Theories from Case Studies in SE and IS research

Roel Wieringa
University of Twente
The Netherlands
http://wwwwhome.ewi.utwente.nl/~roelw/
 r.j.wieringa@utwente.nl
Based on Wieringa, R.J. (2014) *Design science methodology for information systems and software engineering*. Springer Verlag
Day 1
• 9:00
  – What is a case study (Chs 8, 17-19)
  – When to use case studies (Chs 1, 3, 5, 7)
• 10:30 Break
• 10:45
  – Scientific theories (ch 8)
• 12:15 Lunch
• 13:45
  – Scientific theories (chs 8, 9)
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle (chs 10, 11)
• 10:30 Break
• 10:45
  – Case-based inference (chs 12, 14, 15)
• 12:15 Lunch
• 13:45
  – Exercise (Make a case-based design of your research)
• 15:15 Break
• 15:30
  – Assignments, discussion, wrapup (checklist application to two papers)
• 17:00 End of day 2
Day 1
• 9:00
  – What is a case study
  – When to use case studies
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle
• 10:30 Break
• 10:45
  – Case-based inference
• 12:15 Lunch
• 13:45
  – Exercise (Make a case-based design of your research)
• 15:15 Break
• 15:30
  – Assignments, discussion, wrapup (checklist application to two papers)
• 17:00 End of day 2
What is a case

• **A case** is an instance of a class of systems.
  – Systems may be social, technical, and physical
  – They may be natural or artificial
  – They have a structure that produces overall system behavior

• A case displays some partly known, partly understood systematic behavior

• Examples:
  – A *software system*
  – An *information system consisting of software, hardware, roles, procedures, data*
  – A *product development project*
  – A *global SE project*
  – An *e-business network*
What is a case study

• A **case study** is a scientific investigation of phenomena in a case

• Observational case studies
  – *Study of an agile SE project*

• A single-case mechanism experiment in the field
  – *Test of a software prototype in the real world*

• Technical action research
  – *Use of an experimental development method by a researcher for a client*
Characteristics of case studies

• A case study is a scientific investigation of phenomena in a case
  – Takes place in the field (not in the lab)
  – May be observational or experimental
  – Purpose is to acquire knowledge and sometimes also to help a client

• Examples
  – Observational study of political causes of IS implementation failure
  – Observational study of global requirements engineering
  – Experimental evaluation of network performance in a company
  – Field test of an algorithm for cruise control
  – Use of an experimental IT security risk assessment method to help a client

• Non-example
  – Simulation of different patient test scheduling algorithms (not in the field)
Case-based versus sample-based reasoning

- Case-based reasoning:
  1. Describe phenomena a single instance of a class
  2. Explain in terms of mechanisms, causes or reasons
  3. Generalize by analogy to similar cases

- Sample-based reasoning
  1. Select random sample from a population
  2. Describe sample statistics
  3. Infer population parameter
  4. Explain in terms of mechanisms, causes, or reasons
  5. Generalize by analogy to similar populations
1. Select random sample from a population

- Sampling is drawing tickets from a box
- Population must be defined
- Sample may be simple (without replacement), this needs correction factor in the inference

In case studies, the population is often incompletely defined.
- We call it "class of cases", "class of socio/physical/digital systems"
- Sampling is done sequentially (case by case)
2. Describe sample statistics

• Average, median, mode, variance, etc. of some variables.
  – Clean the data: outlier removal, scale transformations

• Descriptive validity

  • In sample-based research, we are interested in sample properties
  • In case-based research, we are interested in an individual’s properties
3. Statistically infer population parameter

- The average, median, mode, variance, etc. of some variables in the population are unknown.

- Hypothesis testing:
  - Assume hypothesis about population parameter,
  - Compute probability of the sample data, given this hypothesis
  - Conclude about plausibility of the hypothesis

- Confidence interval estimation
  - Estimate an interval, in which the population parameter lies in 95% of the times that this research will be replicated

- **Statistical conclusion validity**

  - No statistical inference from cases to population
4. Explain in terms of mechanisms, causes, or reasons

- *E.g. explain difference in maintainability in terms of comments; explain this in turn by cognitive anchoring mechanisms*
- This is a theory of the researchers. Goes beyond the data.
- **Internal validity**
  - Sample-based explanations often refer to causes
  - Case-based explanations often refer to mechanisms.

- We return to this later
4. Generalize by analogy to similar populations

- *E.g. generalize from a study of maintainability in an SE project to all SE projects.*
- Basis of the analogy must be similarity in the underlying mechanism that produces the population property.
- **External validity**
  - Analogic reasoning, same as in case studies.
  - We return to this later
Day 1
• 9:00
  – What is a case study
  – **When to use case studies**
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle
• 10:30 Break
• 10:45
  – Case-based inference
• 12:15 Lunch
• 13:45
  – Exercise (Make a case-based design of your research)
• 15:15 Break
• 15:30
  – Assignments, discussion, wrapup (checklist application to two papers)
• 17:00 End of day 2
When to do a case study

- When the phenomenon of interest cannot be produced in the lab

- Yin:
  - When there are more variables than data points

- Roel:
  - Then you are not focussing your research well enough
Example case studies

• Which phenomena can (not) be produced in the lab?
  – Political causes of IS implementation failure
  – Coordination in global requirements engineering
  – Company network performance
  – Performance of an algorithm for cruise control
  – Usability and utility of an IT security risk assessment method
  – Performance of patient test scheduling algorithms
  – Your examples and non-examples ...
Elaboration for design science

- **Why** study phenomena in design science that cannot be produced in the lab?
Design science

• Design science is the *design* and *investigation* of artifacts in context
  – A.k.a. engineering science
  – Technical science

To design an artifact to improve a problem context

Problems & Artifacts to investigate

Knowledge, Design problems

To answer knowledge questions about the artifact in context

Solving design problems; Primary goal is utility; Design cycle

Answering knowledge questions; Primary goal is truth; Empirical cycle
Framework for design science

**Social context:**
Location of stakeholders

- Source of money and relevance.
- These come and go

**Goals, budgets**

**Designs**

**Design science**

**Improvement design**
Existing problem-solving knowledge, Old designs
New problem-solving knowledge, New designs

**Answering knowledge questions**
Existing answers to knowledge questions
New answers to knowledge questions

**Knowledge context:**
Mathematics, social science, natural science, design science, design specifications, useful facts, practical knowledge, common sense, other beliefs

- Source and destination of theories
- Theories are forever

3-4 Nov 2015 © Roel Wieringa
The engineering cycle

Treatment Implementation
• Transfer to problem context!

Implementation evaluation = Problem investigation
• Stakeholders? Goals?
• Conceptual problem framework?
• Phenomena? Explanations?
• Effects? Contribution to Goals?

Treatment validation
• Context & Artifact \( \rightarrow \) Effects?
• Effects satisfy Requirements?
• Trade-offs for different artifacts?
• Sensitivity for different Contexts?

Treatment design
• Specify requirements!
• Requirements contribute to goals?
• Available treatments?
• Design new ones!

? = knowledge question
! = action
The design cycle

Implementation evaluation = Problem investigation

- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Treatment design

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

Treatment validation

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment Implementation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?

Treatment validation

- Transfer to problem context!
- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Implementation evaluation = Problem investigation

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?
When to use case studies

• Basic research spends its research budget on simulating idealized conditions in the lab
• Design science spends its research budget on approximating conditions of practice
• Case studies needed for
  – Problem investigation (observational case study)
  – Technology validation (simulation, TAR)
  – Implementation evaluation (observational case study)
More realistic conditions of practice

• Scaling up:
  – Lab research: lab tests, simulation.
  – Opinion research: Expert opinion focus groups
  – Field research: Field experiments, action research (performed by researcher), pilot projects (performed by stakeholders)
• Scaling up:
  – Lab research: lab tests, simulation.
  – Opinion research: Expert opinion focus groups
  – Field research: Field experiments, action research (performed by researcher), pilot projects (performed by stakeholders)
Discussion

• What reasons do you have to do a case study?
  – What phenomenon?
  – Why? Problem investigation, technology validation, implementation evaluation?
Sciences of the middle range

Generalization

Universal generalization

Existential generalization

Case description

Basic sciences
Physics, Chemistry, parts of Biology

Special sciences (about the earth):
Biology, Psychology, Sociology, ...

Applied sciences:
Astronomy, Geology, Meteorology, Political sciences, Management science, ...

Design sciences:
Software engineering, Information systems,
Computer sciences, Electrical engineering,
Mechanical engineering, ...

Case research:
Engineering, Consultancy,
Psychotherapy, Health care,
Management, Politics, ...

Realism

Idealized conditions
Realistic conditions
Conditions of practice
Day 1
• 9:00
  – What is a case study
  – When to use case studies
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle
• 10:30 Break
• 10:45
  – Case-based inference
• 12:15 Lunch
• 13:45
  – Exercise (Make a case-based design of your research)
• 15:15 Break
• 15:30
  – Assignments, discussion, wrapup (checklist application to two papers)
• 17:00 End of day 2
• A **theory** is a belief that there is a pattern in phenomena.
  – Speculations: “*The NSA is monitoring all my email*”
  – Idealizations: “*Merging two faculties reduces cost in theory, not in practice.*”
  – Opinions: “*The Dutch lost the competition because they are not a team.*”
  – Wishful thinking: “*My technique works better than the others.*”
  – Scientific theories: *Theory of electromagnetism*
Scientific theories

• A **scientific** theory is a theory that
  – Has survived tests against experience
    • Observation, measurement
    • Possibly: experiment, simulation, trials
  – Has survived criticism by critical peers
    • Anonymous peer review
    • Publication
    • Replication

• Examples
  – *Theory of electromagnetism*
  – *Theory of cognitive dissonance*
  – *The Balance theorem in social networks*
  – *Theories X, Y, Z, and W of (project) management*
  – *Technology Acceptance Model*
Theories are fallible

• Fallibilism: All theories may be wrong!
  – Outside mathematics there is no certainty
  – And even there we may make mistakes (Lakatos’ proofs and Refutations)

• All theories are improvable
Falsificationism

• Introduced by the philosopher Karl Popper
  – Mechanical version: If the prediction of a theory is contradicted by observation, then reject the theory.
  – Sophisticated version: If the prediction of a theory is contradicted by observation, then
    • publish this,
    • try to replicate it,
    • try to understand it, and
    • try to improve the theory so that it can deal with the observation.
The structure of scientific theories

1. Conceptual framework
   – Definitions of concepts.

2. Generalizations
   – Express beliefs about patterns in phenomena.
Theory of electromagnetism

• Conceptual framework:
  – Definitions of electric current, electric charge, potential difference, electric resistance, electric power, capacitance, electric field, magnetic field, magnetic flux density, inductance, ..., ... and their units.

• Generalizations
  – Electric charges attract or repel one another with a force inversely proportional to the square of their distance.
  – Magnetic attract or repel one another in a similar way and always come in north-South pairs.
  – An electric current inside a wire creates a corresponding circular magnetic field outside the wire.
  – A current is induced in a loop of wire when it is moved towards or away from a magnetic field
Theory of cognitive dissonance

• Conceptual framework
  – Cognitive dissonance is the mental stress experienced by an individual who holds contradictory beliefs or values, or is confronted by new information that conflicts with existing beliefs or values

• Generalization
  – People engage in dissonance reduction to bring their beliefs and actions in line with one another four ways:
    • Change behavior to agree with belief (“eat less chocolate”)
    • Change belief to justify behavior (“occasional chocolate eating is OK”)
    • Add new intention or behavior (“I’ll work out tomorrow”)
    • Deny information that conflicts with existing beliefs (“this is low-fat chocolate”)
The Balance Theorem in social networks

- **Conceptual framework**
  - Definition of concepts of graph, link, friend/enemy, complete graph (each pair of nodes connected), balanced graph (no --- or ++- triangles)

- **Generalization**
  - Large call networks are almost balanced (V.D. Blondel, A. Decuyper and G. Krings - "A survey of results on mobile phone datasets analysis")
  - Mathematical theorem: If a labeled complete graph is balanced, then
    - either all pairs of nodes are friends,
    - or else the nodes can be divided into two groups, X and Y, such that every pair of nodes in X like each other, every pair of nodes in Y like each other, and everyone in X is the enemy of everyone in Y.

- **Idealizing assumptions may allow us to understand real-world phenomena**
Technology Acceptance Model

- Conceptual framework
  - Definitions of perceived usefulness, perceived ease of use, perceived resources, attitude towards using, behavior intention to use, actual system use

- Generalization

Theories X, Y, Z, and W of (project) management

• Theory X: By using time and motion studies, the efficiency of tasks improves.
• Theory Y: Output of tasks improves if people’s creativity is stimulated.
• Theory Z: Stimulating individual creativity may create conflict; the potential for conflict can be reduced by creating a common culture.
• Theory W: Different organizations have different cultures. Projects involving several organizations produce their result more efficiently if every stakeholder wins something by producing the output.

• All these generalizations make predictions about the effect of an intervention.
• They are embedded in theories that provide explanations too, and define the appropriate conceptual framework.
Day 1
• 9:00
  – What is a case study
  – When to use case studies
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
    • Conceptual frameworks
    • Generalizations
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle
• 10:30 Break
• 10:45
  – Case-based inference
• 12:15 Lunch
• 13:45
  • Exercise (Make a case-based design of your research)
• 15:15 Break
• 15:30
  – Assignments, discussion, wrapup (checklist application to two papers)
• 17:00 End of day 2
• A **conceptual framework** consists of definitions of concepts, often called **constructs**.

• **Variables, relations among variables**
• **Populations, probability distribution of variable.**
• **Entities, attributes, relations, taxonomy, cardinality, ...**
• **Events, actions, processes, procedures, constraints, ...**
• We can define two kinds of conceptual structures

1. **Architectural structures:** Class of physical/social/technical systems, architecture, components with capabilities, interactions

2. **Statistical structures:** Population, variables with probability distributions, relations among variables
Architectural structures

• The world is structured as a collection of systems
  – Social, physical, technical, psychological, biological, ...
  – Each system has components with capabilities and limitations,
  – that can interact with each other in specific ways.
Theory of cognitive dissonance

• Conceptual framework
  – Class of systems: people interacting with the real world
  – Components: beliefs, intentions, values, facts, observations, conflict between facts and observations
  – Capabilities of people:
    • Change behavior
    • Change a value
    • Change intention
    • Deny observation
    • Deny fact
• Conceptual framework provides a way to describe phenomena (a frame)
• but does not itself provide generalizations about phenomena
  – They cannot be falsified!
  – They can be useful, i.e. contribute to a purpose
Other examples

- In all of these examples we recognize in each case/population element an architecture consisting of components that have capabilities to interact.
  - Theory of electromagnetism
  - The Balance theorem in social networks
  - Technology Acceptance Model
  - Theories X, Y, Z, and W of (project) management
More examples

• Examples
  – A software prototype in a simulated context
  – Software engineering projects in the real world
  – Information systems in organizations
  – Risk assessment methods for IT security
  – Business networks

• Which architecture to recognize in a case? That depends on which generalizations we would like to be able to draw.
History of architectural conceptual structures

• This kind of structure is used in
  – The engineering disciplines: Renaissance machines 1500
  – Astronomy: architecture of solar system; math description 1600
  – Physics: forces among physical bodies
  – Biology: structure and mechanisms in the body
  – Chemistry: composition and mechanisms of combustion 1800
  – Sociology: structure and mechanisms of society, organizations, 1900
  – Psychology: cognitive mechanisms
  – Economy: structure and mechanisms of markets
  – Sociology, economy, computer science, structure & mechanisms of networks and games 2000
Advantages of architectural structures

Architectures can be used to decompose complex problems into simpler problems

• Study one component at a time
• Study an architecture while abstracting from internal structure of components

This in turn can be used for

• **Explanation, diagnosis, debugging**: Trace system-level phenomena to component properties.
• **Prediction, design**: Explore the effects of putting different components together.
• **Generalization**: Reason about similarity of cases or populations.
Statistical structures

• The world is structured as a population.
  – A **population** is set of all objects that satisfy a predicate, called the population predicate.
  – A **(random) variable** is an observable property of population elements
  – A **probability distribution** of a variable $X$ is a mathematical function that summarizes the probability of randomly selecting a value of $X$ from the population

• (``Random’’ means: No systematic selection proces. Internal structure of the selection process does not provide information about the outcome of the selection.)
Population

• **Population** is set of all objects that satisfy a predicate, called the *population predicate*.
  
  – *All software prototypes in a simulated context, similar to a particular prototype & context*
  – *All agile software engineering projects in the real world*
  – *All ERP systems in large organizations*
  – *All risk assessment methods for IT security*
  – *All business networks*

• The population predicate is a *similarity predicate*
Variables

• **A (random) variable** is an observable property of population elements

• **X-box** is the multiset of values of X on a population
Statistical reasoning is about X-boxes

• **A (random) variable** is an observable property of population elements

• **X-box** is the multiset of values of X on a population
  – *Weight of people, effort of SE projects, numbers on football shirts.*

• **XY-box** is the multiset of pairs of values of (X, Y) on a population, etc.
  – *Execution time of implementations, accuracy of output, effort spent on a project, duration of a project, (effort, duration) of a project, etc.*

• In a population,
  – Each variable has properties of variation (average, variance)
  – And sets of two or more variables have relationships of covariation
Probability distributions

- **Probability distribution** of X is a mathematical function that summarizes the probability of selecting a sample of values in a random draw from the population.
Chance models

• **Chance model** of X:
  1. Definition of the meaning of numbers in the X-box
  2. Assumptions about probability distribution of X
  3. Measurement procedure
  4. Sampling procedure
Example

• Population of open source web applications
• Random variable ImpV indicates implementation vulnerabilities.
• Chance model of ImpV:
  1. Definition: The numbers on the tickets in the ImpV-box are proportions of implementation vulnerabilities among total number of vulnerabilities in a web application. (pages 564-565)
  2. Assumptions: binomial distribution. I.e. the proportions of implementation vulnerabilities in different web applications are independent, and the probability that a vulnerability is an implementation vulnerability, is constant across all web applications
  4. Sampling procedure: Not specified. 20 applications are listed.
History of statistical conceptual structures

• Statistical conceptual frameworks are used in
  - Social sciences: human populations
  - Physics: statistical mechanics
  - Biology: populations of animals, plants
  - Psychology: groups of people
  - Information systems: populations of organizations
  - Empirical software engineering: populations of projects, software engineers

  1800 • Population-based statistics (descriptive, including regression)

  1900 • Sample-based statistics (statistical inference)

  2000 • Very large sample (population)-based statistics
Advantages of statistical structures

• Statistical structures can be used to make large-scale population properties visible
  – Even when an individual shows no regularities

This in turn can be used to

• Describe aggregate phenomena in a sample
• Generalize from a sample to a population (sample-based)
  – Estimate patterns in the population not visible at the individual level (e.g. identify needs in a population)
  – Estimate the effect of treatments in the population (prediction of policy impact)
• Useful for policy-makers
Hybrid structures

• Statistical study within a case
  1. Case study of a large global SE project: e.g. an architectural study of roles and interactions among software engineers;
  2. Contains a statistical study of the population of software engineers in this project.

• Case study of a population element
  1. Statistical survey of a sample of projects:
  2. Followed by case studies of a few projects, in order to understand mechanisms responsible statistical results.
If the studied phenomena contain people ...

They too have conceptual frameworks
Sharing and interpreting a conceptual framework

- Concepts shared by people in the domain may be adopted by researchers that investigate the domain
  - *Goal, requirement, effort, etc.*
  - Adopting these concepts in the conceptual research framework may allow additional understanding

- Concepts defined by researchers may be adopted by people in the domain
  - *(software) object program structure, agile, etc.*
  - Adopting these concepts in the domain may allow definition of additional options for action

- Concepts may even make a round trip from domain to researchers to domain
Validity of a conceptual framework

• Conceptual framework cannot be true or false
  – A definition cannot be true or false

• It is a tool
  – People may be able to use it (usability)
  – ... for a useful purpose (utility)

• E.g. the different definitions of “goal”, “requirement”, “specification”, design”.

• “A goal is the space between two trees”
Construct Validity

• **Construct validity** is the degree to which the application of constructs to phenomena is warranted with respect to the research goals and questions.

• “Construct validity is the degree to which a test measures what it claims, or purports, to be measuring.”

• “Construct validity refers to the degree to which inferences can legitimately be made from the operationalizations in your study to the theoretical constructs on which those operationalizations were based.”
Threats to construct validity

• Inadequate definition
  – No identification and classification criterion.
  – *E.g. “agile projects” without instruction how to recognize and count agile projects*

• Construct confounding
  – Our cases/sample may be cases/samples from more than one population. Then what is the target of generalization?
  – *Measuring the effect of an SE method in a sample of students.*
    • *Is this a sample of novice software engineers? Well educated software engineers? Captive software engineers?*
Operationalizations

• Some constructs cannot be measured
• **Operationalize** them:
  – Define them in terms of one or more indicators that *can* be measured
  – An **indicator** is a variable that can be measured
  – In software engineering, often called a *metric.*
Threats to validity of operationalizations

• **Mono-operation bias**
  – Defining only one indicator for a construct
  – *E.g. measuring maintainability by effort to repair a bug only (and ignoring effort to find a bug or to test the repair).*

• **Mono-method bias**
  – Indicator measured in only one way.
  – *E.g. measuring effort to repair by questionnaires only*
Construct Validity, again

• **Construct validity** is the degree to which the application of constructs to phenomena is warranted with respect to the research goals and questions

• “Construct validity is the degree to which a test measures what it claims, or purports, to be measuring.”
  – The paradox of analysis: if a vague concept is operationalized, either the operationalization is wrong or the concept is redefined

• “Construct validity refers to the degree to which inferences can legitimately be made from the operationalizations in your study to the theoretical constructs on which those operationalizations were based.”
  – Indicators do not *causally influence* constructs
  – Paradox of analysis: construct validity is either 0 or 100%
Day 1
• 9:00
  – What is a case study
  – When to use case studies
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
    • Conceptual frameworks
    • Generalizations
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle
• 10:30 Break
• 10:45
  – Case-based inference
• 12:15 Lunch
• 13:45
  • Exercise (Make a case-based design of your research)
• 15:15 Break
• 15:30
  – Assignments, discussion, wrapup (checklist application to two papers)
• 17:00 End of day 2
Target (scope) of generalization

• In case-based reasoning: defined by similarity predicate
  – Agile projects: Projects following one of the recognized agile methods such as Scrum or XP. How crisp is this?

• In sample-based reasoning: Defined by similarity predicate too!
  – Population predicate
    • Generalize, from sample of study population of agile projects, to the study population
  – Similar populations
    • Generalize from study population of agile projects to the theoretical population of all agile projects.
The structure of design generalizations

- (Artifact specification) X (Context assumptions) → Effects
  - (UML) X (SE project) → (Less errors)
  - (Cross-functional teams) X (Global SE project) → (Less errors)
  - (Algorithm) X (Context of use) → (Faster execution)

- Effects satisfy a requirement to some extent
  - Often, researchers have no clear requirements
  - Requirements come from stakeholder goals, so the relevance of a generalization depends on stakeholder goals
Discussion

• Do you recognize this?
• More examples ...
Why do we need theories?

• With a conceptual framework we can:
  – **Frame** a problem or artifact: *choose which concepts to use*
  – **Analyze** conceptual problem structure: *analyze the framework*
  – **Describe** a problem or artifact
  – **Generalize** about problem or artifact

• With a generalization we can
  – **Explain** phenomena
  – **Predict** phenomena
Examples

• Theories:
  – Theory of cognitive dissonance
  – Theory of electromagnetism
  – The Balance theorem in social networks
  – Theories X, Y, Z, and W of (project) management
  – Technology Acceptance Model

• What can we do with these theories:
  – Frame a problem or artifact
  – Analyze conceptual problem structure
  – Describe a problem or artifact
  – Generalize about problem or artifact
  – Explain phenomena
  – Predict phenomena
Explanations

• An **explanation** is hypothesis about how a phenomenon came about.
  – **Causal explanations** explain the occurrence of an event by the occurrence of an earlier event
  – **Architectural explanations** explain the existence of a causal relationship by the mechanisms that produced it
  – **Rational explanations** explain the behavior of actors by their goals.

• *E.g. why is the light on?*
Causal explanations

- Causal explanations say that an earlier event made a difference to a current event.
  - Explanation of lower programming effort: Programming effort is lower because we switched to UML
    - The earlier switch to UML resulted in the current reduction of programming effort
    - If we had not switched to UML earlier, our current programming effort would have been higher, other things being equal.

- Causality is difference-making.
  - It is unobservable.
  - It may be nondeterministic
Architectural explanations

• **Architectural explanations** explain the existence of a causal relationship by the mechanisms that produced it.
  - *Explanation why UML leads to lower programming effort:*
    - UML models resemble the domain more than other kinds of models;
    - they are easier to understand for software engineers;
    - They allow seamless transition from domain models to software models.

• A **mechanism** is the collection of interactions among components (entities, actors) that produces a response from a stimulus.
  - Mechanisms may be social, psychological, physical, technical, ...
  - Mechanisms can be observed; capabilities are unobservable
  - Mechanisms may be nondeterministic
Rational explanations

• **Rational explanations** explain the behavior of actors by their goals.
  – *Why do we have lower programming effort? Because we bid for fixed price contracts. We need to be sure that our software engineers work as efficiently as possible*

• Architectural explanation for social systems that include rational actors
  – We assume rationality in users, programmers, attackers, fraudsters, thieves, politicians, ... but their goals may not be our goals.
Predictions

- Explanations say how the present is an outcome of the past.
- Predictions look at the present and estimate what the future will be.

- Some explanations are too incomplete to be used as predictions
  - *Explanations of the outcome of a football match*
- Some predictions are descriptive, and do not provide an explanation
  - *In CMM 3 organizations developing embedded software, defect removal effectiveness is 98%.*
- Some explanations can be used for prediction too
  - *Most examples of explanations given so far.*
The use of theories in design

Implementation evaluation = Problem investigation

- Stakeholders? Goals?
- Conceptual problem framework?
- Phenomena? Explanations?
- Effects? Contribution to Goals?

Treatment implementation

- Explanations & predications needed

Treatment design

- Specify requirements!
- Requirements contribute to goals?
- Available treatments?
- Design new ones!

Treatment validation

- Context & Artifact → Effects?
- Effects satisfy Requirements?
- Trade-offs for different artifacts?
- Sensitivity for different Contexts?
Usability of design theories

• When is a design theory
  (Context assumptions) X (Artifact design) → Effects
  usable by a practitioner?
  1. He/she is capable to recognize Context Assumptions
  2. and to acquire/build Artifact,
  3. effects will indeed occur, and
  4. will contribute to stakeholder goals/satisfy requirements

• Practitioner has to assess the risk that each of these fails
Example

• \((UML) \times (\text{any SE project}) \rightarrow \text{lower programming effort}\)

• You are a practitioner.
  1. Can you recognize Context Assumptions?
  2. Can you acquire UML?
  3. Are you sure the effects will occur?
  4. Do they contribute to stakeholder goals?
Discussion

• Gregor’s theory of theories.
  – My book page 103.
Day 1
• 9:00
  – What is a case study
  – When to use case studies
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle
• 10:30 Break
• 10:45
  – Case-based inference
• 12:15 Lunch
• 13:45
  • Exercise (Make a case-based design of your research)
• 15:15 Break
• 15:30
  – Assignments, discussion, wrapup (checklist application to two papers)
• 17:00 End of day 2
Nutshell

- A **case study** is the study of a system in the field.
- The **conceptual framework** of a case study should define the case architecture.
- Conceptual frameworks should be assessed on the **construct validity** of the constructs and operationalizations in them.
- An **architecture** consists of components that have behavioral capabilities and can interact.
- An **architectural mechanism** is an interaction among case components that produce a response from a stimulus.
- **Theories** of a case should aim to describe, explain or predict system-level phenomena in terms of architectural mechanisms.
  - Design theories: $A \times C \rightarrow E$
  - Middle range
  - Usable
Exercises

• Do question 1 of chapter 9.
Day 1
• 9:00
  – What is a case study
  – When to use case studies
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – **Empirical research cycle**
• 10:30 Break
• 10:45
  – Case-based inference
• 12:15 Lunch
• 13:15
  • Exercise (Make a case-based design of your research)
• 14:45 Break
• 15:00
  • Assignments, discussion, wrapup (checklist application to two papers)
• 16:30 End of day 2
Checklist for the empirical cycle: context

1. Improvement goal?
2. Knowledge goal?
3. Current knowledge?

17. Contribution to knowledge goal?
18. Contribution to improvement goal?

Design cycle (slide 22)

Empirical cycle
4. ...
....
16. ...

Designing something useful
Answering a knowledge question
Data analysis
12. Data?
13. Observations?
14. Explanations?
15. Generalizations?
16. Answers?

Research execution
11. What happened?

Research problem analysis
4. Conceptual framework?
5. Research questions?
6. Population?

Research setup

Empirical cycle

Data analysis

Research & inference design
7. Object of study?
8. Treatment specification?
9. Measurement specification?
10. Inference?

Design validation
7. Object of study justification?
8. Treatment specification justification?
9. Measurement specification justification?
10. Inference justification?
Research setup

- Treatment instrument
- Measurement instrument

Object of Study

Sample

Population Representation

Researcher
• In sample-based studies, "representation" is achieved random sampling
  – A random sample may be **dissimilar** to any other sample from the same population.
  – No analogic reasoning from random sample to population
  – Randomness guarantees long-run stability of statistical properties

• In case-based research, OoS’s are studied sequentially

• In observational studies there is no treatment
  – But all instruments interact with the OoS’s.

• Which inferences you can do, depends on the research setup
Examples

• What is the population, the sample of objects of study, the measurements, and the treatments (if any)?
  – *Experiment with students to test a new SE technique to see if it leads to less programming effort*
  – *Survey of the use of UML in Brazil*
  – *Case study of IS implementation failure*
  – *Experimental use of a new effort estimation technique in a pilot project to see if it is more accurate*
## Kinds of empirical research methods

<table>
<thead>
<tr>
<th>Case-based: investigate single cases, look at architecture and mechanisms</th>
<th><strong>Observational study</strong> (no treatment)</th>
<th><strong>Experimental study</strong> (treatment)</th>
</tr>
</thead>
</table>
| | **Observational case study** | • Expert opinion,  
• Mechanism experiments,  
• **Technical action research** |

| Sample-based: investigate samples drawn from a population, look at averages and variation | Survey | • Statistical difference-making experiment |
Example observational case study

An Empirical Study of the Complex Relationships between Requirements Engineering Processes and Other Processes that Lead to Payoffs in Productivity, Quality, and Risk Management

Daniela Damian and James Chisan
TSE 32(7) July 2006
Research design

• **Research context**
  1. **Improvement goal:** None. Curiosity-driven research.
  2. **Knowledge goal:** Fill gap between claims about RE and RE practice
  3. **Current knowledge:** A few published surveys, no case studies.
Research design

• **Research problem analysis**

4. Conceptual framework: definitions of requirements, productivity, quality, risk

5. Research questions:
   
   – How do improvements in RE processes relate to improvements in productivity, quality, and risk management?

   1. How do improvements in the RE practice impact the early stages of development?

   2. How do they impact the downstream development stages?

   3. Which components of the RE process were more significant in contributing to this impact?

   4. How could the interaction between REP and other processes have contributed to these results?

6. Population: RE processes in software development organizations
Research design

• Research & inference design

7. Object of study: Introduction of RE at the Australian Center for Unisys Software

8. Treatment specification: No treatment. Observational research.


10. Inference: after the break!
Example Technical Action Research (TAR) study

What is TAR?

- Using an experimental technique
  - to help a client and → Helping a client
  - to learn about its effects in practice. → Experimenting with a technique
## TAR methodology

<table>
<thead>
<tr>
<th>Researcher’s design cycle:</th>
<th>Researcher’s empirical cycle:</th>
<th>Client’s engineering cycle:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design a new technique</td>
<td>Validate a new technique</td>
<td>Solve a problem</td>
</tr>
</tbody>
</table>
TAR methodology

**Researcher’s design cycle:**
- Investigate problem
- Design treatment
- Validate treatment

**Researcher’s empirical cycle:**
- Analyze research problem
- Design research setup and inferences
- Validate
- Execute
- Analyse results

**Client’s engineering cycle:**
- Investigate client problem
- Design treatment
- Validate treatment
- Implement
- Evaluate
**TAR methodology**

**Researcher’s design cycle:**
- Investigate problem
- Design treatment
- Validate treatment

**Researcher’s empirical cycle:**
- Analyze research problem
  - Effects?
  - Satisfy requirements?
  - Comparison?
  - Generalizable?
- Design research setup and inferences
  - Acquire a case
- Validate
  - Internal validity?
  - External validity?
- Execute
  - Do the client cycle
- Analyse results
  - Answer questions.

**Client’s engineering cycle:**
- Investigate client problem
  - Stakeholders, goals?
  - Phenomena?
  - Causes, effects?
- Design treatment
  - Make a plan.
- Validate treatment
  - Check with client.
- Implement
  - Do it.
- Evaluate
  - Client satisfied?
Research Design

Research context

1. Knowledge goal
   – To validate a newly developed confidentiality risk assessment method

2. Improvement goal
   – To develop a confidentiality risk assessment method when IT is outsourced.

3. Current knowledge
   – The method has been tested on small problems in the lab
   – The method has been inspected by stakeholders of a client company
Research Design

Research problem

4. Conceptual framework
   – Concepts from outsourcing and security risk assessment

5. Knowledge questions
   • Q1 Is the technique usable?
   • Q2 Is it less subjective than other available techniques?
   • Q3 Does use of the technique improve the client’s understanding of confidentiality risks, better than competing techniques do?
   • Q4 Is the effectiveness of the techniques dependent on its context of use? If so, how?

6. Population
   – The set of all ERP outsourcing relations where the service consumer must satisfy confidentiality requirements checked by auditors and depends on the outsourcing service provider to satisfy these requirements.
Research design

Research & inference design

7. Object of study
   – large multinational industrial company with a significant corporate security staff keen on improving their procedures.

8. Treatment design
   – The RA method designed by the researcher was customized for the company.
   – Some software tools were used.
   – A schedule of interactions was agreed with the company.
Research design

Research & inference design, continued

9. Measurement design

• Usability measured by effort (time) to use;
• Subjectivity measured by number of questions in the method that require personal judgment;
• Client’s understanding measured by debriefing interview
  – Data sources: Architects, Chief security officer, primary documents
  – Instruments: Diary of researcher, Interviews with key stakeholders
  – Measurement plan: Agreements about what data could be collected, and how

• Inference design: After the break!
Discussion


• Driven by client problem
Mapping of ADR to TAR

**Researcher’s design cycle:**
- Investigate problem
- Design treatment
- Validate treatment
- 3. Reflection and learning: Need to improve the technique?

**Researcher’s empirical cycle:**
- Analyze research problem
  - Effects?
  - Satisfy requirements?
  - Comparison?
  - Generalizable?
- Design research setup and inferences
  - Acquire a case
  - Validate
  - Internal validity?
  - External validity?
- Execute
  - Do the client cycle
- Analyze results
  - 4. Formalization of learning: conclude, generalize, publish

**Client’s engineering cycle:**
- Investigate client problem
  - Stakeholders, goals?
  - Phenomena?
  - Causes, effects?
  - Design treatment
  - Make a plan.
- Validate treatment
  - Check with client.
- Implement
  - 2. Intervention
- Evaluate
  - 2. Evaluation: this cycle successful?

---

3-4 Nov 2015
Action Research (AR) cycle (Susman & Evered 1978)
Mapping of AR to TAR

**Researcher’s design cycle:**
- Investigate problem
- Design treatment
- Validate treatment

**Researcher’s empirical cycle:**
- Analyze research problem
  - Effects?
  - Satisfy requirements?
  - Comparison?
  - Generalizable?
- Design research setup and inferences
  - Acquire a case
- Validate
  - Internal validity?
  - External validity?
- Execute
  - Do the client cycle
- Analyse results
  - Answer questions.

**Client’s engineering cycle:**
- Investigate client problem
  - Stakeholders, goals?
  - Phenomena?
  - Causes, effects?
- Design treatment
  - Make a plan.
  - Validate treatment
  - Check with client.
- Implement
  - Action taking
- Evaluate
  - Client satisfied?

TAR

Specifying learning
Day 1
• 9:00
  – What is a case study
  – When to use case studies
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle
• 10:30 Break
• 10:45
  • Case-based inference
    – Inference design
    – Validity
• 12:15 Lunch
• 13:15
  • Exercise (Make a case-based design of your research)
• 14:45 Break
• 15:00
  – Assignments, discussion, wrapup (checklist application to two papers)
• 16:30 End of day 2
Research problem analysis
4. Conceptual framework?
5. Research questions?
6. Population?

Data analysis
12. Data?
13. Observations?
14. Explanations?
15. Generalizations?
16. Answers?

Research execution
11. What happened?

Empirical cycle

Design validation
7. Object of study justification?
8. Treatment specification justification?
9. Measurement specification justification?
10. Inference justification?

Research & inference design
7. Object of study?
8. Treatment specification?
9. Measurement specification?
10. Inference?
Case-based inference design

Case-based research
1. Describe
2. Explain in terms of architecture & mechanisms
3. Generalize by architectural analogy

Sample-based research
1. Describe
2. Statistically infer study population properties
3. Explain in terms of variables & causes
4. Generalize to theoretical population by architectural analogy
Case-based research
1. Describe
2. Explain in terms of architecture & mechanisms
3. Generalize by architectural analogy

Sample-based research
1. Describe
2. Statistically infer study population properties
3. Recognize confusions of variables & causes
4. Generalize to theoretical population by architectural analogy

**Case-based research**

**Sample-based research**

**Descriptive validity**

**Internal validity**

**External validity**
Validity

• Validity is the degree of support for your inferences
  – It is not the degree of truth of your conclusions

• Validity discussion: discuss all of the possible ways in which your inferences could be wrong
  – Legal reasoning: your opponent will try to undercut all your arguments.
Sample—based inference design

Case-based research
1. Describe
2. Explain in terms of architecture & mechanisms
3. Generalize by architectural analogy

Sample-based research
1. Describe
2. Statistically infer study population properties
3. Explain in terms of causes and mechanisms
4. Generalize to theoretical population by architectural analogy
Sample—based inference design

Data -> Description -> Observations

Abduction (Explanation) -> Explanations

Analogy -> Generalizations

Statistical induction

Case-based research
1. Describe
2. Explain in terms of architecture & mechanisms
3. Generalize by architectural analogy

Sample-based research
1. Describe
2. Statistically infer study population properties
3. Explain in terms of causes and mechanisms
4. Generalize to theoretical population by architectural analogy

Descriptive validity
Statistical conclusion validity
Internal validity
External validity

Internal validity

External validity

Descriptive validity
Statistical conclusion validity

3-4 Nov 2015
© Roel Wieringa 114
Descriptive inference design

- **Descriptive inference**
  - i. Data preparation
    - Deal with missing data, outliers, perform transformations, etc.
  - ii. Data interpretation
    - Conceptual analysis of text, images etc., content analysis, grounded theory, etc.
  - iii. Summaries
    - Textual, numerical, graphical, qualitative.

- **Descriptive validity**
  - Triangulation (multiple sources, multiple interpretation methods, multiple coders)
  - Member checking: check with subjects
  - Peer debriefing: check with peers
Statistical inference design

• Statistical inference
  – There can be no statistical inference from a sample of one or more case studies
The study population is a subset of the theoretical population.
What is the basis for generalizing to the theoretical population?
Can we generalize numerical average and variance to the theoretical population?
Targets of generalization

- Case studies are about the analogies among the bullets
Abductive inference design

- **Abduction**
  - What possible explanations of case phenomena can you foresee?
    - Causes?
    - Mechanisms?
    - Reasons?

- **Internal validity**
  - What justifications of these explanations can you foresee?
  - Prior knowledge (common sense, scientific theories)
Analogic inference design

• **Generalization by analogy**
  – What is the scope of your explanation?
  – Similarity predicate of this class of cases? Shared architecture

• **Analytical induction**
  – Select a series of cases that you expect to confirm/disconfirm your generalization. If expectation is violated,
  – Redefine the similarity predicate to define away the problem or
  – Update the generalization to apply to all cases so far.

• **External validity**
  – Under which assumptions is the generalization justified?
Steps in sample-based research

- Define theoretical population (similarity predicate)
- Define study population (list of elements that serves as sampling frame)
- Define chance model
- Select sample
- Do measurements
  1. Describe sample demographics & measurements
  2. Statistically infer study population properties (test a hypothesis about, estimate confidence interval of, population parameter)
  3. Explain in terms of causes, and explain causes in terms of mechanisms
  4. Generalize to theoretical population by architectural analogy
Steps in case-based research

- Define theoretical population (similarity predicate)
- Select next case (analytical induction strategy)
- Do measurements
  1. Describe case architecture & measurements
  2. Explain in terms of architecture & mechanisms
  3. Generalize by architectural analogy
Inferences in example observational case study

An Empirical Study of the Complex Relationships between Requirements Engineering Processes and Other Processes that Lead to Payoffs in Productivity, Quality, and Risk Management

Daniela Damian and James Chisan
TSE 32(7) July 2006
A causal model presented in the paper.

Practitioners’ opinions!

Researcher-observed practitioner-perceived correlations
The variables characterize the way of working of a cross-functional team, and its effects.
An architectural model constructed from the paper.

Mechanisms that explain improved productivity, quality, risk mgmt:

**C-F team**
- improves shared feature understanding,
- reduces rework

**C-C board**
- prevents requirement creep

Which inferences can we now support?

- **Abduction:**
  - It is plausible that this architecture promotes these mechanisms, which contributed to the effect
  - Other mechanisms in this case that may have contributed to these effects:
    - New management may have impacted the effect variables as well
    - Other processes were changed too
    - The product was mature

- **Generalization:**
  - About 150 members, small teams, tech, project, product mgrs
  - Ineffective customer communication

- Need analytical induction to say more
Example Technical Action Research (TAR) study

Inference as designed

• **Descriptive inference**
  – Diagrams and other work products of applying the method are to be presented

• **Abduction**
  – The production of the outcome (a risk assessment) is explained by the application of the method.
  – Treatment (the method) was applied correctly: documenting the method in advance and continuously referring to it during use
  – No other mechanism that could account for outcome
  – Other explanations: competence of the researcher who applied the method

• **Generalization**
  – Scope: Large organizations; Outsourcing; Significant security department; Obligation to show that company is in control of their information assets.
Inferences actually done

• **Descriptions of the sample of OoS**
  • Some of the produced diagrams are reported
  • A few notes from primary documents and interviews are reported.

• **Explanations**
  • All products of applying the method were produced as prescribed by the method; the reason is that the researcher is the author of the method and understands exactly what needs to be done.

• **Generalizations**
  – Assumptions about context that make the scope more specific:
    • Industrial organization, very cost-sensitive environment (they often take confidentiality risks in order to save money)
    • Other users would need proper tool support
Day 1
• 9:00
  – What is a case study
  – When to use case studies
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle
• 10:30 Break
• 10:45
• Case-based inference
  – Inference design
  – Validity
• 12:15 Lunch
• 13:15
  • Exercise (Make a case-based design of your research)
• 14:45 Break
• 15:00
  – Assignments, discussion, wrapup (checklist application to two papers)
• 16:30 End of day 2
Validity

• Research setup must be valid wrt planned inferences
• Inferences actually done must be valid wrt research setup
Validity of research setup

• Inference support
  – What inferences can be supported by this research setup?

• Repeatability
  – Can other researchers repeat this research?

• Ethics
  – Does the setup satisfy ethical norms? (informed consent, no harm, fairness, confidentiality etc.)
Validity of inferences

• See chapters 12 (descriptive inference), 14 (abductive inference), 15 (analogic inference)
Day 1
• 9:00
  – **What is a case study**
  – When to use case studies
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle
• 10:30 Break
• 10:45
  – Case-based inference
• 12:15 Lunch
• 13:15
  • **Exercise (Make a case-based design of your research)**
• 14:45 Break
• 15:00
  – Assignments, discussion, wrapup (checklist application to two papers)
• 16:30 End of day 2
A case study is the study of a system in the field.

The conceptual framework of a case study should define the case architecture.

Conceptual frameworks should be assessed on their construct validity.

An architecture consists of components that have behavioral capabilities and can interact.

An architectural mechanism is an interaction among case components that produce a response from a stimulus.

Theories of a case should aim to describe, explain or predict system-level phenomena in terms of architectural mechanisms.

Knowledge questions can be answered scientifically by following the empirical cycle.

Research setup can be case-based vs sample-based, and observational vs experimental.

Observational case studies, technical action research.

Inferences in case-based research: descriptive, abductive, analogic.

Case selection strategy: Analytical induction.

Validity is the degree of support for your inference.

Checklist for empirical research.
The goal of empirical research is to develop, test or refine theories. We never start empty-handed.
Rules of the empirical research game

• **Rule of posterior knowledge**: Beliefs created by the research are present *after* execution of the research, and are absent before executing the research.
  – *E.g. formulating hypotheses after the fact, and reporting that you had them before*

• **Rule of prior ignorance**: Any beliefs present *before* doing the research may influence the outcome of the research.
  – *E.g. believing that the tested technique is better than its alternatives*

• If you violate these rules, you cheat yourself and your readers
Discussion

• Make a case-based design of your research
• Summarize on a flip-over
Day 1
• 9:00
  – What is a case study
  – When to use case studies
• 10:30 Break
• 10:45
  – Scientific theories
• 12:15 Lunch
• 13:45
  – Scientific theories
• 15:15 Break
• 15:30
  – Exercises, discussion (questions to chapters 8 and 9)
• 17:00 End of day 1

Day 2
• 9:00
  – Empirical research cycle
• 10:30 Break
• 10:45
  – Case-based inference
• 12:15 Lunch
• 13:415
  • Exercise (Make a case-based design of your research)
• 14:45 Break
• 15:00
  – Assignments, discussion, wrapup (checklist application to two papers)
• 16:30 End of day 2
Assignments

• Read the papers

• Apply the checklist for empirical research to each of them
  – Be sure to distinguish prior theories and posterior theories of the authors
  – If a piece of information is missing, say so, and indicate whether you think it should/need not have been included, and why

Deadline 7 december

3-4 Nov 2015 © Roel Wieringa

• Wieringa, R.J. (2014) *Design science methodology for information systems and software engineering*. Springer Verlag


