Design Science and Software Engineering

Roel Wieringa

University of Twente

The Netherlands

Outline

- Practical problems versus knowledge problems
  - Problem choice
- Design science and software engineering
  - Theories
  - Research methods

Information systems research problems (Department of management science)

- 1980s
  - Complaints about lack of empirical rigour
  - Papers about empirical methods for IS research
- 1990s
  - Empirical papers
- 2000s
  - Complaint about lack of relevance
  - "Relevance will improve if we include designing in our research".

Example papers at Int'l Conf. on Information Systems 1997:

"Successful IS innovation: the contingent contributions of innovation characteristics and implementation process"

"The effects of task interruption and information presentation on individual decision making"

"The impact of CASE on IS professionals' work and motivation to use CASE"

"The impact of information technology on coordination costs: implications for firm productivity"

....

Software engineering research problems (Department of computer science)

- 1990s
  - Complaints about lack of relevance of SE techniques delivered by Academia
  - Papers about how to do empirical validation of techniques
- 2000s
  - Increasing number of papers validate their solution
  - Complaints that solutions solve no relevant problems

Int'l Conf. on Software Engineering 2003:

"Improving web application testing with user session data"

"Constructing test suites for interaction testing"

"Improving test suites via operational abstraction"

"Recovering documentation-to-source-code traceability links using latent semantic indexing"

"Computer-assisted assume/guarantee reasoning with VeriSoft"

The problem of the problem

- “Unvalidated technology will not be used” (SE)
  - But validated solutions to irrelevant problems will not be used either
- “Applicable knowledge consists of solutions to design problems“ (IS)
  - But designs can be irrelevant too
- Reflection
  - Relevance is context-dependent
  - Relevance is time dependent
  - Relevance is fitness to solve a practical problem

- We should look at the kind of problems we want to solve!

Practical problems versus knowledge problems

Practical problems versus knowledge problems

- Practical problem
  - Difference between current state of the world and what a stakeholder would like it to be
    - To solve it, stakeholder must change the world
- Knowledge problem
  - Difference between what current stakeholder knows and what the stakeholder wants to know
    - To solve it, stakeholder needs to change their knowledge of the world

Knowledge question or practical problem?

- What are the goals of these users?
  – K. Empirical question
- What would be a good procurement process for Office supplies?
  – P. Design an improved procurement process
- What is the complexity of this algorithm?
  – K. Analytical question
- Why is this algorithm so complex?
  – K. Analytical question
- Find an algorithm to solve this problem
  – P. Design an algorithm to solve this problem
- How do users interact with this system?
  – K. Empirical question
- Why do users interact with the system this way?
  – K. Empirical question
- What would be a good architecture for hospital-insurance company communication?
  – P. Design an architecture

What kind of problem?

- *What is the architecture of the communication infrastructure between A and B?*
  - *K Problem: infrastructure exists, stakeholder does not know what its architecture is*

- *What is a communication infrastructure between …*
  - *P Problem: A blueprint must be made*

Misleading!

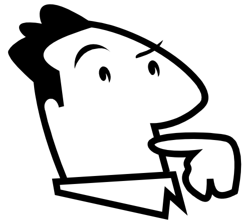
- *Design a communication infrastructure between …*
  - *P Problem: A blueprint must be made*

Heuristics

- Practical problems
  - Are solved by changing the state of the world
  - Solution criterion is utility
    - Problem-dependent: stakeholders and goals
    - Many solutions; but trade-offs
- Knowledge questions
  - Are solved by changing the knowledge of stakeholders.
  - Solution criterion is truth
    - Problem-independent: no stakeholders
    - One solution; but approximations

Doing
Changing the world
Future-oriented

Thinking
Changing our mind
Past-oriented



Science versus engineering

- Practical problems
  – Are solved by changing the state of the world
- Knowledge questions
  – Are solved by changing the knowledge of stakeholders.

Engineering = rational search for new or improved technology

Science = rational search for new or improved knowledge

Rational = Being able to justify your answers
•in terms of alternatives not chosen
•and in terms of goals to be achieved

Engineering cycle

- **Problem investigation:** What is the problem?
- **Solution design:** Specify a solution
- **Design validation:** Does it solve the problem?

Design cycle / RE

- **Selection**
- **Specification implementation**
- **Implementation evaluation:** Did it solve the problem?

•Engineering is a rational way to solve a practical problem
•Specification: Make a blueprint before acting
•Validation: Be critical about the blueprint, consider alternatives

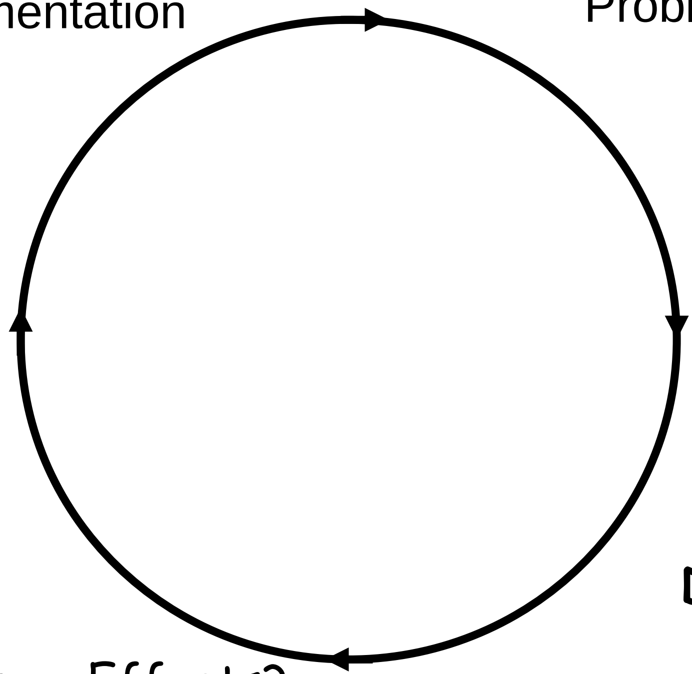
Engineering cycle

IS papers

K Implementation evaluation = Problem investigation

- K Stakeholders?
- K Their goals?
- K Problematic phenomena?
- K Their causes?
- K Impacts?
- K Solution criteria?

A Specification implementation



SE papers

D Solution design

- K Available solutions?
- D Design new ones

K Design validation

- K Context & Solution → Effects?
- K Effects satisfy goals?
- K Whose goals?
- K Trade-offs for different Solutions?
- K Sensitivity for different Contexts?

Information systems research problems (Department of management science)

Example papers at Int'l Conf. on Information Systems 1997:

"Successful IS innovation: the contingent contributions of innovation characteristics and implementation process"

"The effects of task interruption and information presentation on individual decision making"

"The impact of CASE on IS professionals' work and motivation to use CASE"

"The impact of information technology on coordination costs: implications for firm productivity"

....

Implementation evaluation

Software engineering research problems (Department of computer science)

Int'l Conf. on Software Engineering 2003:

"Improving web application testing with user session data"

"Constructing test suites for interaction testing"

"Improving test suites via operational abstraction"

"Recovering documentation-to-source-code traceability links using latent semantic indexing"

"Computer-assisted assume/guarantee reasoning with VeriSoft"

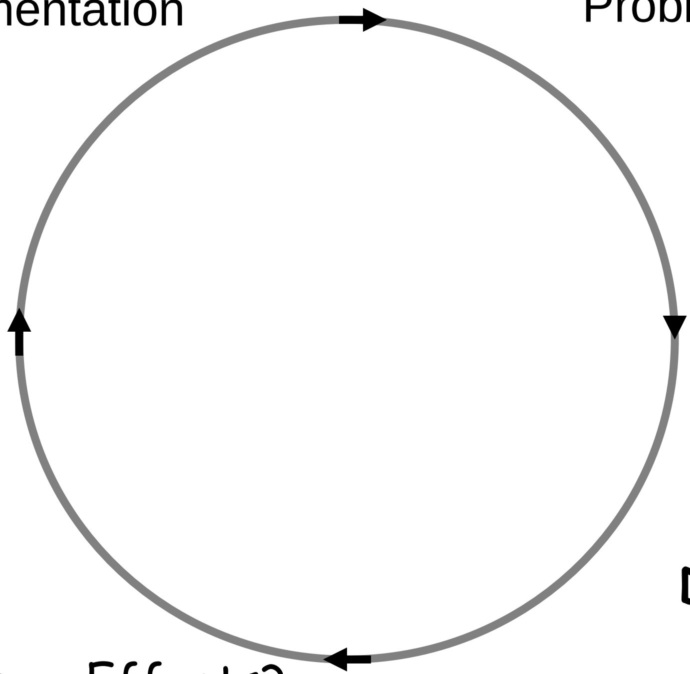
Solution design

Engineering research questions

K Implementation evaluation = Problem investigation

- K Stakeholders?
- K Their goals?
- K Problematic phenomena?
- K Their causes?
- K Impacts?
- K Solution criteria?

A Specification implementation



D Solution design

- K Available solutions?
- D Design new ones

K Design validation

- K Context & Solution → Effects?
- K Effects satisfy goals?
- K Whose goals?
- K Trade-offs for different Solutions?
- K Sensitivity for different Contexts?

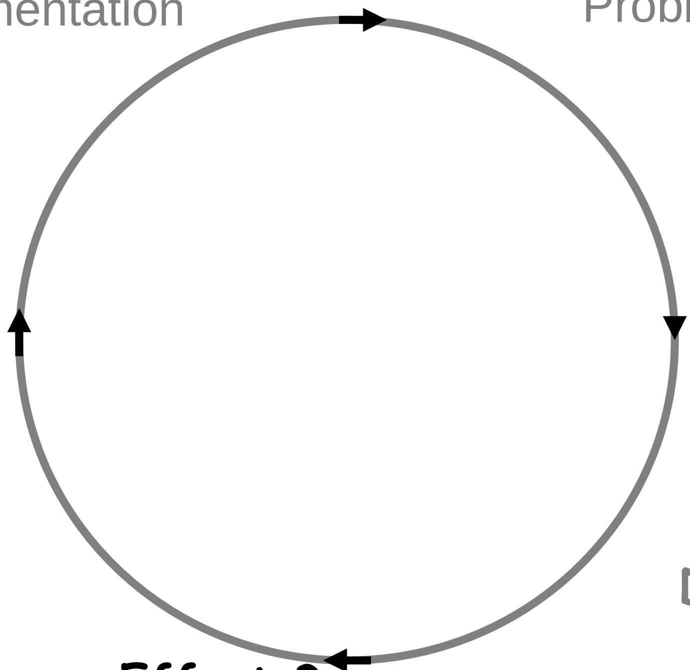
Engineering research questions

K Implementation evaluation = Problem investigation

- K Stakeholders?
- K Their goals?
- K Problematic phenomena?
- K Their causes?
- K Impacts?
- K Solution criteria?

A Specification implementation

Problem nesting

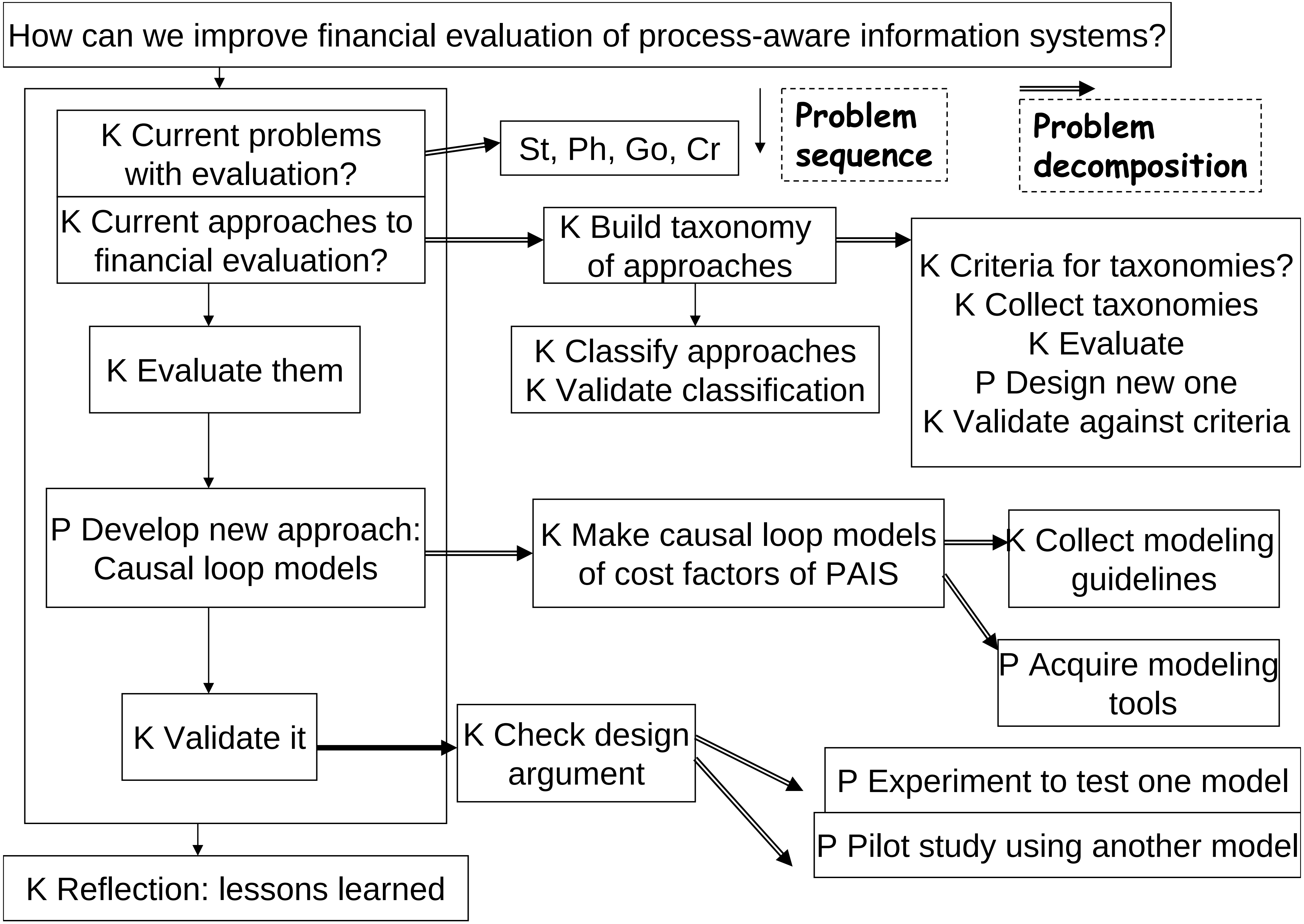


D Solution design

- K Available solutions?
- D Design new ones

K Design validation

- K Context & Solution → Effects?
- K Effects satisfy goals?
- K Whose goals?
- K Trade-offs for different Solutions?
- K Sensitivity for different Contexts?


How can we improve financial evaluation of process-aware information systems?
Problem sequence
Problem decomposition
K Current problems with evaluation?
St, Ph, Go, Cr
K Current approaches to financial evaluation?
K Build taxonomy of approaches
K Criteria for taxonomies?
K Collect taxonomies
K Evaluate
P Design new one
K Validate against criteria
K Classify approaches
K Validate classification
K Evaluate them
P Develop new approach: Causal loop models
K Make causal loop models of cost factors of PAIS
K Collect modeling guidelines
P Acquire modeling tools
K Validate it
K Check design argument
P Experiment to test one model
P Pilot study using another model
K Reflection: lessons learned

Bulleted list form

- Improving the financial evaluation of PAIS
- Problem investigation
  - Current problems with financial evaluation
    - Current approaches
      - Taxonomies of approaches
      - Our taxonomy
    - Evaluation of approaches
- Solution approach
  - Causal loop models
  - CLDs of cost factors
- Validation
  - The engineering argument
  - Experiment
  - Pilot study
- Discussion and lessons learned
- Appendices
  - Modeling guidelines for CL modeling
  - Tools for CL modeling

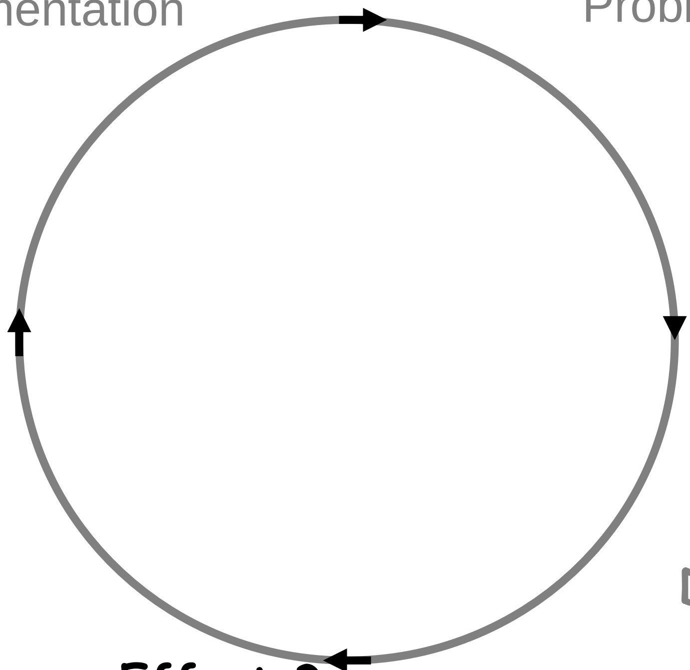
Very good PhD thesis outline

Engineering research questions

K Implementation evaluation = Problem investigation

- K Stakeholders?
- K Their goals?
- K Problematic phenomena?
- K Their causes?
- K Impacts?
- K Solution criteria?

A Specification implementation



D Solution design

- K Available solutions?
- D Design new ones

K Design validation

- K Context & Solution → Effects?
- K Effects satisfy goals?
- K Whose goals?
- K Trade-offs for different Solutions?
- K Sensitivity for different Contexts?

Engineering questions at ICSE02 (Mary Shaw reformulated by me)

Design

- …esign of X

Validation

- …een X…

- **Curiosity-driven engineering**
- **Aim for radical technology?**

Research questions at ICSE02 (Mary Shaw reformulated by me)

Descriptive research

What is X

Relation of X and Y

- **Explanatory research of the kind "Why does this happen?" is absent**

Design science and software engineering

Theories

Levels of knowledge

Abstraction ↑

N=∞ ……………………….Universal theories………………… Public — *Non-mission-oriented science*

N=k

Diagnostic theories about problem mechanisms

Treatment theories about solution mechanisms

Design science: *solving classes of practical problems*

Caricature: Every case is unique

Context

X ——→ Effects, should satisfy Goals

Solution

N=1

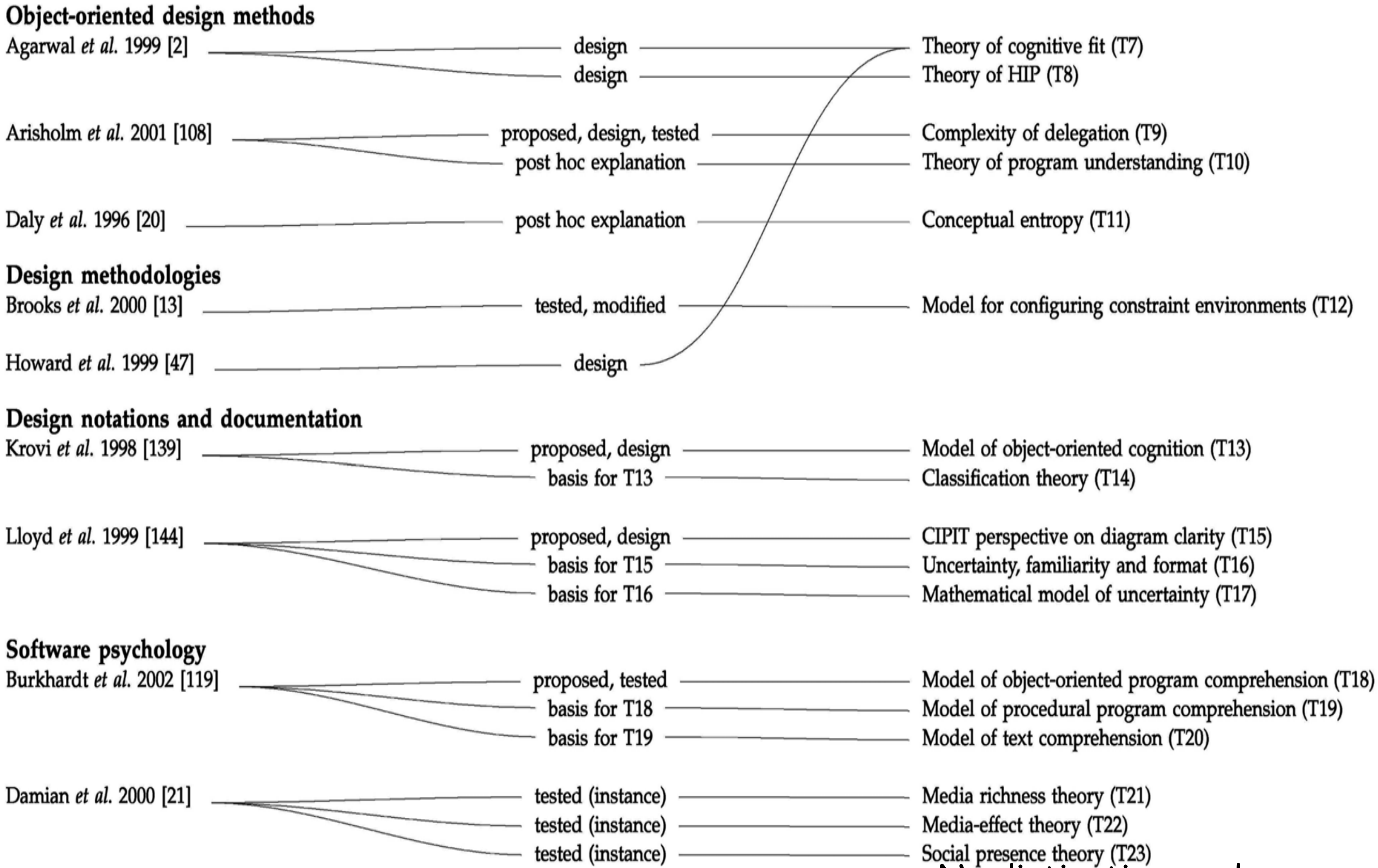
Conditions of practice

Engineering, consultancy: solving known problems

Confidential ↑ Public

Risk: other effects than predicted

Risk: "side" effects that do not satisfy goals



- J.E. Hannay et al. "A systematic review of theory use in software engineering experiments". *IEEE TSE,* 30(2), February 2007, pages 87-107.

No distinction made between universal theories and design theories

Frequently used IS theories

http://www.fsc.yorku.ca/york/istheory/wiki/index.php/Main_Page

1. Diffusion of innovations theory
   - Agarwal, R., & Prasad, J. (1997). The role of innovation characteristics and perceived voluntariness in the acceptance of information technologies. *Decision Sciences,* 28(3), 557-582.
   - Armstrong, J. S., & Yokum, J. T. (2001). Potential diffusion of expert systems in forecasting. *Technological Forecasting and Social Change,* 67(1), 93-103.
   - Baskerville, R L & Pries-Heje, J (2001). A multiple-theory analysis of a diffusion of information technology case. *Information Systems Journal,* 11(3), 181-212.
   - Beatty, R. C., Shim, J. P., & Jones, M. C. (2001). Factors influencing corporate web site adoption: A time-based assessment. *Information & Management,* 38(6), 337-354.
2. Technology acceptance model
3. Contingency theory
4. Organizational culture theory
5. Resource-based view of the firm

No distinction made between universal theories and design theories

Design science and software engineering

Do we need any special methods?

Validation methods in SE

Zelkowitz & Wallace 1998	**Description**	**This talk**
Project monitoring	Collection and storage of project data	Measuring instrument (primary sources)
Case study	Collection of project data with a research goal in mind	Research method
Assertion	The researcher has used the technique in an example, with the goal of showing that the technique is superior	Not a research method
Field study	Collection of data about several projects with a research goal in mind	Research method
Literature search		Metaresearch
Legacy data	Collection of project data after the project is finished	Measuring instrument (primary sources)

Methods mixed up with measuring instruments

Validation methods in SE

Zelkowitz & Wallace 1998	Description	This talk
Lessons learned	Study of documents produced by a project	Data analysis method (Conceptual analysis)
Static analysis	Studying a program and its documentation	Measuring instrument (Primary sources)
Replicated experiment	Several projects are staffed to perform a task in multiple ways	Research method (field experiment)
Synthetic environment experiment	Several projects are performed in an artificial environment	Research method (lab experiment)
Dynamic analysis	Instrumenting a software product to collect data	Measuring instrument (monitoring devices)
Simulation	Executing a product in an artificial environment	Research method (lab experiment)

Validation methods in design science

Scaling up	Cond. of pract.	Cntrl of cntxt	Unit of data c.	Example	User	Goals
Illustration	no	yes	model	small	designer	illustration
Opinion	imagined	yes	model	any	Stakeh.	support
Lab demo	no	yes	model	realistic	designer	knowledge
Lab expt.	no	yes	model !	artificial		
Benchmark	no	yes	model	standard		
Field trial	yes	yes	model	realistic		
Field experiment	yes	yes	model	realistic	Stakeh.	knowledge
Action cas	yes	no	model	real	designer	Knowledge and change
Pilot projec	yes	no	model	realistic	Stakeh.	knowledge
Case study	yes	no	model	real	Stakeh.	Knowledge and change

Standard methods, but need to scale up to conditions of practice

Discussion