Open your mind. LUT.

Lappeenranta University of Technology

Approaches to Software Engineering as Human Activity

Kari Smolander

Software Engineering and Information Management

Lappeenranta University of Technology

kari.smolander@lut.fi

Kari Smolander

Contents

- What kind of research is software engineering?
  - Is there a human component in it?
- Research approaches in software engineering
  - How to study software development as human activity?
  - A classification by Järvinen (2004) on possible research approaches
- An example of empirical research in software testing

My own background

- Ph.D., Lappeenranta University of Technology, 2003
- Professor, Lappeenranta University of Technology, 2006-
- Industry experience
  - MetaCase Consulting, co-founder, head architect, 1991-1994
  - Tieto corporation, Department manager, 1994-1997
- Previous research
  - 1990s in Jyväskylä, Finland, metamodelling, method engineering, MetaEdit CASE tool development
  - My Ph.D. thesis, 2001-2003: Empirical field research in software development organizations about their architectural practices
- Current research
  - Empirical research in various areas of software engineering
  - Observation of the practice of software development

Software engineering

- IEEE 610.2 Standard Glossary of Computer Applications Terminology
  - The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software
- Two views on software engineering
  - Software as a technical artifact
    - The structural qualities of software
    - The qualities of software development tools
  - Software development as an industrial activity
    - Development processes
    - How human organizations work when developing software
    - Practical usability and applicability of tools and development principles

Software development as human activity

- Software is a technical artifact (although there is much human knowledge encoded in it) but **software development** is a human activity
    - How to study human activities?
    - How to measure and evaluate human activities?

- It is (relatively) easy to measure qualities and properties of software, but more difficult to measure and evaluate software development

Open your mind. LUT.
Lappeenranta University of Technology

Software development as human activity

- What affects on human activities?
  - Genetics
  - Attitudes
  - Social norms
  - Perceived behavioral control, experiences on difficulties and rewards
  - Faith, beliefs, religion, philosophy
  - Instincts, fear, habits
  - Etc.
- How to measure these unambiguously in software development?
  - Is it possible?

An example

- **Agile manifesto** (2001)
- This is **not** about software technology
- This is about **human activity/ behaviour**

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	

Scientific research

- An activity that sets and solves research problems
    - How to evaluate meaningfulness of a research problem? How to set research problems?
    - How to do research? What kind of research methods can be used?
- Research is done using methods accepted by the **scientific community**
    - The scientific community is not homogeneous – it is a group of peers that interact through various forums, such as conferences and journals
    - There is not one and only scientific method
    - There are many kind of methods and approaches for different purposes

Scientific research continued

- The solution for a research problem must be justified and/or tested
  - A subjective opinion is not enough
  - Its intersubjective validity is evaluated by the scientific community
    - The scientific community is a broad concept – it contains multiple conceptions on
      - knowledge,
      - scientific methods, and
      - the purposes of science.
  - The relation on prevailing knowledge is important
    - Especially conflicts with prevailing knowledge must be proved
    - If no conflicts → accumulation of knowledge
    - Scientific breakthroughs

Positivistic method

- The method of natural sciences
    - The purpose of science is explanation, finding general laws and causal connections
    - Science is value-free and based on objective, measurable facts
    - The researcher is an independent observer
    - Studies and measurements can be repeated any time and by anyone with the same results
    - Research results increase the amount of scientific knowledge
    - The accumulated scientific method forms an internally harmonious system (no conflicts)

Kari Smolander

Is positivistic method the criterion of science?

- In natural sciences only the research that fulfils the criteria in the previous slide is acceptable
  - The method of natural sciences is the typical view of scientific method
- The method of human sciences
  - The positivistic method does not work in social sciences where the understanding and interpretation of human activity is in the center
    - How to measure unambigously attitudes, social norms, beliefs, perceptions, fears, motives, etc.?
    - The researchers cannot be completely neutral, independent, and value-free
    - The presence of a researcher may have an effect on the studied activity
    - It is often not possible to repeat the study with unchanged conditions

- Is software engineering a natural science?

Does software engineering study humans?

– An example: a researcher wishes to study how agile methods can be taken into use in a very large software organization

– Software engineering research must often evaluate the actions of individuals and human organizations
  – It may be essential to understand the motives, values, expectations and objectives of individuals and organizations
  – ➔ Software engineering must take also humans as the research subjects

Järvinen: On research methods (2004)

Kari Smolander
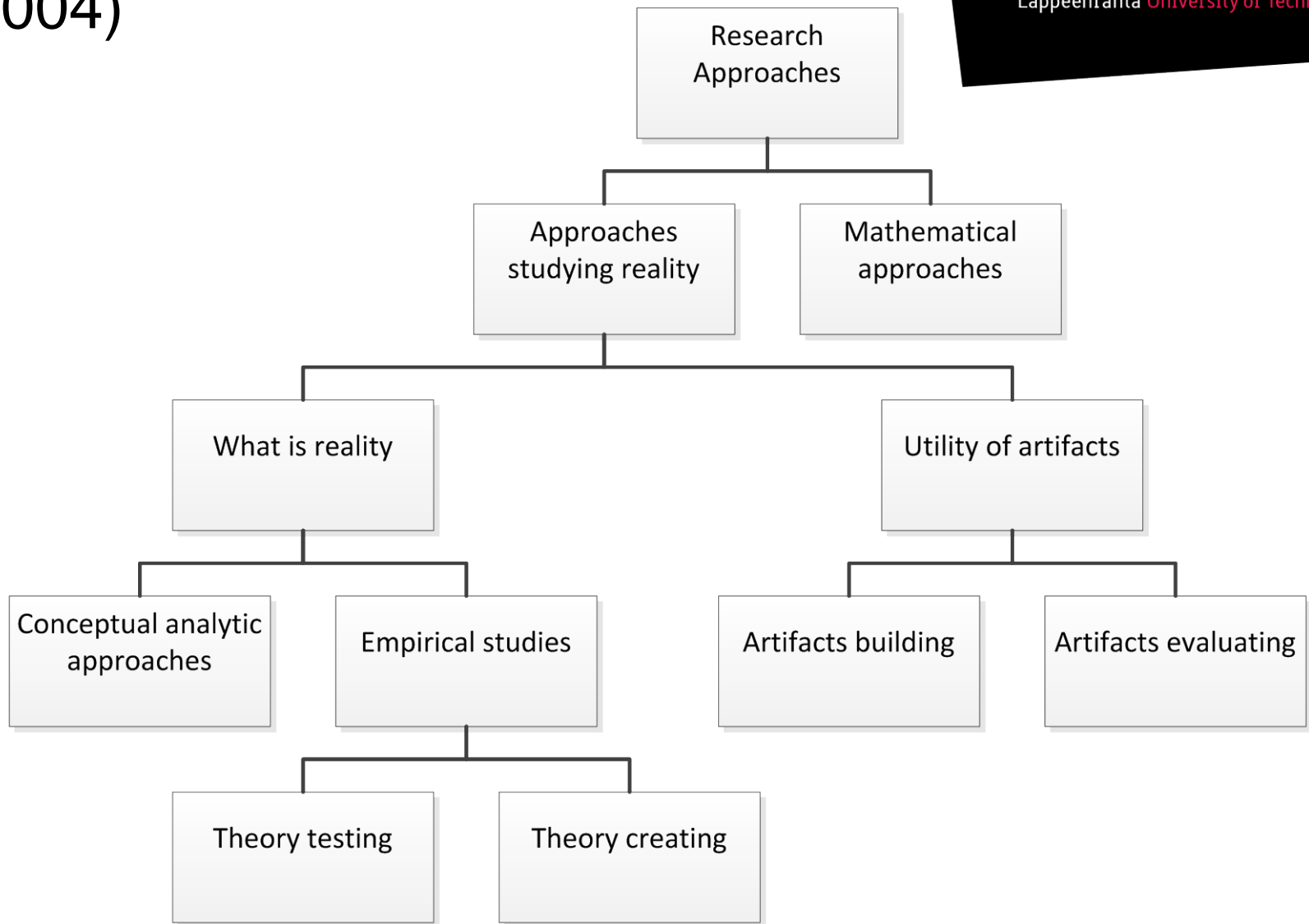
Mathematical approaches

- Proving of mathematical theorems

- Used in computer science – not very relevant in software engineering

Approaches studying reality

Kari Smolander
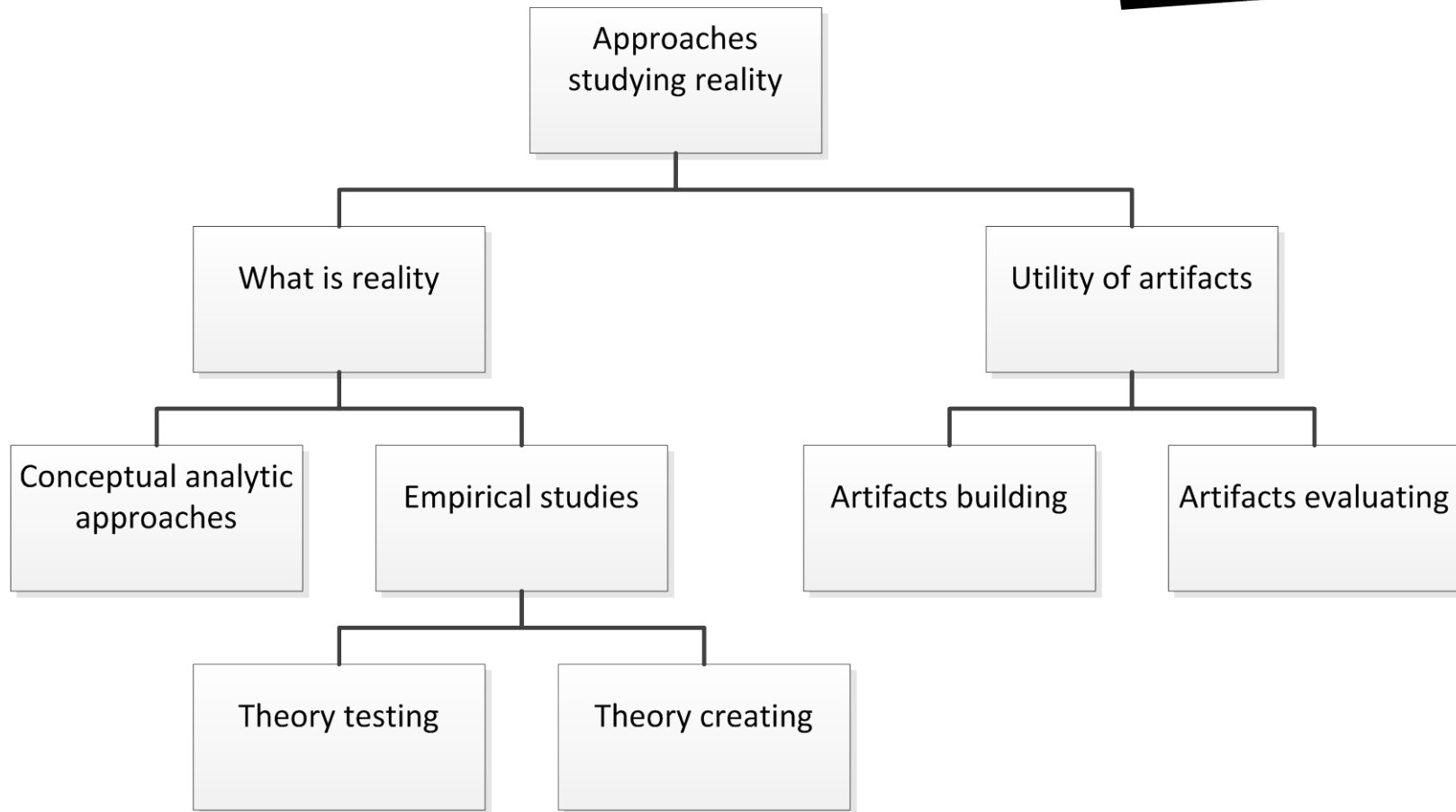
Conceptual-analytic approaches

- Analysis of assumptions behind theoretical constucts
- Identification of theories, models, and frameworks in earlier research
- Includes logical reasoning

- Does not require original observations of "reality"

- Possible to use in software engineering

- Problem: what parts of the results are based on opinions and intuition instead of scientific inquiry?

Theory testing approach

- Laboratory experiment, controlled experiment
- Survey – collect and analyze statistical data from a population
- Field study, field test

- The theory is taken from literature or developed/refined

- The study tests if the theory is "true"

- Problem: how to select and develop the theory? Where does it come from?

Theory creating approach

- A multitude of approaches
  - Case studies
  - Ethnography
  - Grounded theory
  - Discourse analysis
  - Etc.
- Observation of the practices, "reality" → theoretical conclusions
- Various methods of data collection
  - Interviews, direct observations, collecting archive material, …
- Inductive theory creation → observations are refined and classified into theories

- Problem: is it possible to generalize the theories to other contexts?

Building new artifact as research approach

- A particular construct, method, or model is applied and an artifact is produced
- The utility aspects of the artifact are considered
- The output is reported and the usefulness of the artifact is discussed

- A very common approach in engineering

- Design research – the science of the artificial
  - Or: studying the design process itself

- Problem: we have the artifact, so what? How to evaluate it? What is its scientific value?

Evaluating the artifact

- The built artifact is evaluated according to designed criteria
- Measurements/observations are performed
- Can be a controlled experiment, but in real-life that is seldom possible

- Action research:
    - Diagnose the problem
    - Design an intervention or an artifact to solve the problem
    - Do the intervention or take the artifact into use
    - Evaluate the intervention
    - Learn from the evaluation

RE: Research approaches (Järvinen, 2004)

Open your mind. LUT.
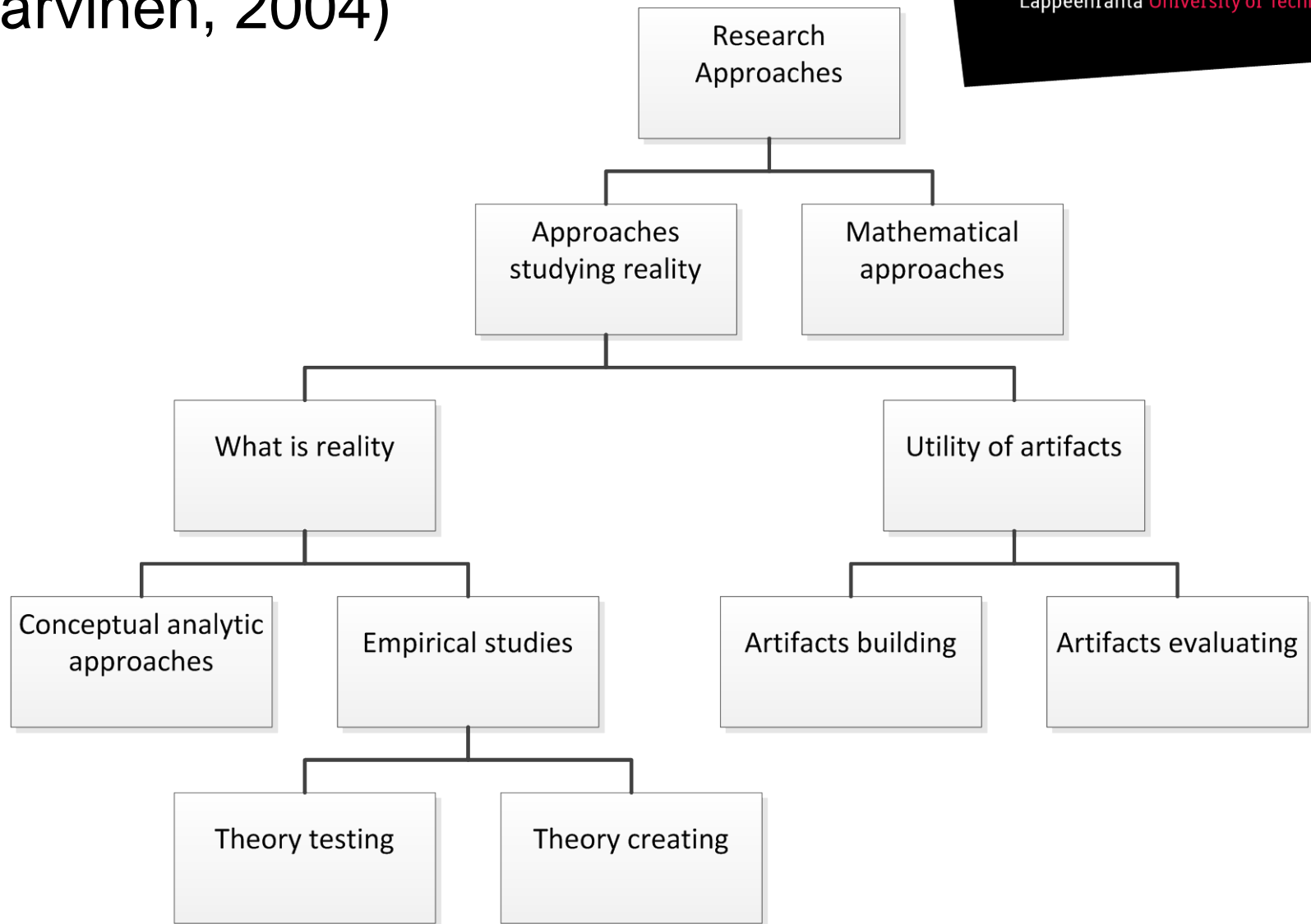
Lappeenranta University of Technology

Example: ESEM 2010

Test Case Selection and Prioritization: Risk-Based or Design-Based?

Jussi Kasurinen, Ossi Taipale, <u>Kari Smolander</u>
Software Engineering and Information Management
Lappeenranta University of Technology
Lappeenranta, Finland

Kari Smolander

Content

- Introduction to the research area
  - How are test cases selected in the industry?
- Using grounded theory as the research method
  - Data collection and analysis
- Observations on test case selection
  - A set of hypotheses
  - Two stereotypical approaches
- Conclusions

Introduction:
testing and test case selection

- Testing is an expensive process
  - One estimate: 50% of total development costs (Kit, 1995)
- Full-coverage testing is in practice impossible
  - Exponential growth in size and complexity
- "Let go – deliver now and correct later" causes too much expenses in the long run
- → a strategy or method for test case selection is needed in any real-world development effort

Research question

- How real-world software organizations select their approach to test case selection?
    - Observation of the practices
    - Identifying how organizations select their test cases
    - Analyzing and explaining why they apply this type of approach
- Belongs to a series of empirical studies of testing practice (2004-), e.g.
    - Testing process problems and enhancement strategies
    - Testing resources
    - Test automation in practice
    - Testing outsourcing

Research process: background

− Testing is a complex phenomenon in practice

  − A human, organizational activity with different approaches even in similar organizations

  − No existing theory-base that could adequately explain this complex, human, organizational activity

  − → a decision to approach the subject qualitatively by observing the practice

  − Grounded theory as the research method

    − Observing and describing real-life testing practice within its social and organizational context

  − Organizational unit as the unit of observation and analysis

    − A part of an organization that deploys one process or has a coherent process context – operates within a set of business goals and policies

Research process: data collection

- Three data collection rounds
  - 12 focus organizations (theme-based interviews, Rounds 1 & 2)
  - A survey in 31 organizations (Round 2 – not reported here)

Round type	Number of interviews	Interviewee role	Description	Themes
1) Semi-structured	12 focus OU interviews	Designer or Programmer	The interviewee was responsible for or had influence on software design.	Design and development methods, Testing strategy and methods, Agile methods, Standards, Outsourcing, Perceied quality
2) Structured with Semi-structured	31 OUs, including 12 focus OUs	Project or Testing manager	The interviewee was responsible for the sofware project or testing phase of the software product.	Test processes and tools, Customer participation, Quality and Customer, Software Quality, Testing methods and resources
3) Semi-structured	12 focus OU interviews	Tester or Programmer	The interviewee was a dedicated tester or was responsible for testing the software product.	Testing methods, Testing strategy and resources, Agile methods, Standards, Outsourcing, Test automation and services, Test tools, Perceived quality, Customer in testing

Data collection: focus organizations

OU	Business	Company size / Operation
Case A	MES producer and electronics manufacturer	Small / National
Case B	Logistics software developer	Large / National
Case C	ICT consultant	Small / National
Case D	Internet service developer and consultant	Small / National
Case E	Naval software system developer	Medium / International
Case F	Safety and logistics system developer	Medium / National
Case G	Financial software developer	Large / National
Case H	ICT developer and consultant	Large / International
Case I	Financial software developer	Large / International
Case J	SME business and agriculture ICT service provider	Small / National
Case K	MES producer and logistics service systems provider	Medium / International
Case L	Modeling software developer	Large / International
19 survey-only cases	Varies; from software consultancies to software product developers and hardware manufacturers.	Varies

Research process: analysis

- Grounded theory process was followed:
    - Open coding
        - Search for pertinent items and phenomena in the data: codes and categories
        - Use of "seed categories" derived from the research question
        - Open coding produced 166 codes grouped into 12 categories
    - Axial coding
        - Identification of causal conditions or any kinds of connections between the categories
        - Collecting chains of evidence from the data
    - Selective coding
        - Selection of the core category and relating it to other categories
        - "applied test case selection approach" in relation to other categories that explain it

Test case selection: developed categories

Category	Description
Applied selection approach	The method the organization is currently using to select which test cases are included in the test plan.
Software type	The type of software the OU is developing.
Test designers	The personnel responsible for designing and selecting the test cases.
Development approach	The method the organization is currently using to develop software.
Testing resources	An approximation on how large an amount of testing resources the organization currently has access to, in comparison to the optimal, ie. perfect amount of resources.
Customer influence	The type and method of customers to influence the organization's software test process.
Selection problem	The most common process hindrance the test case selection method causes to the organization.
Explorative testing	Does the organization apply non-predefined test cases in their test plan?

Test case selection: observations in focus organizations

Case	Applied selection method	Software type	Test designers	Development approach	Testing resources	Customer influence	Test case selection problem	Explorative testing
A	Risk-based with changes first	Software module for hardware	Programmers	Plan-driven supported by agile	Low	Approves product	Important test cases are discarded	Yes, programmers do it.
B	Risk-based	Software product	Designers	Agile	Moderate	Participates in testing	Agile products seem to be difficult to test.	No, only defined cases are tested.
C	Risk-based with changes first	Software product	Programmers with clients	Agile	Moderate	Participates in testing	Some test cases are not implemented.	Yes, programmers do it.
D	Risk-based	Software service	Programmers	Plan-driven supported by agile	Low	Approves testing plan	Some test cases are not implemented	Yes
E	Risk-based	Software module for hardware	Programmers	Agile supported by plan-driven	High	Approves product	Important test cases are discarded	Yes, some phases apply.
F	Risk-based with conformance	Software module for hardware	Designers	Plan-driven	Moderate	Approves product	Some test cases are not implemented	Yes
G	Design-based with conformance	Software service	Test manager with testers	Plan-driven	High	Approves testing plan	Validating functionalities is difficult.	No, only defined cases are tested.
H	Design-based	Software service	Designers with clients	Plan-driven	High	Approves testing plan	Amount of policies affect test effectiveness.	No, not enough time.
I	Design-based	Software service	Test manager with testers	Plan-driven	High	Approves design	Too large reliance on test manager experience	No
J	Risk-based, changes first	Software product	Project manager	Plan-driven supported by agile	High	Participates in testing	Important test cases are discarded	Yes
K	Design-based	Software module for hardware	Project manager, clients	Plan-driven supported by agile	Moderate	Participates in test design	Some test cases are not implemented	Yes, in some projects.
L	Design-based	Software product	Project manager with designers	Plan-driven	High	Approves product	Test management in large projects	Yes, several phases apply.

Test case selection:
a set of hypotheses

- Cross-case comparison as an explanatory tool
  - For example, in several cases, design-based approach appeared to exist together with plan-driven product development
- We were able to classify the selection method to either risk-based or design-based (with variations)
  - Risk-based selection
    - "What causes the largest expenses when broken?"
  - Design-based:
    - "Which are the main functionalities the software is supposed to do?"
  - How to describe and explain the difference between these methods?
  - What is their effect on testing practice and management?
- As the result, four hypotheses were derived

Test case selection: four hypotheses

- Hypothesis 1: Risk-based selection is applied when the software design is not fixed at the design phase
    - Risk-based selection was preferred when
        - The organization used agile methods
        - A customer had a direct influence on the later parts of the process
    - Design-based selection co-occurred with
        - Plan-driven methods
        - Indirect customer influence
- Hypothesis 2: The design-based approach is favored in organizations with ample resources and it requires more management
    - Organizations using the design-based approach had more testing resources (73%) than the others (49%)
    - The process difficulty differed:
        - Risk-based: test coverage – including all critical cases
        - Design-based: managing and planning the testing process

Test case selection: four hypotheses

- Hypothesis 3: The use of test automation is not affected by the case design or case selection approach
  - We identified no pattern of the feasibility of automation in relation to test case selection
- Hypothesis 4: Exploratory testing may be seen by policy-makers as an unproductive task because of its ad hoc nature
  - In exploratory testing testers do non-predefined activities as a part of standard process
  - Risk-based selection co-existed with high level of exploratory testing
  - Design-based selection and large organizations were less related to exploratory testing
  - Exploratory testing is difficult to document and therefore causes additional requirements for management and policies
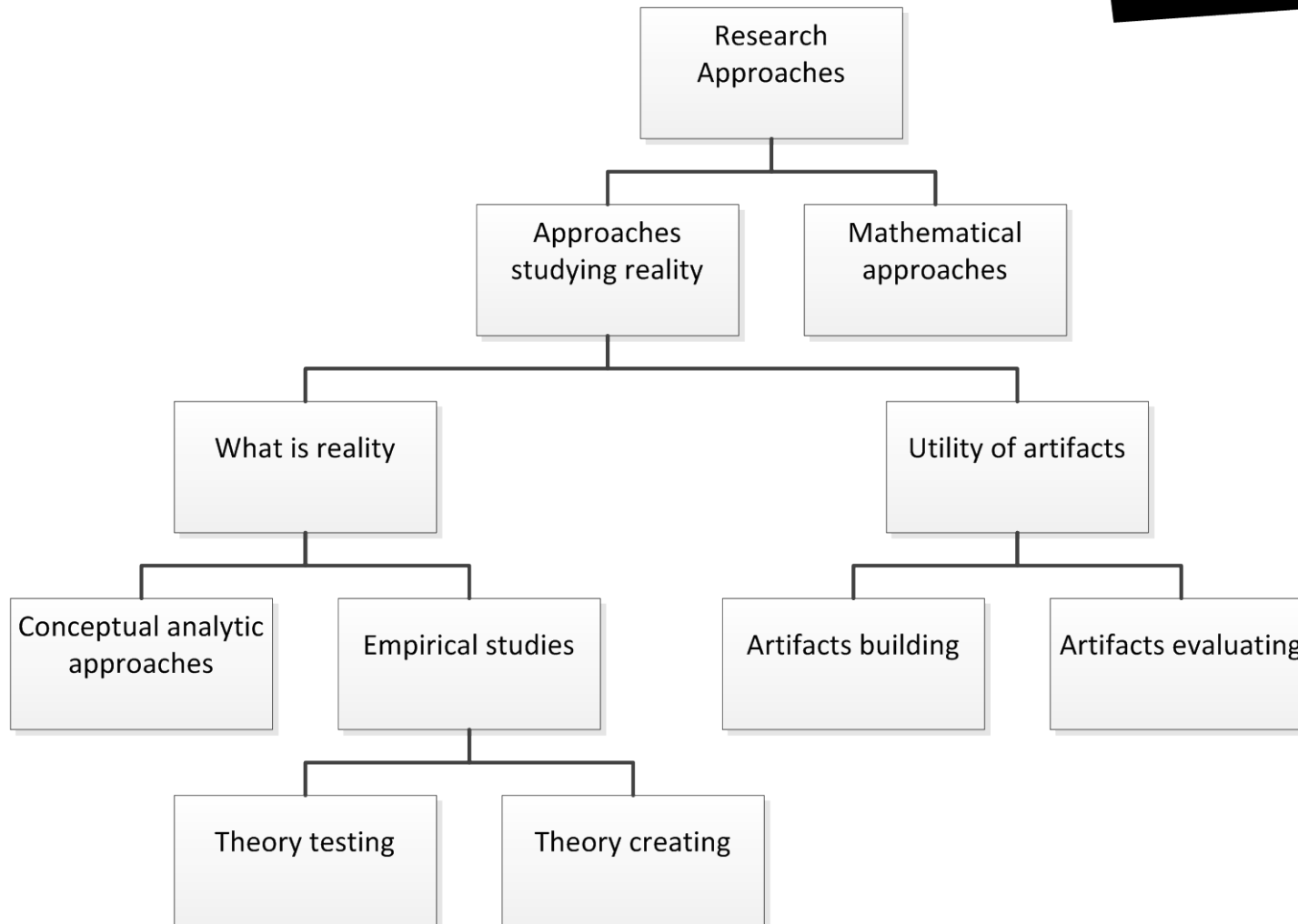
Test case selection:
two stereotypical approaches

Category	Risk-based selection	Design-based selection
Test designers	Developers: programmers and testers	Managers: test and project managers
Development approach	Leans towards agile methods	Leans towards plan-driven methods
Testing resources	Limited	Sufficient
Explorative testing	Applied commonly	Applied rarely
Effect of policies in decisions on testing.	Small; most decisions done in project level.	Large; most decisions are based on company policies or customer requirements.
Customer influence	In the testing process	In the design process
Limitations of the model	Test case coverage may become limited.	Test process may become laborous to manage
Design concept	"What should be tested to ensure smallest losses if the product is faulty?"	"What should be tested to ensure that the product does what it is intended to do?"

Conclusions

- Two main approaches to test case selection were identified: risk-based and design-based
- Risk-based co-occurred with limited testing resources and higher flexibility requirements
- Design-based co-occurred with more sufficient resources and plan-driven processes
- Awareness of this distinction between the approaches helps software organizations to understand and enhance their testing practices

What was the research approach?

Thank you!

Questions? Comments?