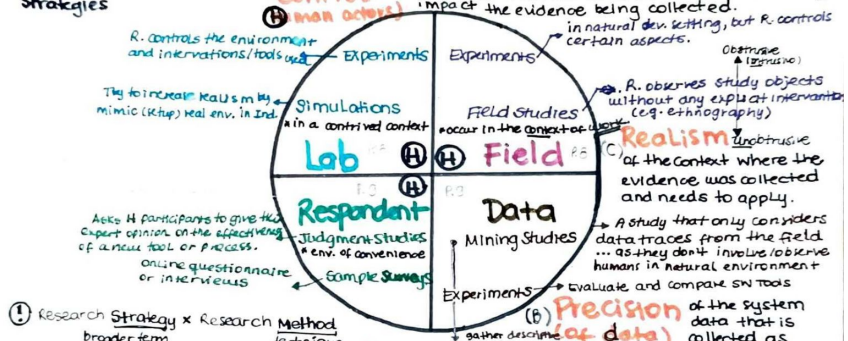
- **Descriptive**
  understand the nature of problems and real-world design contexts
- **Prescriptive/Solution**
  guide the design or improvement of solutions that address engineering prob

HOW?

Strategies

Empirical Strategies

Control of extraneous human behaviour variables that may impact the evidence being collected.



(!) Research Strategy × Research Method

broader term. technique used to collect data.

Ex. Interviews may be used as a method for collecting info for a Field Study or Sample Survey.

Generalizability (A) of the evidence over the population of human or system actors studied.

Universal ↔ particular

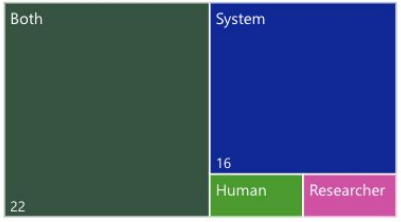
Non-Empirical Strategies

Formal Theory

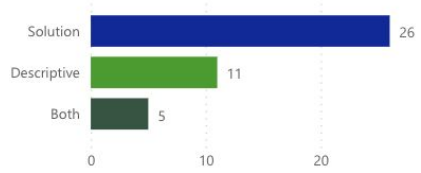
Meta

SANER 2021 Papers

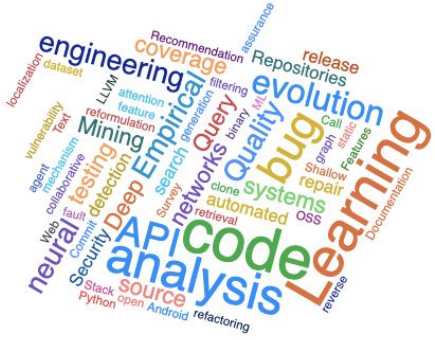
Who? (Beneficiary)



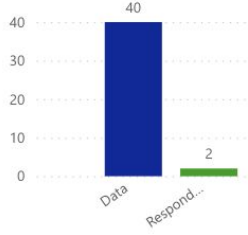
What? (Contribution)



Keywords (>=2 occurrences)



How? (Strategies)



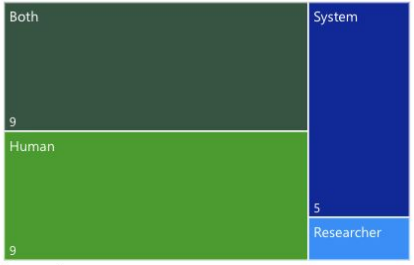
How? (Secondary strategies)

Search Abstract

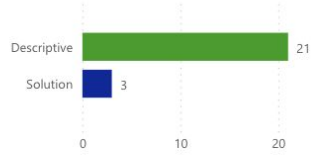
Search

ESEM 2021 Papers

Who? (Beneficiary)

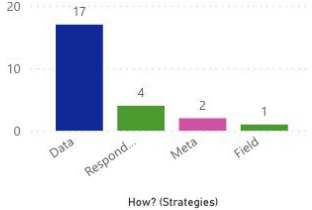


What? (Contribution)



Keywords

How? (Strategies)



How? (Secondary strategies)

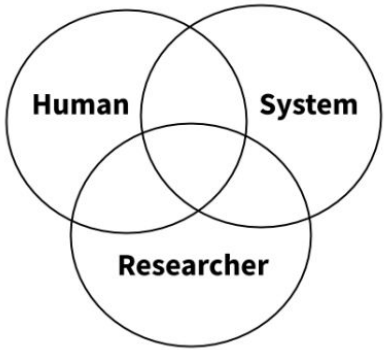
Search Abstract

Search

Who, What, How Framework applied to SANER 2021 and ESEM 2021

https://bit.ly/esem2021papers

Who?
(is the main beneficiary)

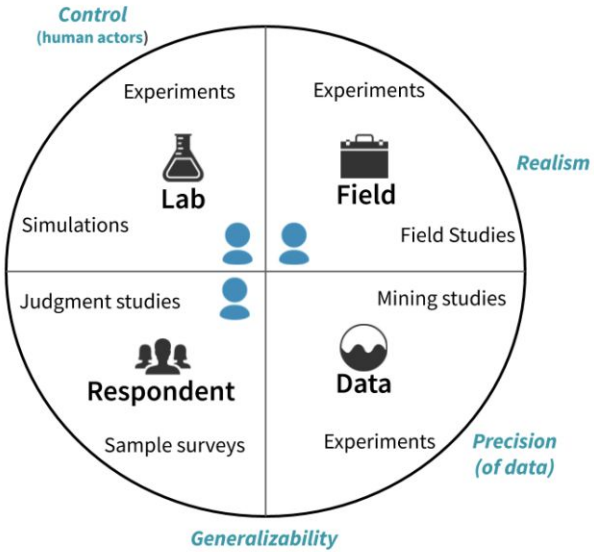


What?
(type of research contribution)

Descriptive **Solution**

How?
(which research strategies are used)

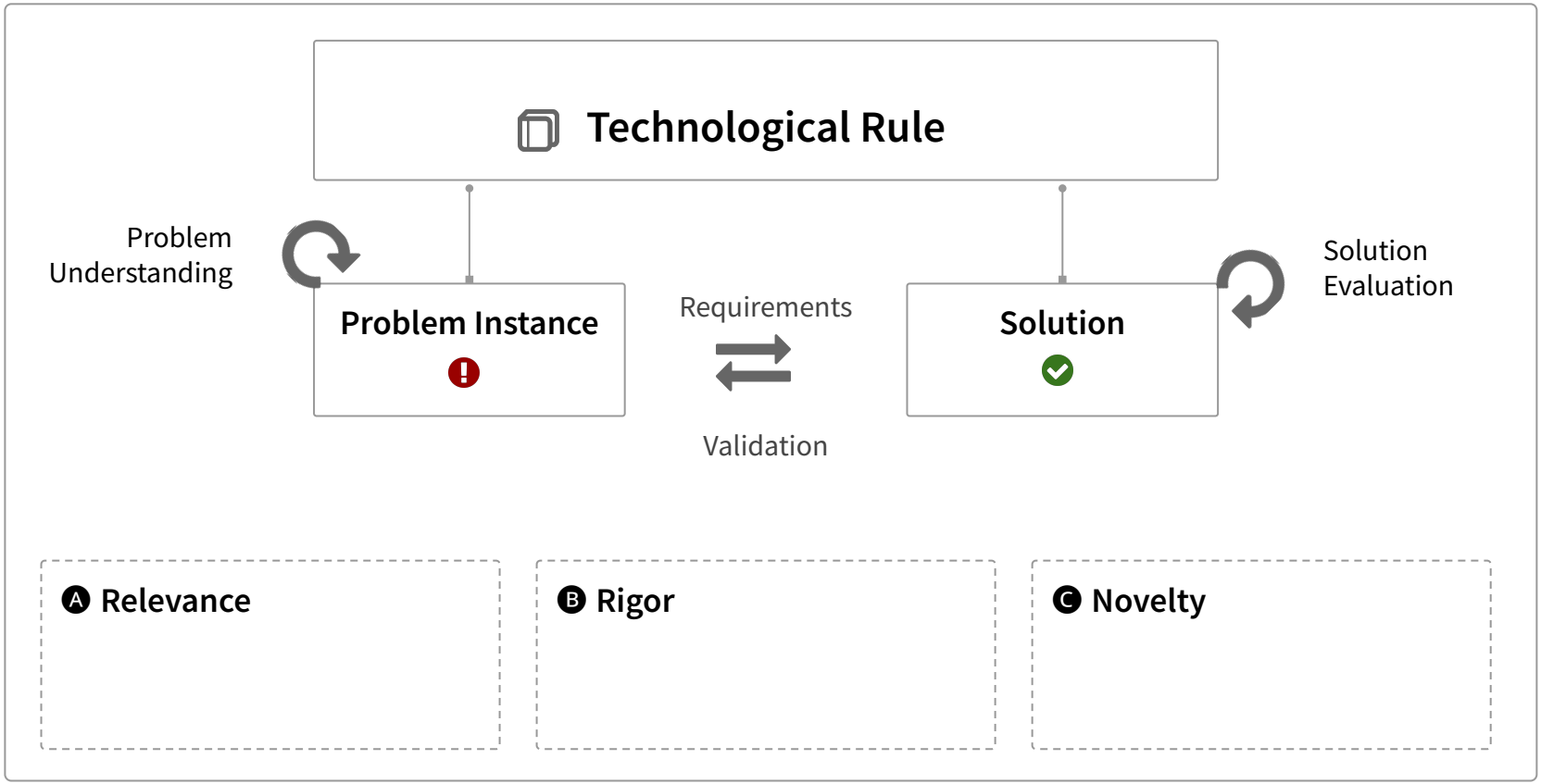
Empirical Strategies



Non-Empirical Strategies

Formal Theory **Meta**

Applying the Who, What, How Framework to your research



Design Science Template

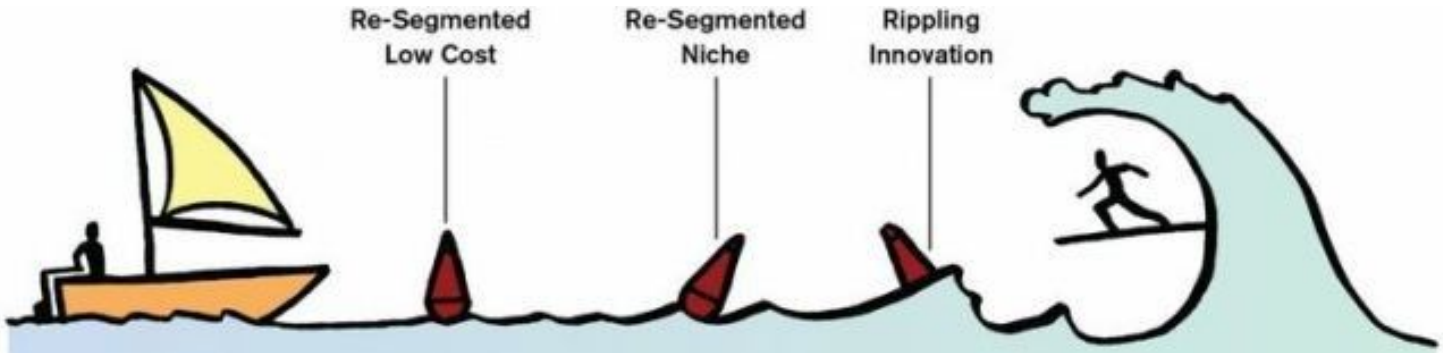
[Template link]

A Research Framework

- Who
- What
- How

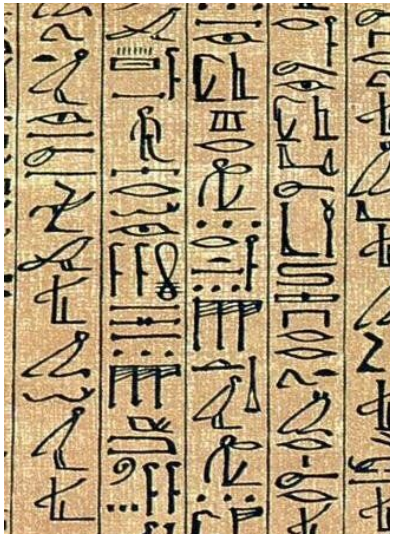
Disruptive software development innovations

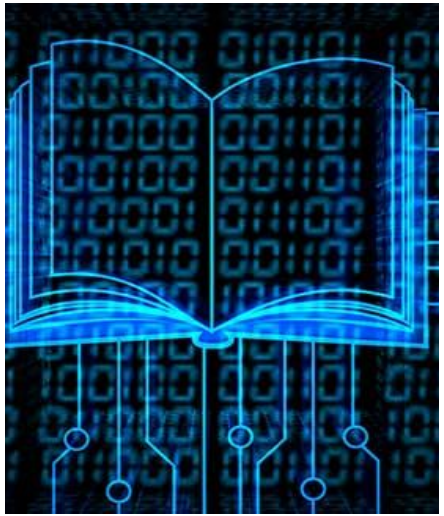
Disruptive innovations?



Sustaining Innovation	Disruptive Innovation
Problem is well understood	Problem not well understood
Existing Market	New Market
Innovation improves performance, lower cost, incremental changes	Innovation is dramatic and game changing
Customer is believable	Customer doesn't know
Market is predictable	Market is unpredictable
Traditional business methods are sufficient	Traditional business methods fail

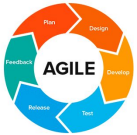
Automation disruptions

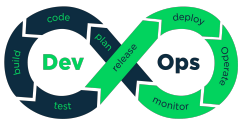


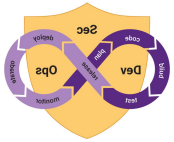


Disruptive innovations that **augment** human capabilities

Disruptive innovations in SE?







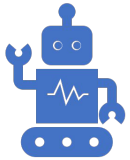
Refactoring (Johnson)

GitHub Copilot Service

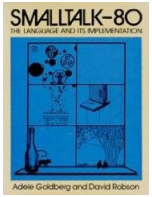
Continuous Integration (Booch)

GNU Make

Jenkins



Test Automation (IBM)



1968 1970 1980 1990 2000 2010 2020

Disruptive innovations in software engineering industry

Where do innovations come from?

Incremental innovations (e.g., automated testing techniques, bug prediction)

Breakthrough innovations (e.g., automated testing, CoPilot and use of NLP)

Experiential innovations (inspired by observing people or feeling pain)

Three types of **innovations**…

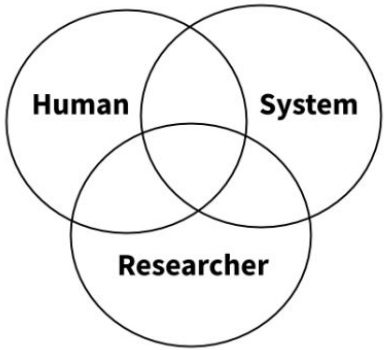
Feldman was inspired to write Make by the experience of a coworker in futilely debugging a program of his where the executable was accidentally not being updated with changes:

> Make originated with a visit from Steve Johnson (author of yacc, etc.), storming into my office, cursing the Fates that had caused him to **waste a morning debugging** a correct program (bug had been fixed, file hadn't been compiled, `cc *.o` was therefore unaffected). As I had spent a part of the previous evening coping with the same disaster on a project I was working on, **the idea of a tool to solve it came up**. It began with an elaborate idea of a **dependency analyzer, boiled down to something much simpler,** and turned into Make that weekend. Use of tools that were still wet was part of the culture. Makefiles were text files, not magically encoded binaries, because that was the **Unix ethos: printable, debuggable, understandable stuff.**
>
> —Stuart Feldman, *The Art of Unix Programming*, Eric S. Raymond 2003

Historical **glimpse into build automation**

How to study innovations in SE?

Who?
(is the main beneficiary)

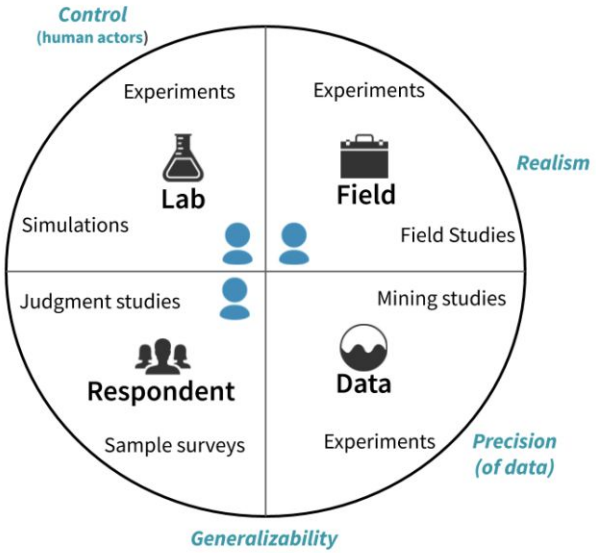


What?
(type of research contribution)

Descriptive **Solution**

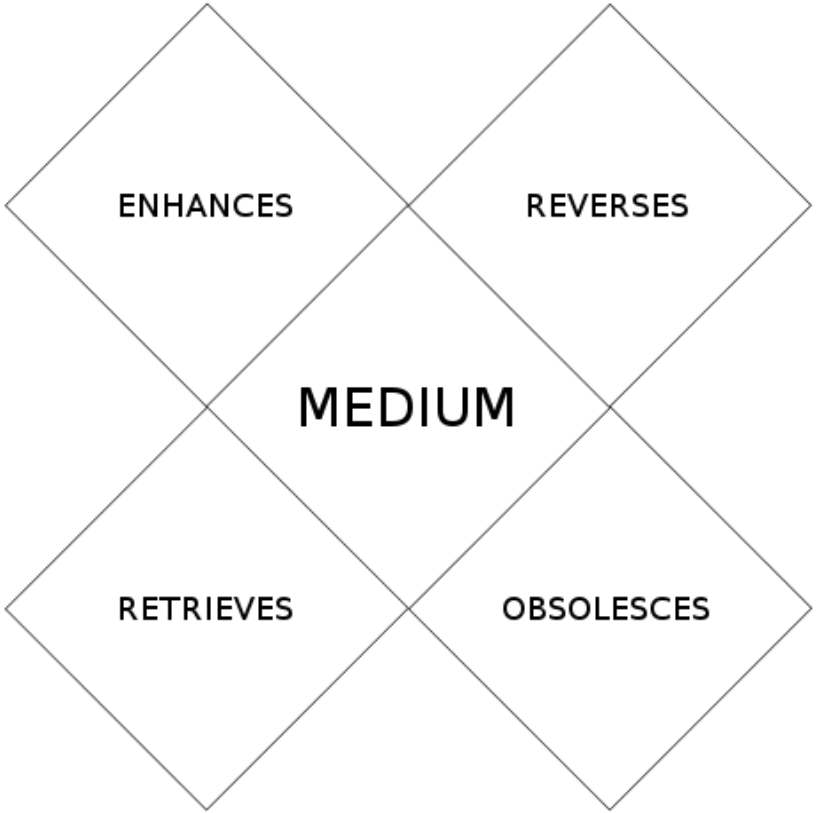
How?
(which research strategies are used)

Empirical Strategies



Non-Empirical Strategies

Formal Theory **Meta**

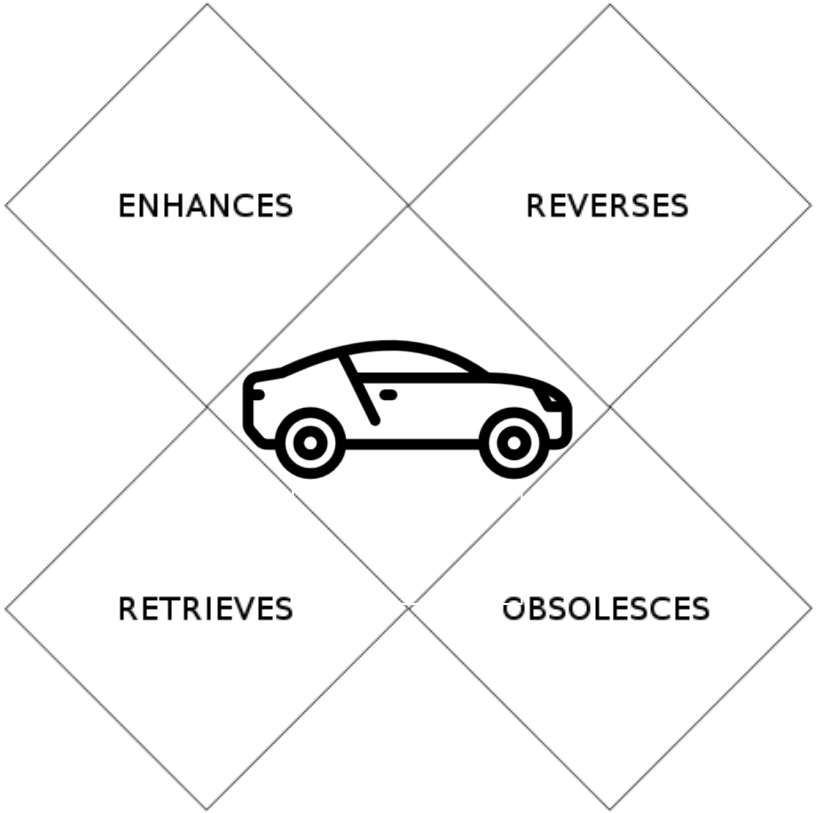


McLuhan – innovative disruptions that extend humans

See https://www.owenkelly.net/984/mcluhans-tetrads/

Travel speed

Traffic jams

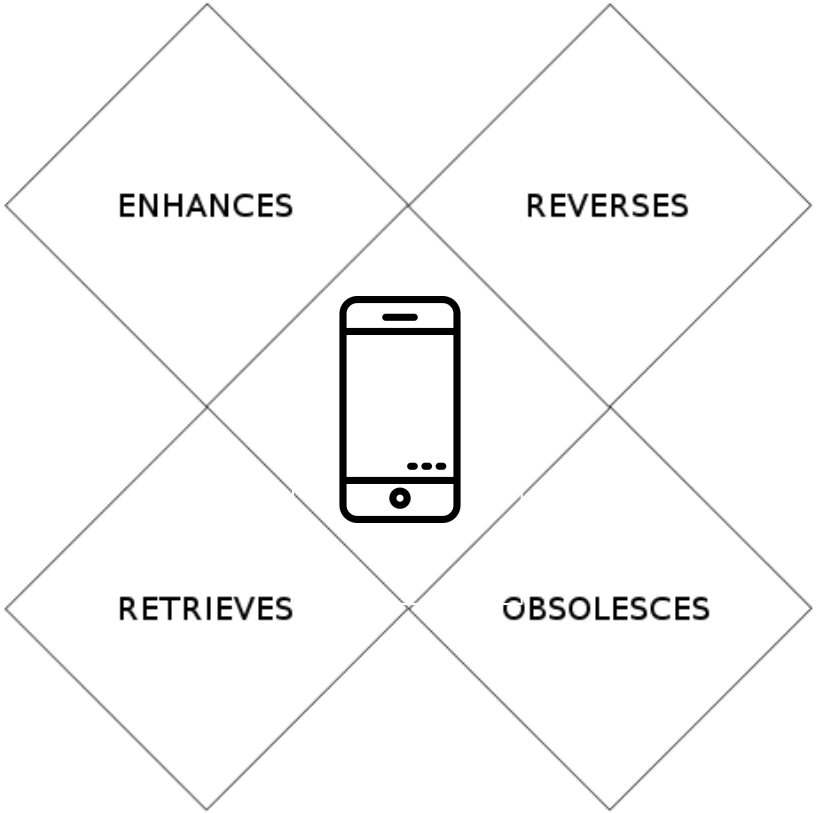


Knights in shining armour

Horse and buggy

McLuhan's **tetrad** example

Access to resources, Memory, Navigation



Addiction, Navigation skills (Flips into: watch)

Cameras, Headphones

Paper calendars, Maps

McLuhan's **tetrad** example

The McLuhan Institute
@McLinstitute

“The toughest [part of the #tetrad] is of course ‘retrieval:’ it takes serious learning and a deep knowledge of the subject. What’s been off-stage for a long time that serves as the etymology of this new thing, as it’s formal structuring principle?”

Eric McLuhan
letter, ‘90s

Programming knowledge (speed, quantity), Community

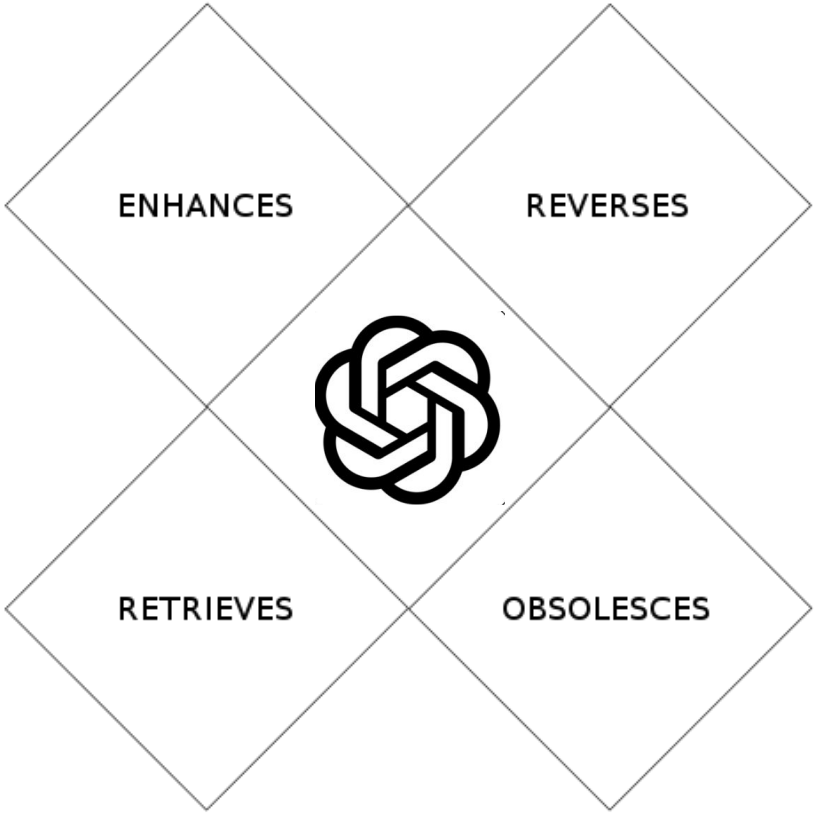
Lacking trust

Flips into: LLMs (need for customization)

Gurus, Portfolios

Email, Documentation, Onboarding

Programming speed,
Synthesized insights



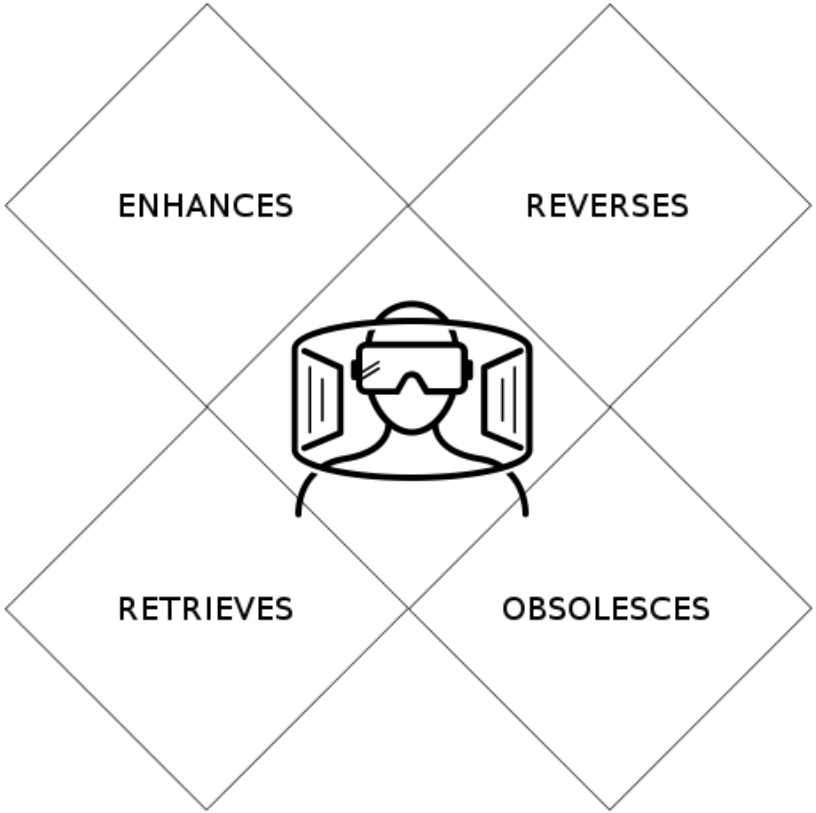
Code free development,
Echo chambers

Pseudocode,
Chatbots

Stack Overflow,
Education,
Creativity

McLuhan applied to **LLMs in SE**

Collaboration, Comprehension



Lost sense of reality and time Overload

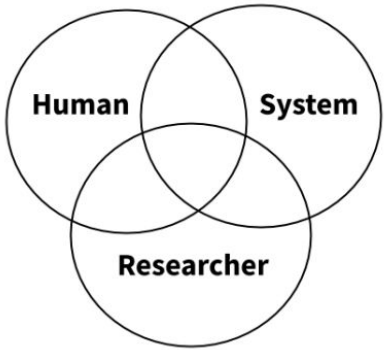
Flips into?

Face to face interactions at the whiteboard, Visualizations

Team rooms, Slack? 2D Vis

McLuhan applied to **AR/VR in SE**

Who?
(is the main beneficiary)

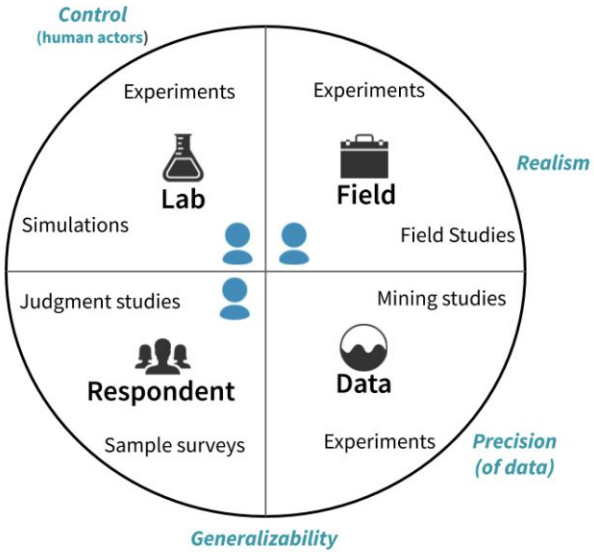


What?
(type of research contribution)

Descriptive **Solution**

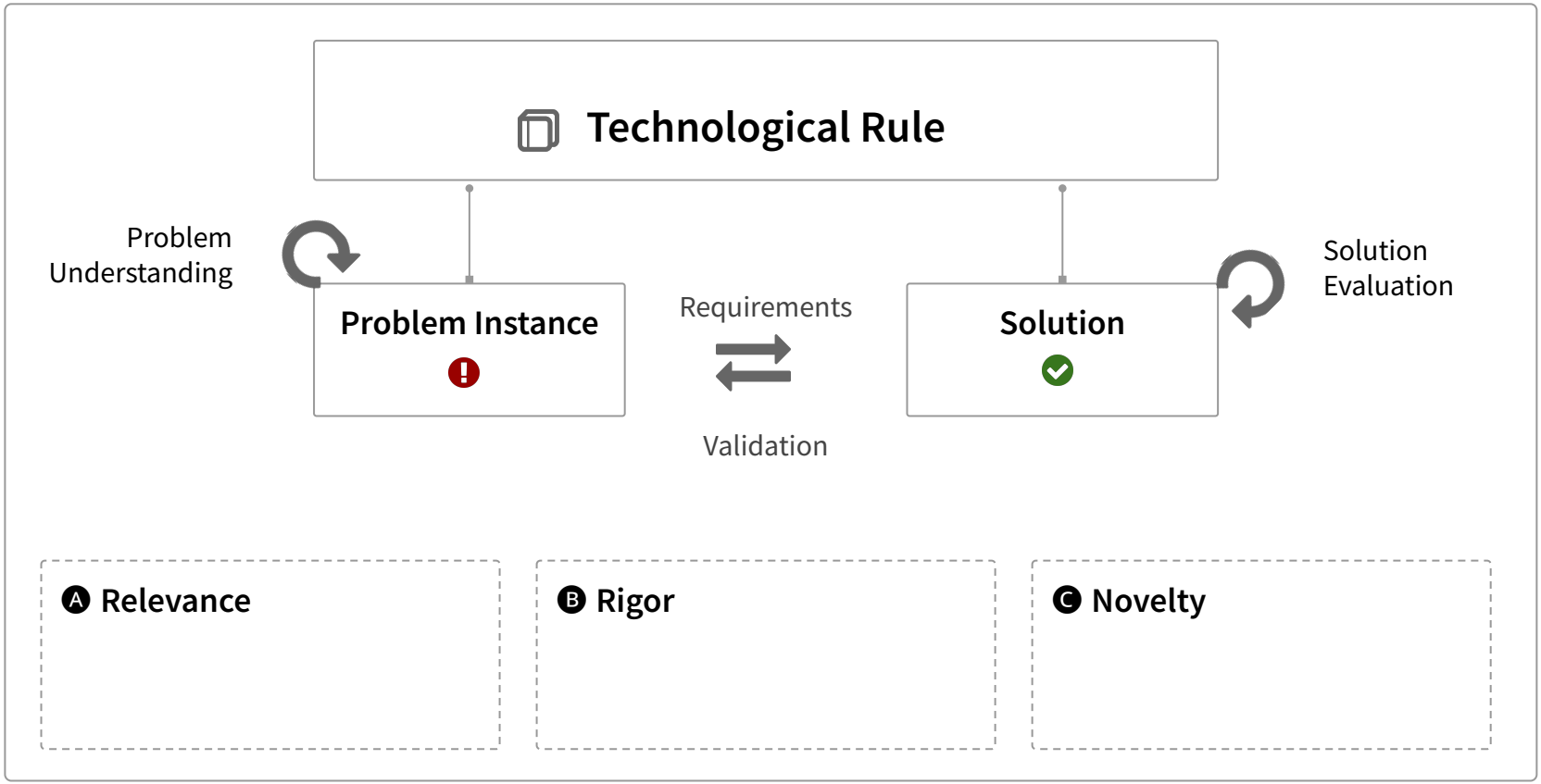
How?
(which research strategies are used)

Empirical Strategies



Non-Empirical Strategies

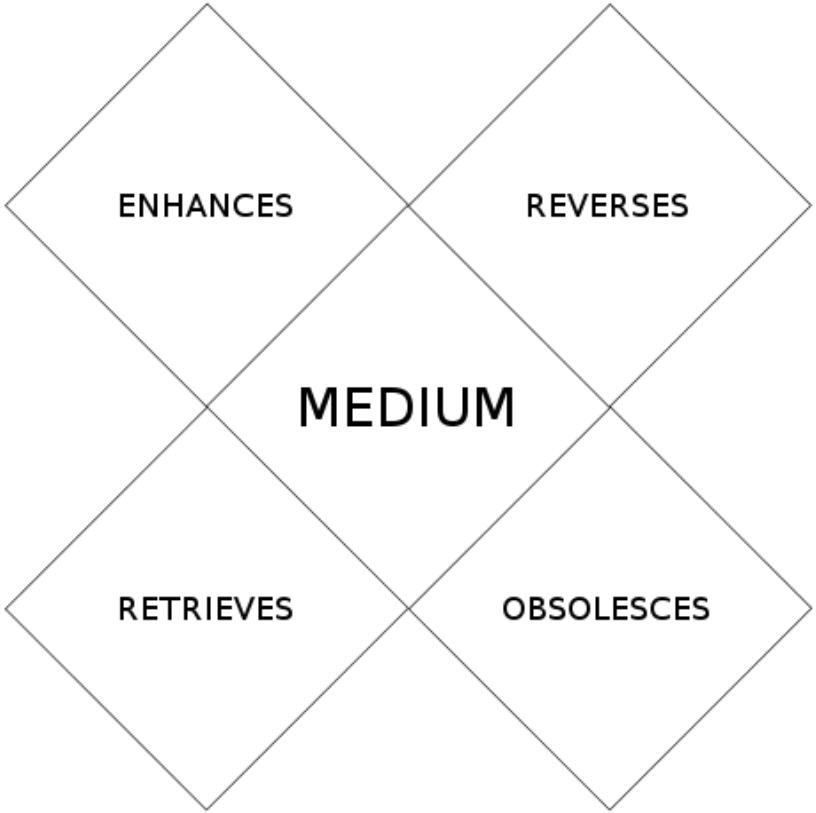
Formal Theory **Meta**



Design Science Template

[Template link]

“We shape our tools and thereafter our tools shape us”



 | **McLuhan** – innovative disruptions that extend humans

See https://www.owenkelly.net/984/mcluhans-tetrads/

Don’t forget about the **humans** we aim to support

Consider **McLuhan’s tetrad** when studying disruptions in SE

Use **mixed methods** and **innovative methods**
(disruptions change us)

Look for and **innovate future disruptions**

How will **AI rewire** the programmer's brain? the collective brain?

How will **AR/VR** change the face of collaboration?

What are the **next disruptions**? What will they make obsolete?

What does it mean to be **human**? Do we know? How to train developers to use new disruptive innovations?

Developments in software development outstrip our ability to understand the effects… should we **slow down**?

Storey, MA., Ernst, N.A., Williams, C. et al. The who, what, how of software engineering research: a socio-technical framework. Empir Software Eng 25, 4097–4129 (2020). [PDF]

“Using a visual abstract as a lens for communicating and promoting design science research in software engineering”, Storey, Engström, Höst, Runeson, Bjarnason, ESEM 2017. [PDF]

Engström, E., Storey, MA., Runeson, P. et al. How software engineering research aligns with design science: a review. Empir Software Eng 25, 2630–2660 (2020). [PDF]

S Easterbrook, J Singer, MA Storey, D Damian. Selecting empirical methods for software engineering research. 2008. [PDF]

Mixed Methods in SE: A Tutorial (coming soon, email me if you want an early version that will be ready in a few weeks!)

References

Joseph. E. McGrath. Methodology matters: Doing research in the behavioral and social sciences. 1972 [PDF]