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)

Int'l Conf. on Software Engineering 2003:

• 1990s
  – Complaints about lack of relevance of SE techniques delivered by Academia
  – Papers about how to do empirical validation of techniques
    “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”

• 2000s
  – Increasing number of papers validate their solution
  – Complaints that solutions solve no relevant problems
    “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

• 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?
- 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

A Specification implementation

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?

K Implementation evaluation = Problem investigation
  K Stakeholders?
  K Their goals?
  K Problematic phenomena?
  K Their causes?
  K Impacts?
  K Solution criteria?

D Solution design
  K Available solutions?
  D Design new ones
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

A Specification implementation

K Implementation evaluation = Problem investigation
  K Stakeholders?
  K Their goals?
  K Problematic phenomena?
  K Their causes?
  K Impacts?
  K Solution criteria?

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?

D Solution design
  K Available solutions?
  D Design new ones
Engineering research questions

Implementation evaluation = Problem investigation
- Stakeholders?
- Their goals?
- Problematic phenomena?
- Their causes?
- Impacts?
- Solution criteria?

A Specification implementation

K Implementation evaluation = Problem investigation

Problem nesting

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

D Solution design
- Available solutions?

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

26th July 2009 ICSOFT09
How can we improve financial evaluation of process-aware information systems?

- Current problems with evaluation?
- Current approaches to financial evaluation?
- Evaluate them

- Develop new approach: Causal loop models
- Validate it

- Reflection: lessons learned

- Problem sequence
- Problem decomposition

- Build taxonomy of approaches
- Classify approaches
- Validate classification

- Criteria for taxonomies?
- Collect taxonomies
- Evaluate
- Design new one
- Validate against criteria

- Make causal loop models of cost factors of PAIS

- Collect modeling guidelines
- Acquire modeling tools

- Experiment to test one model
- Pilot study using another model
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

\[ \text{Implementation evaluation} = \text{Problem investigation} \]

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

\[ \text{Specification implementation} \]

- **A** Specification implementation

\[ \text{Design validation} \]

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

\[ \text{Solution design} \]

- **D** Solution design

- **K** Available solutions?
- **D** Design new ones
Engineering questions at ICSE02
(Mary Shaw reformulated by me)

- How to create X
- How to automate X
- What is a design of X
- What is a better design of X
- How to evaluate X
- How to choose between X and Y

Validation
Design

- Curiosity-driven engineering
- Aim for radical technology?
Research questions at ICSE02 (Mary Shaw reformulated by me)

- s.o.t.p. of X?
- s.o.t.a. of X?
- Does X exist?
- Model of X?
- Kinds of Xs?
- Properties of X?
- Property P of X?
- Relationships among Xs?
- What is X given Y?
- How does X compare to Y?

Explanatory research of the kind “Why does this happen?” is absent
Design science and software engineering

Theories
Levels of knowledge

\[ N = \infty \]

\[ N = k \]

\[ N = 1 \]

Context

\[ X \rightarrow \text{Effects, should satisfy Goals} \]

Solution

Diagnostic theories about problem mechanisms

Treatment theories about solution mechanisms

Risk: other effects than predicted

Risk: "side" effects that do not satisfy goals

Abstraction

Non-mission-oriented science

Design science: solving classes of practical problems

Engineering, consultancy: solving known problems

Universal theories

Public

Confidential

26th July 2009
Frequently used IS theories

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

1. Diffusion of innovations theory

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

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

*Methods mixed up with measuring instruments*
# Validation methods in SE

<table>
<thead>
<tr>
<th>Zelkowitz &amp; Wallace 1998</th>
<th>Description</th>
<th>This talk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lessons learned</td>
<td>Study of documents produced by a project</td>
<td>Data analysis method (Conceptual analysis)</td>
</tr>
<tr>
<td>Static analysis</td>
<td>Studying a program and its documentation</td>
<td>Measuring instrument (Primary sources)</td>
</tr>
<tr>
<td>Replicated experiment</td>
<td>Several projects are staffed to perform a task in multiple ways</td>
<td>Research method (field experiment)</td>
</tr>
<tr>
<td>Synthetic environment experiment</td>
<td>Several projects are performed in an artificial environment</td>
<td>Research method (lab experiment)</td>
</tr>
<tr>
<td>Dynamic analysis</td>
<td>Instrumenting a software product to collect data</td>
<td>Measuring instrument (monitoring devices)</td>
</tr>
<tr>
<td>Simulation</td>
<td>Executing a product in an artificial environment</td>
<td>Research method (lab experiment)</td>
</tr>
</tbody>
</table>
## Validation methods in design science

<table>
<thead>
<tr>
<th>Scaling up</th>
<th>Cond. of pract.</th>
<th>Cntrl of cntxt</th>
<th>Unit of data c.</th>
<th>Example</th>
<th>User</th>
<th>Goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Illustration</td>
<td>no</td>
<td>yes</td>
<td>model</td>
<td>small</td>
<td>designer</td>
<td>illustration</td>
</tr>
<tr>
<td>Opinion</td>
<td>imagined</td>
<td>yes</td>
<td>model</td>
<td>any</td>
<td>Stakeh.</td>
<td>support</td>
</tr>
<tr>
<td>Lab demo</td>
<td>no</td>
<td>yes</td>
<td>model</td>
<td>realistic</td>
<td>designer</td>
<td>knowledge</td>
</tr>
<tr>
<td>Lab expt.</td>
<td>no</td>
<td>yes</td>
<td>model !</td>
<td>artificial</td>
<td>designer</td>
<td>knowledge</td>
</tr>
<tr>
<td>Benchmark</td>
<td>no</td>
<td>yes</td>
<td>model</td>
<td>standard</td>
<td>designer</td>
<td>knowledge</td>
</tr>
<tr>
<td>Field trial</td>
<td>yes</td>
<td>yes</td>
<td>model</td>
<td>realistic</td>
<td>designer</td>
<td>knowledge</td>
</tr>
<tr>
<td>Field experiment</td>
<td>yes</td>
<td>yes</td>
<td>model</td>
<td>realistic</td>
<td>Stakeh.</td>
<td>knowledge</td>
</tr>
<tr>
<td>Action case</td>
<td>yes</td>
<td>no</td>
<td>model</td>
<td>real</td>
<td>designer</td>
<td>Knowledge and change</td>
</tr>
<tr>
<td>Pilot project</td>
<td>yes</td>
<td>no</td>
<td>model</td>
<td>realistic</td>
<td>Stakeh.</td>
<td>knowledge</td>
</tr>
<tr>
<td>Case study</td>
<td>yes</td>
<td>no</td>
<td>model</td>
<td>real</td>
<td>Stakeh.</td>
<td>Knowledge and change</td>
</tr>
</tbody>
</table>

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