SW Engineering Research Methods: A Guide for the Perplexed

Prof. Dr. Lutz Prechelt

2023-07

27 slides

Overview

1. 3 modes of SE research work
   - Theory, Construction, Empiricism
2. Quality criteria for empirical work
   - Credibility, Relevance
3. Research method archetypes
   - 3 dimensions → 4 common combinations
4. Some helpful method templates
   - Tool benchmarking, tool field trial, interviews+survey, process investigation
5. Some common mistakes
   - confusing engineering with science
   - making unwarranted assumptions (generalization, cost/benefit, meaning of measurements, human behavior)

1. SE research modes and output types: Theory, Construction, Empiricism (T, C, E)

- Theory (T):
  - Devising conceptual frameworks (definitions etc.) or theorems.
- Construction (C):
  - Building technical artifacts (e.g. software development tools).
- Empiricism (E):
  - Determining properties of artifacts or of the world.
- At any one time, you work in only one of these modes.

In the following, we focus on methods for Empiricism

- stand-alone empiricism or tool-related empiricism

Overview

Freie Universität Berlin

1. 3 modes of SE research work
   - Theory, Construction, Empiricism
2. **Quality criteria for empirical work**
   - **Credibility, Relevance**
3. Method archetypes
   - 3 dimensions → 4 common combinations
4. Some helpful method templates
   - Tool benchmarking, tool field trial, interviews+survey, process investigation
5. Some common mistakes
   - confusing engineering with science
   - making unwarranted assumptions (generalization, cost/benefit, meaning of measurements, human behavior)

Credibility (C)

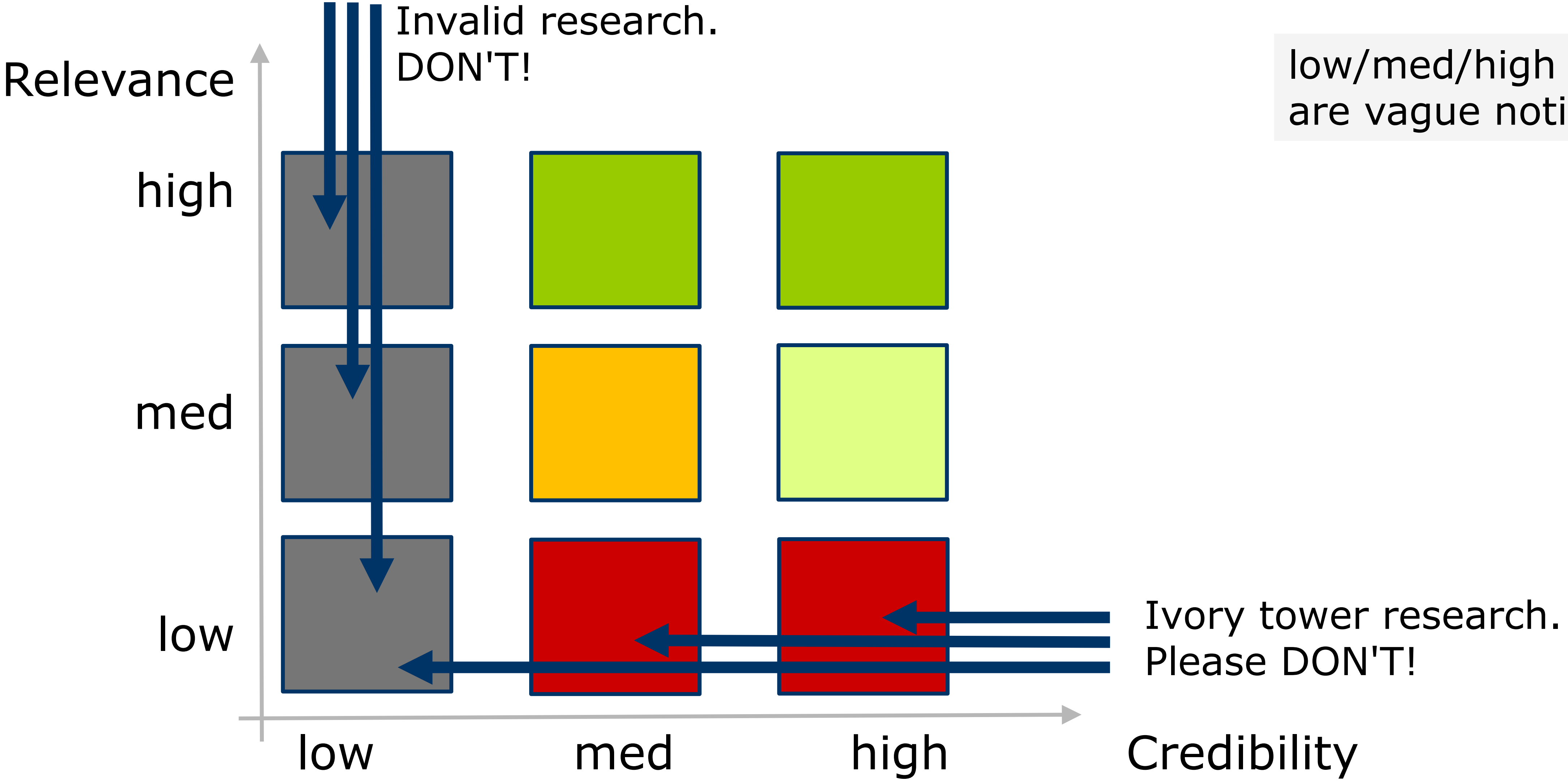
"How much do I trust these conclusions?"

Relevance (R)

"How valuable is it to know these conclusions?"

depends on the question, answer,
and applicability to my case

Insist on sufficient credibility and relevance!



low/med/high
are vague notions

Overview

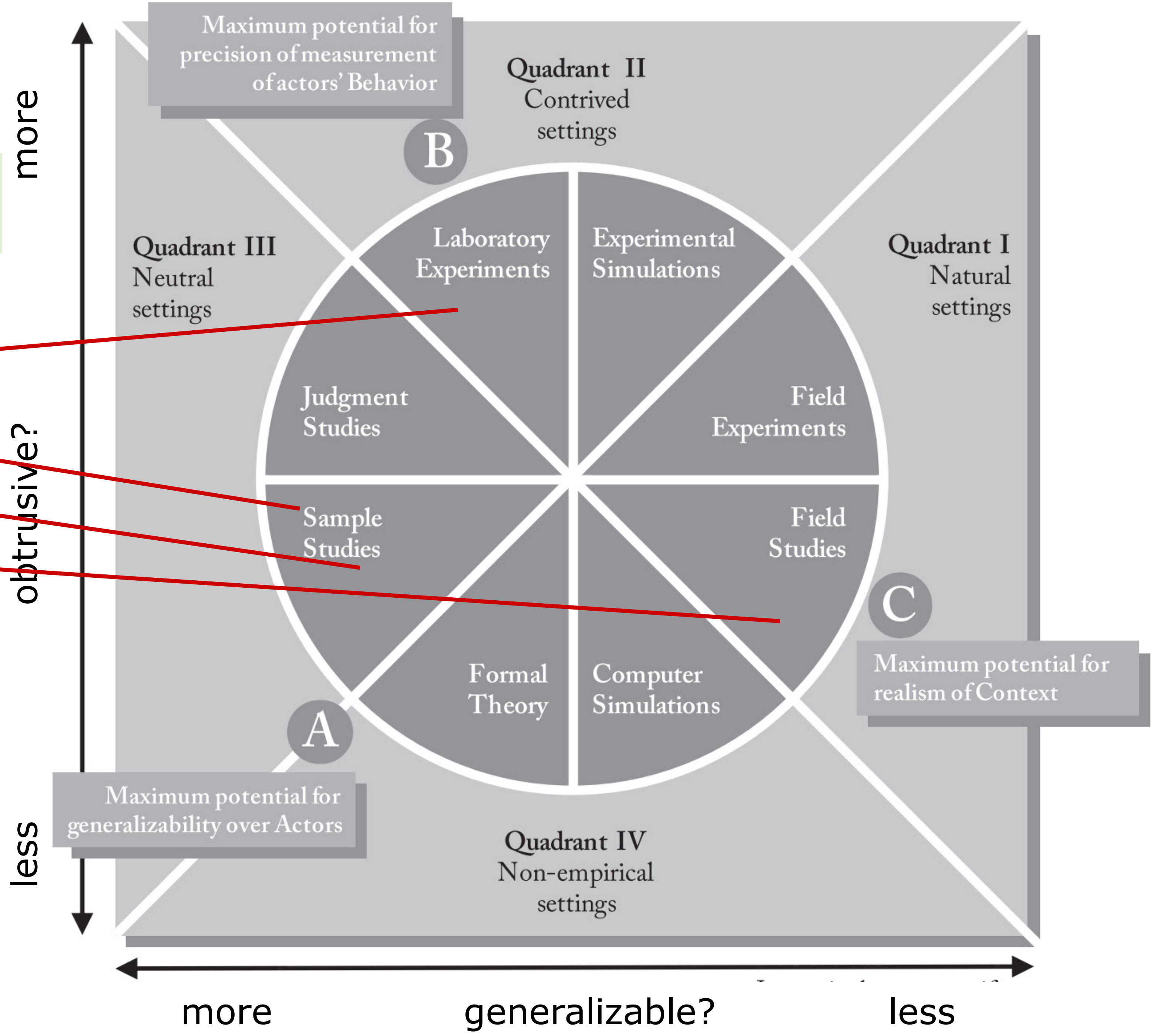
Freie Universität Berlin

1. 3 modes of SE research work
   - Theory, Construction, Empiricism
2. Quality criteria for empirical work
   - Credibility, Relevance
3. **Method archetypes**
   - **3 dimensions → 4 common combinations**
4. Some helpful method templates
   - Tool benchmarking, tool field trial, interviews+survey, process investigation
5. Some common mistakes
   - confusing engineering with science
   - making unwarranted assumptions (generalization, cost/benefit, meaning of measurements, human behavior)

Stol's method archetypes

Stol & Fitzgerald: "[The ABC of SW Eng. Research]", TOSEM 2018

1. controlled experiment
2. questionnaire survey
3. MSR correlational study
4. case study



We will use a *different* structure for forming archetypes:

Our method archetypes

Methods space is spanned by

- Research question nature:
  Howmuch? | Why? How?
- Situation wrt. repeatability:
  Humans | Machines
- Observations wrt. complexity:
  Numbers | Concepts

But not all 8 combinations occur:

4 Method archetypes:

- Quantitative [Numbers]
  - Experiments with groups of humans
    [Howmuch+Why, Humans, Nums]
  - Repeatable experiments
    [Howmuch+Why, Machines, Nums]
  - Fact-finding and correlation studies
    [Howmuch, X, Numbers]
- Qualitative [Concepts]
  - Sensemaking
    [Why/How, Humans, Concepts]

Overview

1. 3 modes of SE research work
   - Theory, Construction, Empiricism
2. Quality criteria for empirical work
   - Credibility, Relevance
3. Method archetypes
   - 3 dimensions → 4 common combinations
4. **Some helpful method templates**
   - **Tool benchmarking , tool field trial, interviews+survey , process investigation**
5. Some common mistakes
   - confusing engineering with science
   - making unwarranted assumptions (generalization, cost/benefit, meaning of measurements, human behavior)

Some useful method templates to take home

Study type "Automated tool benchmarking"

- When:
  - Validate effectiveness of an automated (analysis) tool [Howmuch?]
- What:
  - Collect a suitable corpus of objects; run tool;
    carefully judge each outcome [Machines, Numbers]
- Strengths:
  - Can use broad sets of inputs → Good generalizability
  - Easy to understand for readers
- Beware of:
  - Not discussing limits of applicability
  - Misjudging your own judgment
  - Being optimistic about users' judgment skills

Study type "Holistic field trial of tool"

Freie Universität Berlin

- When:
  - Validate *actual* usefulness and usability of a tool [How? Howmuch?]
- What:
  - Convince a team to use tool; study their work before and after introduction; analyze effort, benefits, difficulties [Humans, Concepts, Machines, Numbers]

- Strengths:
  - Insights with lots of structure and detail
  - Realistic, hence convincing
- Beware of:
  - Too-idiosyncratic settings → lack of generalization
  - Jumping to conclusions
  - Difficult and lots of effort!

Study type "Interviews + Survey"

Freie Universität Berlin

- When:
  - Measure attitudes and subjective appraisals regarding topic X [Howmuch?]
- What:
  - Interviews to find the relevant aspects of topic area [Humans, Concepts]; representative survey to measure distribution [Humans, Numbers]
- Strengths:
  - Can determine adequate questions and paint a realistic picture
  - Allows correlational analysis
- Beware of:
  - Self-selection bias
  - Ambiguous formulations
  - Respondent biases
  - Interpreting opinions as true statements of facts

Study type "Open process investigation"

- When:
  - To understand a relevant SW development process phenomenon [Why? How?]
- What:
  - Collect diverse types of data in the field (not only interviews!); perform sensemaking [Humans, Concepts]
- Strengths:
  - Statements grounded in specific instances → strong credibility
  - Captures phenomena that exist → strong generality
  - Provides better mental models for research and practice → strong relevance
- Beware of:
  - Jumping to conclusions
  - Risky: Takes looong, but it's unclear how interesting the results will be

Other

- Correlational studies of other sorts can be helpful as well [Howmuch?]
  - Mining software repositories
  - Special-purpose process metrics

- Meta-Scientific studies can be helpful as well [Why? How?]
  - Systematic Literature Reviews [X, Concepts/Numbers]
  - Credibility criticism studies [Concepts]
  - Relevance criticism studies [Concepts]

- And certainly more I have overlooked today.

Overview

1. 3 modes of SE research work
   - Theory, Construction, Empiricism
2. Quality criteria for empirical work
   - Credibility, Relevance
3. Method archetypes
   - 3 dimensions → 4 common combinations
4. Some helpful method templates
   - Tool benchmarking, tool field trial, interviews+survey, process investigation
5. **Some common mistakes**
   - **confusing engineering with science**
   - **making unwarranted assumptions (generalization, cost/benefit, meaning of measurements, human behavior)**

How to ruin your study

(some common mistake templates)

Popular-mistake type

PMT 1: Confusing engineering with science

Frederick Brooks: "The Computer Scientist as Toolsmith II", CACM 1996

The scientist *builds in order to study;*
the engineer *studies in order to build.*

- Science is about knowledge
- Engineering is about usefulness
    - Cf. the IEEE's mission statement:
      "IEEE's core purpose is to foster technological innovation and excellence for the benefit of humanity."

Therefore:

- **Articles that do not explain how their contribution might be useful are (presumably) not Software Engineering.**

Less dangerous for tool builders

PMT 2: Conclusions fuck-up

Broken tradeoff between credibility and relevance. Example:

- Facts:
  - 42 student subjects from University U; 2 pairs of toy programs of ~300 LOC; compare program variants with/without design pattern;
    measure time to finish an extension task correctly.
    Finished 16% faster ($p = 0.03$) with (vs. without) Observer pattern.
    Finished 29% faster ($p = 0.005$) with (vs. without) Decorator pattern
- Acceptable conclusion:
  - For subjects with similar background as ours, using the Observer or Decorator patterns can help finish program extension tasks faster – at least for small and clean programs.
- Botched conclusion:
  - Programs using design patterns are 16% to 29% faster to maintain than equivalent programs that do not use design patterns.

PMT 3: Assuming ROI silently

Pointing out benefits while ignoring the cost to get them.

- Example:
  - A tool analyzes source code to point out various classes of potential defects. Precision is shown to be 50%

  Typical assumptions:
  - Each of these defects is worth analyzing and understanding
  - The effort for recognizing the false positives to be false is not a problem

  - (Automated repair has an even more complex cost/benefit situation.)

PMT 4: Assuming ideal behavior

Assuming developers will do the Right Thing™ right away,
ignoring what happens otherwise.

- Example (continued):
  - A tool analyzes source code to point out various classes of potential defects. Precision is shown to be 50%

  Additional typical assumption:

  - User will not break correct code by "fixing" a defect that is in fact no defect.

PMT 5: Optimistic interpretation of measurements

Applying the most favorable interpretation of some measurement, ignoring several alternative interpretations.

in particular: seeing a specific causation in a correlation

- Example finding: 100 Java Projects exhibit a much lower fraction of methods with the "long method" code smell than 100 Python projects.
  - Conclusion: Java developers care more about their code
  - BUT perhaps it's just the many getters/setters that don't exist in Python?
- Example finding: Ditto, but Java has *higher* fraction than Python
  - Conclusion: Python developers care more about their code
  - BUT does the smell really indicate a problem or is it often just a matter of taste?
  - BUT is binary classification of smell vs no smell appropriate?
  - BUT Java is more verbose. Is the same threshold appropriate in both languages?

Good studies must be handcrafted.

Standardized recipes are rarely adequate.

Summary

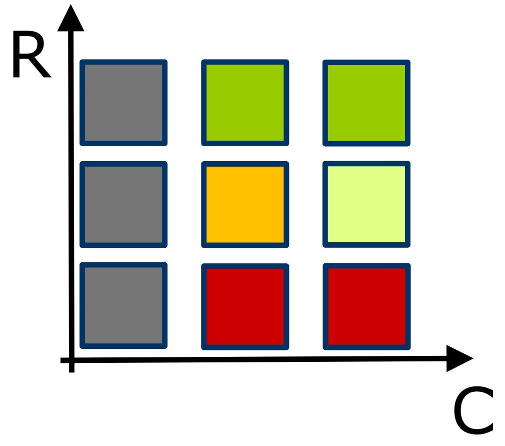
Freie Universität Berlin

1. Empirical work strives for sufficient Credibility and high Relevance
2. Methods are quantitative [Howmuch, X, Numbers]
   - e.g. benchmarking of automatic tools (in the laboratory)
3. or qualitative/sensemaking [How|Why, Human, Concepts]
   - e.g. case study of human-operated tools (in the field)
4. They can be varied endlessly and can be combined
   - e.g. Interviews(sensemaking) followed by Survey(correlational)
5. Watch out to avoid common types of mistake
   - e.g. not explaining usefulness
   - e.g. making unwarranted assumptions
     - regarding generalizability
     - regarding the cost/payoff situation
     - regarding the meaning of measurements

Thank you!

and now...

Discussion, please!



0. Questions, anybody?

1. Did you have an aha-moment? Which?

2. Do you have new ideas now wrt your emprical work?

Rational research progression (per strand of empirical SE research)

Given a broad research interest, e.g.

- How should we use X?
  - e.g. models or modeling or pair programming or ...
- How does X compare to Y?
  - e.g. maintainability of Java code versus Python code, or ...

- A sensible progression of research could be:
  - Understand relevant factors
    - identify, describe
  - Formulate a theory of their relationships (mechanisms)
    - talks about the development process
  - Validate the theory
  - Measure the size of certain effects in the theory
    - Quantification, based on the qualitative theory

Rational research progression (per strand of tool-building SE research)

Freie Universität Berlin

Given a broad research interest, e.g.

- How can we best solve X?
  - by any kind of tool support

- A sensible progression of research could be:
  - Understand relevant problems
    - identify, describe
  - Formulate a theory of their relationships (mechanisms)
    - talks about the development process
  - Validate the theory
  - Find one or more points of attack
    - where improvements will be most useful
  - Devise and build helpful tools

Premature tool-building is much like premature quantification