



## Empirical Methods and Evidence-Based Decisions in Software Engineering

Magne Jørgensen

[magnej@simula.no](mailto:magnej@simula.no)

*Material to be found at:*

[tinyurl.com/innsbruck-jorgensen](https://tinyurl.com/innsbruck-jorgensen)

## Course Assumptions and Goals

- **ASSUMPTION:** Important decisions and actions in software engineering should, as far as possible, be evidence-based, i.e., based on collection and critical evaluation of research results and practice-based experience.
- **LEARNING GOALS:**
  - Increased understanding of the importance of evidence-based practices (= good use of empirical methods for decisions and judgments)
  - Better ability to collect, evaluate and generate evidence
  - More critical attitude towards claims and better ability to evaluate argumentations

## **Agenda**

Friday: 14.00 – 18.00  
Saturday: 9.00 – 13.00 (meet 8.45 at the entrance)

## **Topics**

- Why we need more and better evidence-based practises in software engineering (EBSE).
- The steps of EBSE
- Evaluation of argumentation
- Empirical methods

[ **simula** . research laboratory ]

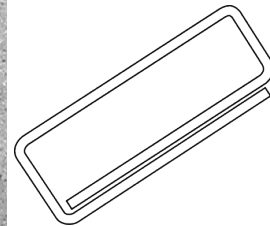
**Why do we believe in what isn't so?**

**Why do we accept and get affected by over-simplifications, non-validated claims and content-less statements?**

**A few illustrative examples**

[ **simula** . research laboratory ]

**The paper clip  
was invented by a  
Norwegian**



[ **simula** . research laboratory ]

**Short men are more  
aggressive**

**(The Napoleon  
complex)**



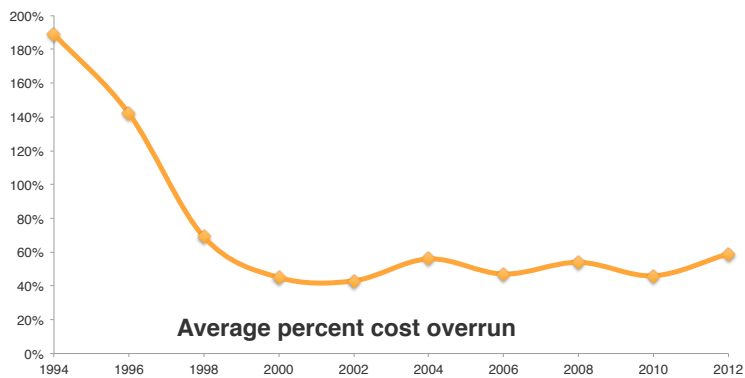
[ **simula** . research laboratory ]

**Most (93%) of our communication is non-verbal**



[ **simula** . research laboratory ]

## There were/is a software crisis



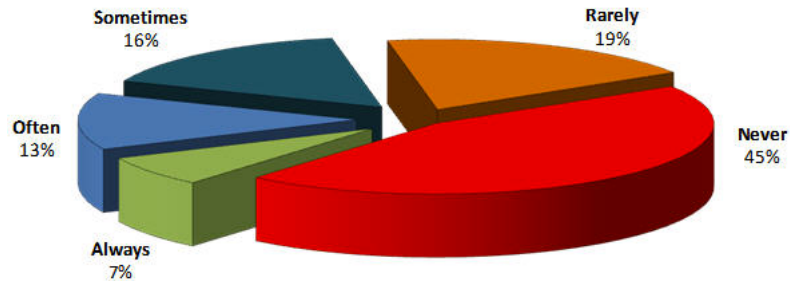
**THE  
STANDISH  
GROUP**



(page 13 of their 1994-report): *“We then called and mailed a number of confidential surveys to a random sample of top IT executives, asking them to share failure stories.”*

[ **simula** . research laboratory ]

**45% of features of “traditional projects” are never used**  
 (source: The Standish Group, XP 2002)



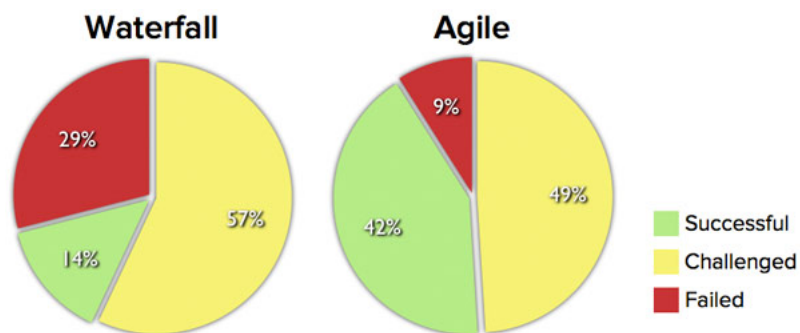
No-one seems to know (and the Standish Group does not tell) anything about this study!

Why do so many believe (and use) this non-interpretable, non-validated claim?

They benefit from it (agile community)  
 + confirmation bias (we all know at least one instance that fit the claim)

[ [simula](#) . research laboratory ]

**14% Waterfall and 42% of Agile projects are successful**  
 (source: The Standish Group, The Chaos Manifesto 2012)



Source: The CHAOS Manifesto, The Standish Group, 2012.

Successful = “On cost, on schedule and with specified functionality”  
 Can you spot a serious error of this comparison?

[ [simula](#) . research laboratory ]

## The number one in the stink parade ...

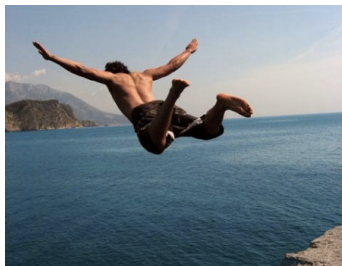


[ **simula** . research laboratory ]

## The ease of creating myths: Are risk-willing or risk-averse developers better?

Group A:

**Initially**  
Average 3.3  
**Debriefing**  
Average 2: 3.5  
**2 weeks later**  
Average 3: 3.5



Group B:

**Initially**  
Average 5.4  
**Debriefing**  
Average 2: 5.0  
**2 weeks later**  
Average 3: 4.9

Study design: Research  evidence + Self-generated argument.

**Question:** Based on your experience, do you think that risk-willing programmers are better than risk-averse programmers?

1 (totally agree) – 5 (No difference) - 10 (totally disagree)

Neutral group: Average 5.0

[ **simula** . research laboratory ]

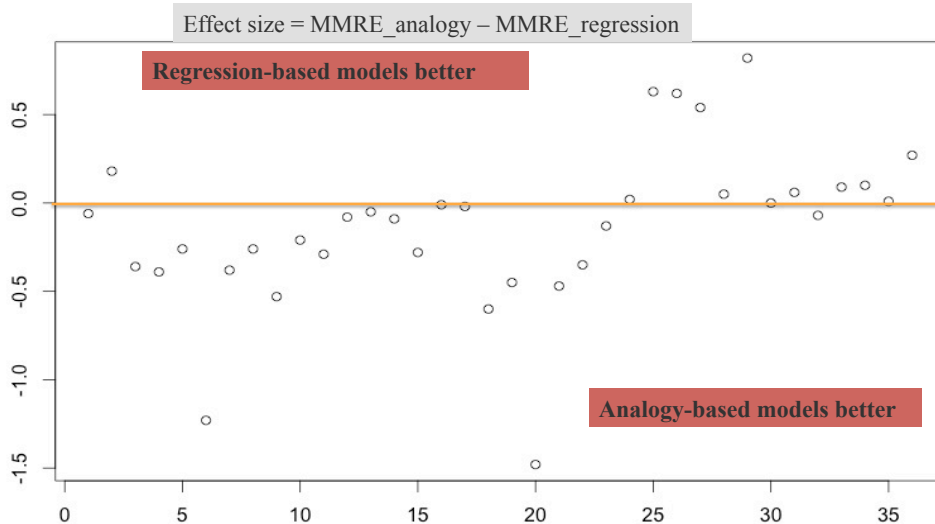
## “I see it when I believe it” vs “I believe it when I see it”

- 26 experienced software managers
- Different preferences on contract types: Fixed price or per hour
  - Clients tended to prefer fixed price, while providers were more in favor of per hour
- Presentation of a data set of 16 projects with information about contract type and project outcome (client benefits and cost-efficiency of the development work)
- Results: Chi-square of independence gives  $p=0.01$

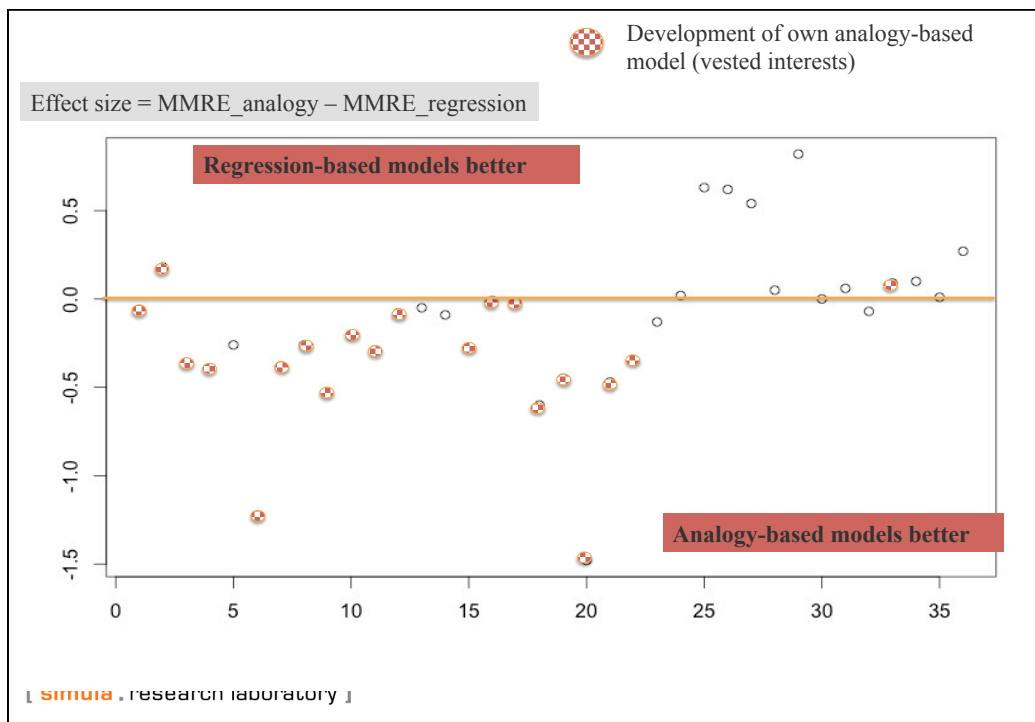
Prior preference	Project data provides support to		
	Fixed Price	Neither	Hourly
Fixed Price	7	8	0
Neutral	0	2	0
Hourly	2	3	4

[ **simula** . research laboratory ]

## Bias among researchers ...



[ **simula** . research laboratory ]



## Exercise: What does these statements mean? Are they (and can they be) evidence-based?

**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.

[ s



## Why we believe in what isn't so (1)

- Confirmation bias, i.e., we see patterns that are not there if we expect or want to see them (“we see it, when we believe it”)
- Poor studies, e.g., use of non-representative samples, researcher bias and publication bias
- Misunderstood or over-generalized research results
- Usefulness bias, i.e., we benefit from a claim being true and are for this reason less motivated check its validity properly
- Insufficient check of the validity, scope and robustness of the evidence, e.g., not reading the original study leading to the claim
- Poor precision level of claims, which makes it easier to recall confirming evidence and more difficult to falsify

[ **simula** . research laboratory ]

## Why we believe in what isn't so (2)

- A tendency towards interpreting a claim with the intention to believe rather than disbelieve it (one or two supporting recalled experiences suffice + understanding is believing)
- Desire for simple, deterministic relationship
- Belief in authorities
- Repetitions, i.e., the more frequently a claim is repeated, the more we believe in it, even when all claims are based on the same, perhaps misunderstood, source of evidence.

[ **simula** . research laboratory ]

## Evidence-based software engineering

### Learning goals of this lecture:

- Understand the goals of the course.
- Knowledge about the main steps of evidence-based decision processes.
- Introduction to the importance of critical appraisal of evidence and argumentation.

### Supporting text:

- Tore Dybå, Barbara Kitchenham and Magne Jørgensen, Evidence-based Software Engineering for Practitioners, IEEE Software, Vol. 22, No. 1, Jan-Feb 2005.

## Selection of software development methods: Fashion or evidence-based?

- **Has been fashion (traditional):** Waterfall model, sashimi model, rapid application development (RAD), unified process (UP), modified waterfall model, spiral model development, iterative and incremental development, evolutionary development (EVO), feature driven development (FDD), design to cost, 4 cycle of control (4CC) framework, rapid prototyping, timebox development, joint application development (JAD), adaptive software development, dynamic systems development method (DSDM), extreme programming (XP), pragmatic programming, test driven development (TDD), model-driven development, agile unified process, behavior driven development, code and fix, design driven development, V-model-based development, solution delivery, cleanroom development , ....
- **Current fashion (modern):** Agile development, lean development, scrum
- **Future fashion:** Elastic????

What do you think are the drivers for what is a "modern" development method?

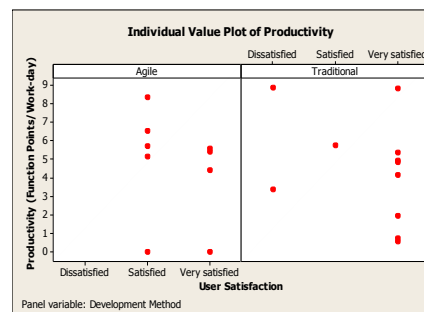
## Why Do We Need Evidence-based Practices? Are Agile Methods Better?

- **Participants:** 50 developers from a Polish company.
- **Strong belief in agile:** Before the study I collected their believes about agile methods.
  - 84% believed agile methods led to higher productivity (only 6% believed same or lower productivity), and 66% believed it led to more user satisfaction (only 8% same or lower).
- **Design of study:**
  - Generation of 10 project data sets (see example next page) with the triples: Development method (agile or traditional), Productivity (FP per work-day), and, User satisfaction (dissatisfied, satisfied, very satisfied).
  - All values were RANDOMLY generated.
  - A control gave that there were no (statistically) significant differences in the average values. The average values were slightly in favor of the traditional (non-agile) methods.
  - Each developer was randomly allocated to one of the data sets and asked to interpret it – **based on the measured data alone**.

[ **simula** . research laboratory ]

## Are Agile Methods Better?

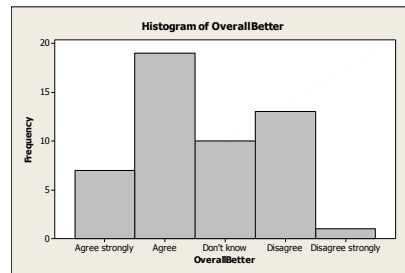
- **Instruction:**
  - “Assume that this [the data set] is the only you know about the use of agile and traditional development methods in this company and that you are asked to interpret the data. The organization would like to know what the data shows related to whether they have benefited from use of agile methods or not.”
- **Results:**
  - The interpretations of the data set related to productivity and user satisfaction as isolated variables were reasonable unbiased.
  - The interesting finding was related to the more complex interpretation of the **combined** (total) effect on productivity and user satisfaction.



[ **simula** . research laboratory ]

## Are Agile Methods Better?

- **Question:** How much do you agree in: *“Use of agile methods has caused a better performance when looking at the combination of productivity and user satisfaction.”*
- **Result:** Strong bias in favor of agile methods (see figure).
  - The agreement in the claim depended on their previous belief in agile methods.
  - Previous belief: Agile methods are better (wrt productivity and user satisfaction) → 20 of 32 agreed
  - Previous belief: Agile methods are not better (on at least one aspect) → 1 of 7 agreed
  - Previous belief: Neutral → neutral answers
- The real-life bias is probably much stronger:
  - Lack of objective measurement. More bias in favor of the preferred method.
  - More variables of importance, i.e., more complex interpretation and more space for wishful interpretation.



[ **simula** . research laboratory ]

Throw	Seq 1	Seq 2	Seq 3
1	#	o	o
2	#	#	#
3	o	o	o
4	o	#	o
5	o	#	#
6	o	o	#
7	o	o	#
8	#	o	#
9	#	o	o
10	o	#	#
11	#	o	#

### Basketball or coin?



**Seq. 1: 70% likely to keep previous.**  
*This is what most believe is the basketball player (hot hand illusion), but it is **not**.*

**Seq. 3: 70% likely to change from previous.**  
*This is what most believe is the coin, but it is **not**. It is **not** the basketball player either.*

**Seq. 2: Random sequence **and** basketball player**  
*But, does Seq. 2 look random? Too many clusters!*

## Representativeness bias (seeing patterns that are not there)

**Question:** Assume five throws with a fair coin. Which of the following sequences is more likely to occur?

**Alt. 1:** Head-Head-Head-Head-Head

**Alt. 2:** Head-Tail-Head-Head-Tail

**Answer:** Same probability

**Relevance:** We tend to use the representative heuristic (Alt 2. is more “representative” of sequence of coin flipping) and think that non-representative sequence (such as Alt. 1) are surprising patterns.

[ **simula** . research laboratory ]

## Failure of seeing true patterns

**Question:** Assume a sequence of throws with a fair coin. Which of the following two sequences is more likely to occur **FIRST**?

**Alt. 1:** Head-Head

**Alt. 2:** Tail-Head

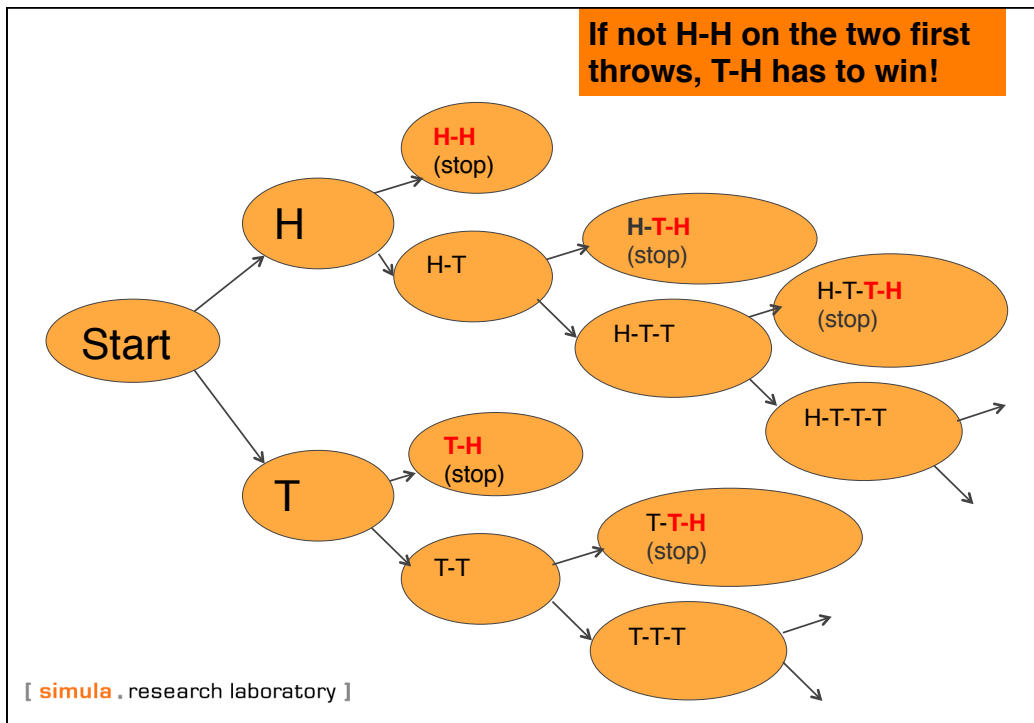
**Example:** Head-Tail-Tail-Head-Head...

→ Tail-Head occurs before Head-Head

**Answer:** It is 75% likely to first observe Tail-Head and only 25% likely to first observe Head-Head

**Relevance:** Some probabilistic connections are connected, hidden and non-intuitive. Difficult to see them ...

[ **simula** . research laboratory ]



## One more... (mainly for fun, but also to show how poor our probabilistic intuition is)

- A country has regulated that no family is allowed to have more than one son, but as many daughters as they want.
- This means that allowed sequences of child-births are:
  - Boy (stop, not allowed to hav more children)
  - Girl-Boy (stop)
  - Girl-Girl-Boy (stop)
  - etc.
- **Question:** How does this law affect the proportion of men and women in the country?
- **Answer:** Not effect at all. There will still be about 50-50 men and women

[ simula . research laboratory ]

## Why Systematic Evaluations?

- The most common decision method in software development is based on “gut feeling” (intuition, expert judgment, unconscious mental processes). This method has many strengths:
  - We believe in the outcome (frequently essential for commitment)
  - It can be very fast and inexpensive (does not require data collection)
  - It is sometimes just as good as more scientific methods (no methods are free from subjectivity and biases)
- Pure judgment (not following a systematic, scientific process) has, however, limitations:
  - We have no access to the real argumentation. (We are, however, very good at rationalizing.)
  - People are sometimes strongly impacted by “wishful thinking” and other judgmental biases, WITHOUT knowing about it.
  - Judgment-based processes are typically easy to manipulate (by sellers and gurus)
  - Important information may be missing due to lack of systematic search.
- When it is important to make the right decision, expert judgment should frequently not be the only decision method. We need systematic approaches based on scientific method.

[ **simula** . research laboratory ]

## Software professionals seem to rely very much on own and other people’s judgments

- Experiment:
  - **Subjects:** 52 software professionals
  - **Context:** Evaluation of a course in software testing.
  - **Question:** How much do you agree in the statement: “*most of the participants of this testing course will substantially increase their efficiency and quality of test work*”.
  - **Treatment:** Different types of supportive evidence.
  - **Results:** As much as 15% reported that they would emphasize a positive course evaluation of a friend who had participated in the course more than supporting evidence from an independent study conducted by scientific researchers at a well-known university. If they themselves had participated and found the course useful, as many as 80% would believe more in their own, specific experience, than in the scientific study providing aggregated information.
  - **Implication:** This experiment illustrates that even in situations where the normative response would be to use the aggregated and more objective information, many people seem to prefer the highly specific.

[ **simula** . research laboratory ]

## Do as the others ....

What do you think about these “facts”?

SAP UNITED STATES SAP.com Home

**THE FACTS SPEAK FOR THEMSELVES, AND SO DO OUR CUSTOMERS.**

When you improve ROI, lower TCO, improve efficiency, and deliver true value to the bottom line, not only will the facts and figures tell the story – so will your customers.

HENKEL GROUP

“Our sales force spends less time looking for transaction data and can devote more time maintaining our relationships with current customers and developing relationships with new customers.”  
Robert Rainier, Vice President of Operations  
INTERMEC TECHNOLOGIES CORP. | [Download PDF](#)

“(SAP) improves and accelerates decision making on all levels.”  
Michael Rauch, Director of Management Reporting  
HENKEL GROUP | [Download PDF](#)

“Users of data-based business processes are impressed by the up-to-the-minute and accurate quality of the information.”  
Peter Havelka, Project Manager  
BMW MOTOREN | [Download PDF](#)

“The benefits gained from the SAP PLM initiative.”

[ **simula** . research laboratory ]

## What is valid evidence? A real-life example (1)

- A software development department wanted to replace their old-fashioned development tool with a more modern and more efficient one.
- They visited many possible vendors, participated at numerous demonstrations, and contacted several “reference customers”. Finally, they chose a development tool. The change cost about 10-20 million NOK + training and other indirect costs.
- A couple of years after the change, the department measured the change in development efficiency (not common – most software organizations never study the effect of their choices).
- Unfortunately, the development efficiency had not improved and the new development tool was far from as good as expected.
- This illustrated that even when applying much resources and time to collect evidence, software professionals may fail in making good decisions. What went wrong in this case?

[ **simula** . research laboratory ]



## What went wrong? A real-life example (2)

- The collection and evaluation of evidence had focused on “tool functionality”, following the principle “the more functionality, the better”.
- The demonstrations focused on strengths of the tools, not on weaknesses. Although, the software professionals were aware of this, they probably failed to compensate for what the demonstrations did not demonstrate. (We are not good at identifying lacking information!)
- The reference customers had themselves invested much money in the new tool. As long as they do not plan to replace the tool, then they would however not be reference customers anymore, they will tend to defend their decisions. (Avoidance of cognitive dissonance.)
- Although the amount of information (evidence) was high, they organization lacked the most essential information (independent evaluations of the tools in context similar to their own) and processes for critical evaluation of the information.
- In addition, they lacked the awareness of how they were impacted by the tool vendors persuasion techniques.
- Guidance in the principles of evidence-based software engineering would, we think, improved the decision.

[ **simula** . research laboratory ]

## What could have been done better?

- Collection of research studies comparing the tools.
  - At that time, there were no such studies, but possibly studies on related tools.
- Less biased and more systematic use of practice-based experience.
  - They could, e.g., try to find tool customers similar to one’s own organization and use more structured and critical experience elicitation processes.
  - They should avoid that the tool vendor chose the reference customers.
- Completion of own empirical studies.
  - Invite the tool vendors to solve problems specified by the department itself at the department’s own premises.
  - Many vendors seem to accept this type of “competition”, given an important client. If not, pay them to to some work on a representative project.
- Avoid demonstrations, dinners with the tool vendors and other situations known to include more persuasion than valid information (or, at least, they should not let those who were exposed to this type of impact participate in the decision.)

[ **simula** . research laboratory ]

## A better process: Evidence-based software engineering (EBSE)

- Tore Dybå, Barbara Kitchenham and Magne Jørgensen, Evidence-based Software Engineering for Practitioners, IEEE Software, Vol. 22, No. 1, Jan-Feb 2005.
- *The main steps of EBSE are as follows:*
  - *Convert a relevant problem or need for information into an answerable question.*
  - *Search the literature and practice-based experience for the best available evidence to answer the question.*
  - *Critically appraise the evidence for its validity, impact, and applicability.*
  - *Integrate the appraised evidence with practical experience and the client's values and circumstances to make decisions about practice.*
  - *Evaluate performance in comparison with previous performance and seek ways to improve it.*

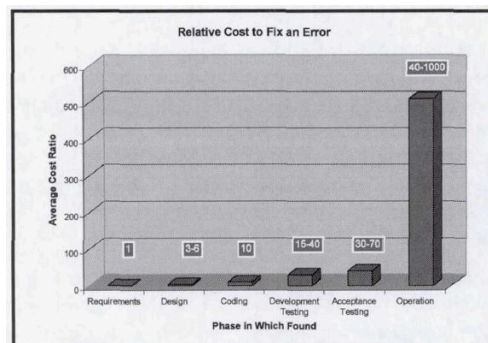
[ **simula** . research laboratory ]

## Exercise

It is often claimed that there is a strong increase in effort to correct errors the later the error is identified and corrected.

*An error that costs 1 hour to correct in the requirement phase may take about 5 hours to correct in the design, 10 hours in the programming, 50 hours in the test phase and 500 hours in the operational phase.*

- Clarify what the claim means
- What is the consequence if it is true?
- Outline good evidence-based steps for evaluation of it.
- Search for evidence using google scholar and summarize what you can learn from the evidence.



[ **simula** . research laboratory ]

## Exercise

Assume that you have to decide on whether “pair programming” is a worthwhile practice in your development team.

Outline good evidence-based steps for this decision.

[ **simula** . research laboratory ]

[ **simula** . research laboratory ]

## Argumentation Analysis

**Learning goals:** Improved ability to identify essential argumentation elements and to use this to evaluate the quality of argumentations.

**Supporting texts:**

- Alec Fisher, The logic of real arguments, Chapter 2: A general method of argument analysis. Cambridge University Press. 2004. p 15-28.
- Karyn Charles Rybacki and Donald Jay Rybacki, Advocacy and opposition, Chapter 8: What should I avoid? Pearson. 2004. p 142-163.

## Argumentation: Definitions

From “Advocacy and opposition”, by Rybacki og Rybacki:

- “**Argumentation** is a form of instrumental communication relying on reasoning and proof to influence belief or behavior through the use of spoken or written messages.”
- “**Persuasion** is an attempt to move an audience to accept or identify with a particular point of view.”

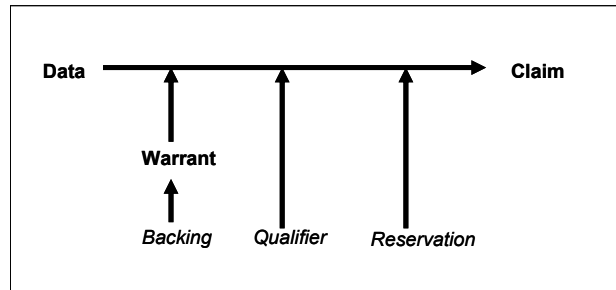
[ **simula** . research laboratory ]

## Warm-Up Exercise

- Pascal’s Wager:
  - Either there is a Christian God or there isn’t. If you believe in Him and live a Christian life, then if He exists you will enjoy eternal bliss and if He doesn’t exist you will lose very little [in comparison].
  - On the other hand, if you don’t believe in Him and don’t live a Christian life, then if He doesn’t exist you will lose nothing [and not win much in comparison to eternal bliss], but if He does exist you will suffer eternal damnation!
  - So it is rational to believe in God’s existence and live Christian life. [even if the likelihood of a God is very small].
- Intuitively most disagree with the argument, but what is wrong, if anything?

[ **simula** . research laboratory ]

## Toulmin's Model of Argumentation



- The primary elements of an argument, according to Toulmin's model, are in **bold** letters, and the secondary elements in *italic*. Toulmin's model of argumentation can be viewed as a layout of argument.
- More details in Appendix 1 of: [M. Jørgensen](#), B. Kitchenham and [T. Dybå](#). [Teaching Evidence-Based Software Engineering to University Students](#), In 11th IEEE International Software Metrics Symposium, Como, Italy, September 19-22. , 2005.

[ [simula](#) . research laboratory ]

## Toulmin's Model of Argumentation

- Start with the identification of the **claims** or conclusions made by the authors. These are normally found in the conclusion section of the papers or in the abstract, but may be found other places as well. Poor papers may, in fact, have no explicit claims at all. Evaluate the claim, e.g., whether the claim is circular or vague.
- Identify the **qualifiers**, i.e., statements about the strength of the claim, and the **reservations**, i.e., statements about the limitations of the claim. These are important when later evaluating the relevance of the evidence and the connection between evidence and claim. For example, a claim that is qualified with "this weakly indicates a cause-effect relationship" should be evaluated differently from the claim "there is a cause-effect relationship."
- Look for the **data**, i.e., the evidence supporting the claim. In particular, we ask them to evaluate the relevance of the evidence. We frequently find that the students are surprised by how little relevant evidence a lengthy software engineering paper contains.

[ [simula](#) . research laboratory ]

## Toulmin's Model of Argumentation

- Finally, we ask the students to look for the **warrant**, i.e., the supporting connection between the data and the claim. This is frequently the most difficult part of the evaluation of the argumentation, where the critical appraisal ability and analytical skill of the students is most important.
- Evaluate the degree to which the relevant data supports the claim. The warrants may have a **backing**, i.e., an argument that supports a connection of confirmation or deduction between the data and the claim. When it is not obvious that the connection between data and claim is valid (or invalid), search for elements that the authors use to support it (the backing). This may, for example, consist of analytical argumentation or evidence supporting the specific interpretation of data conducted by the authors.

[ **simula** . research laboratory ]

## Argumentation types

From "Advocacy and opposition", by Rybacki og Rybacki:

- Argumentation from cause.
  - Suggests a temporal connection between phenomena.
  - When we can document effect, we may reason as to its cause; when we can document cause, we may reason as to its effect.
  - A necessary cause is a factor that must be present to bring about an effect, but will not in and of itself produce the effect.
  - A sufficient cause includes all factors needed to produce a particular effect.
  - Control questions:
    - Is the cause capable of producing the effect?
    - Is the effect produced by the cause or does the effect occur coincidentally to the cause?
    - Are there other potential causes?
    - Has this effect consistently followed from this cause?
  - Example: Smoking increases the likelihood of lung cancer.

[ **simula** . research laboratory ]

## Argumentation types

- Argumentation from sign (indicators):
  - Connect phenomena with conditions that merely exist (correlation, prediction).
  - Tells what is the case (description), while a cause explains why it is the case.
  - Signs are observable symptoms, conditions, or marks used to prove that a certain state of affairs exist.
  - Sign reasoning is assessed on the basis of the presence of a sufficient number of signs or the certainty of an individual sign's strength

[ **simula** . research laboratory ]

## Argumentation types

- Argumentation from generalization:
  - A form of inductive reasoning in which one looks at the details of examples, specific cases, situations, and occurrences and draws inferences about the entire class they represent.
  - Should be based on a sufficiently large sample of cases.
  - Instances cited in making the generalization should be representative of all members of the group.
  - Negative (non-confirming) instances should sometimes be explained or accounted for.
  - Example: My random sample of projects in of Norwegian sw development companies shows that the average effort overrun (of all Norwegian sw companies) is about 40%.

[ **simula** . research laboratory ]

## Argumentation types

- Argument from parallel case:
  - Reason on the basis of two or more similar events or cases; because case A is known to be similar to case B in certain ways, we can appropriately draw inferences from what is known to what is unknown.
  - For the argument from parallel cases to be valid, the cases must not only be similar but their similarities must also pertain to important rather than trivial factors.
  - Example: If you liked the book X, you will probably also like the book Y. They are written by the same author and have the same “style”.

[ **simula** . research laboratory ]

## Argumentation types

- Argument from analogy:
  - Similar to “parallel case”, but related to dissimilar cases with some fundamental sameness between characteristics.
  - Considered to be the weakest type of argumentation.
  - Frequently only used rhetorically.
  - Example: Students need more structure. Students are very much like children. We all know that children need other people to structure their lives.

[ **simula** . research laboratory ]



## Argumentation types

- Argument from authority:
  - Relies on the credibility and expertise of the source.
  - Only credible within their fields of expertise.
  - Look for biases.
  - If the authority express an opinion at odds with the majority of experts in the field, the arguer should establish the credibility of that view.
  - The opinions should have a basis in facts.
  - Example: My experience [and I'm an expert in the field] is that the main problem with software projects is the lack of customer involvement.

[ **simula** . research laboratory ]

## Argumentation types

- Argument from dilemma:
  - Built with two or more arguments from cause that embody undesirable consequences.
  - Example: We need higher taxes to improve the health system. The extra burden we put on tax paying people is less negative than the suffering by those in need of better health services.

[ **simula** . research laboratory ]

## How to build a good argumentation

### Preparation phase:

- Collect relevant and valid information from many perspectives
- Have a critical distance to the validity of the information
- Try not do make up your mind before all information is collected and analyzed
- Try to avoid irrelevant and misleading information
- Understand your own biases and prejudices.

### Argumentation building phase

- Clarify the frames and context of your argumentation (define concepts, perspectives, assumptions, motivation, level of competence, goal of argumentation, ...)
- Include all relevant arguments, not only those in favor of your conclusion. The strength of the conclusion should be based on a balanced evaluation of all relevant arguments, and, known missing information.
- Focus the argumentation on the most relevant and valid evidence.
- Emphasize the logical connection between evidence and conclusion.

### Improvement phase

- Critically evaluate your argumentation and improve (play the devil's advocate)

[ **simula** . research laboratory ]

## Argumentation – What should be avoided?

- Hasty generalization
  - Example: The other day I met a group of Danish people. None of them understood what I said. I don't think Danish people are able to understand Norwegian.
- Transfer
  - Example: Bill Clinton lied about Monica Lewinsky. We can never trust what he says. Irrelevant arguments
- Circular reasoning (repeating the claim, so that it looks like an argument)
  - Example: If people exercised enough we would have no obesity. The fact that obesity is a health problem, shows that people do not exercise enough.
- Avoiding the issue
  - Example: We cannot listen to X's arguments related to speed limits. As an adult he was penalized for speeding several times.
- Forcing a dichotomy
  - Example: Should we force the children to go to bed at a time solely decided by their parents, or should we treat them as individual beings with own rights?

[ **simula** . research laboratory ]

## How to evaluate argumentations

- Be a skeptic!
- Remember that it is the argument that you are supposed to evaluate, not how much you agree with the claims.
- Start with the identification of the main claims. The claim is frequently part of an “abstract” or present in the conclusion.
- Assess the relevance of the claims for your purpose.
- Stop for a while and reflect on what evidence would convince you that the claim was true.
- Before you read the paper, assess whether it is likely that the authors have vested interests in the claims. If yes, how might this affect the results? What is the background and scope of the previous experience of the author? Is it likely that this biases the search for evidence and the conclusion?
- Read the paper with the purpose of identifying evidence that supports the claims. Skip the less relevant parts the first time you read the paper.

[ **simula** . research laboratory ]

## How to evaluate argumentations

- Evaluate the relevance and validity of the evidence. Assess whether it is opinion-based, example-based, based on a systematic review of scientific studies, etc. Is the evidence credible?
- Evaluate the connection between the evidence and the claim. Is the claim a possible, likely, or, necessary consequence?
- Check the use of measures and statistical methods. In particular, assess randomness in selection of subjects and allocation of treatment when statistical hypothesis testing is used. If not random, assess the effect of the non-randomness. [You will learn more about how to do this, later.]
- Search for manipulating elements, e.g., text that is not relevant for the argument, or loaded use of terminology used to create sympathy or antipathy. If large parts of the text are not relevant, evaluate the intended function of that part. Be aware of rhetorical elements.

[ **simula** . research laboratory ]

## How to evaluate argumentations

- Assess the degree to which the “what to avoid” is present.
- Assess whether the inclusion of evidence is one-sided or gives a wrong picture.
- Assess whether weaknesses of the study are properly discussed. If not discussed at all, why not?
- Try to identify missing evidence or missing counter-arguments. Be aware of your tendency to evaluate only what is present and forget what is not included.
- Be particularly careful with the evaluation of the argumentation if you are sympathetic to the conclusion. Our defense against "theory-loaded evaluation" and "wishful thinking" is poor and must be trained. Put in extra effort to find errors if you feel disposed to accept the conclusion in situations with weak or contradictory evidence.
- Do not dismiss an argument as having no value, if it has shortcomings. There are very few bullet-proof arguments and we frequently have to select between weak and even weaker arguments in software engineering contexts. A weak argument is frequently better than no argument at all.

[ **simula** . research laboratory ]

## Exercise

Evaluate the argumentation in the article “Aim, fire” (about the benefits of test first) by Kent Beck by using the steps outlined.

[ **simula** . research laboratory ]

## Empirical methods

- a) Scientific method
- b) Experiments
- c) Surveys

**Learning goals:** Improved ability to understand, design and evaluate research studies based on experimental methods.

**Supporting texts:**

- Briony J Oates. Researching Information Systems and Computing (Section 3: Overview of the research process, Section 7: Surveys, and Section 9: Experiments)
- Barbara Kitchenham et al., Preliminary Guidelines for Empirical Research in Software Engineering, IEEE Transactions on Software Engineering, 2002.
- [www.freeinquiry.com/intro-to-sci.html](http://www.freeinquiry.com/intro-to-sci.html)

## Science - Wikipedia

**Science** (from the Latin scientia, 'knowledge'), in the broadest sense, refers to any systematic knowledge or practice. In a more restricted sense, science refers to a system of acquiring knowledge based on the scientific method, as well as to the organized body of knowledge gained through such research. ...

## What is science?

### Important elements of science (most researchers will agree on these):

- Empirical evidence (exception for mathematics?)
- Logical reasoning
- Skeptical attitude

The following slides describes professor **Steven D. Schafersman's** viewpoints on these elements. He is a geologist, i.e., is from "natural sciences".

Many researchers will not agree with him in everything he claims. His viewpoints, however, are typical for scientists with a strong "positivistic" (more on this later) attitude and represent well the "traditional" view on science.

[ **simula** . research laboratory ]

## What is science?

[www.freeinquiry.com/intro-to-sci.html](http://www.freeinquiry.com/intro-to-sci.html)

### The Use of Empirical Evidence

- "*Empirical evidence is evidence that one can see, hear, touch, taste, or smell*; it is evidence that is susceptible to one's senses. Empirical evidence is important because it is evidence that others besides yourself can experience, and it is repeatable, so empirical evidence can be checked by yourself and others after knowledge claims are made by an individual. Empirical evidence is the *only* type of evidence that possesses these attributes and is therefore the only type used by scientists and critical thinkers to make vital decisions and reach sound conclusions."

[ **simula** . research laboratory ]

## What is science?

[www.freeinquiry.com/intro-to-sci.html](http://www.freeinquiry.com/intro-to-sci.html)

### Rationalism: The Practice of Logical Reasoning

- “Scientists and critical thinkers always use logical reasoning. *Logic allows us to reason correctly*, but it is a complex topic and not easily learned; many books are devoted to explaining how to reason correctly, and we can not go into the details here. However, I must point out that most individuals do not reason logically, because they have never learned how to do so. Logic is not an ability that humans are born with or one that will gradually develop and improve on its own, but is a skill or discipline that must be learned within a formal educational environment. Emotional thinking, hopeful thinking, and wishful thinking are much more common than logical thinking, because they are far easier and more congenial to human nature. Most individuals would rather believe something is true because they feel it is true, hope it is true, or wish it were true, rather than deny their emotions and accept that their beliefs are false.<sup>2</sup>

[ [simula](#) . research laboratory ]

## What is science?

[www.freeinquiry.com/intro-to-sci.html](http://www.freeinquiry.com/intro-to-sci.html)

### Skepticism: Possessing a Skeptical Attitude

- “The final key idea in science and critical thinking is skepticism, the *constant questioning of your beliefs and conclusions*. Good scientists and critical thinkers constantly examine the evidence, arguments, and reasons for their beliefs. Self-deception and deception of yourself by others are two of the most common human failings. Self-deception often goes unrecognized because most people deceive themselves. The only way to escape both deception by others and the far more common trait of self-deception is to repeatedly and rigorously examine your basis for holding your beliefs. You must question the truth and reliability of both the knowledge claims of others and the knowledge you already possess. One way to do this is to test your beliefs against objective reality by predicting the consequences or logical outcomes of your beliefs and the actions that follow from your beliefs. If the logical consequences of your beliefs match objective reality--as measured by empirical evidence--you can conclude that your beliefs are reliable knowledge (that is, your beliefs have a high probability of being true).”

[ [simula](#) . research laboratory ]

## Why do science?

[www.freeinquiry.com/intro-to-sci.html](http://www.freeinquiry.com/intro-to-sci.html)

“Science has unquestionably been the most successful human endeavor in the history of civilization, because it is the only method that successfully discovers and formulates reliable knowledge.

The evidence for this statement is so overwhelming that many individuals overlook exactly how modern civilization came to be (our modern civilization is based, from top to bottom, on the discoveries of science and their application, known as technology, to human purposes.).

Philosophies that claim to possess absolute or ultimate truth invariably find that they have to justify their beliefs by faith in dogma, authority, revelation, or philosophical speculation, since it is impossible to use finite human logic or natural evidence to demonstrate the existence of the absolute or ultimate in either the natural or supernatural worlds.

Scientific and critical thinking require that one reject blind faith, authority, revelation, and subjective human feelings as a basis for reliable belief and knowledge. These human cognitive methods have their place in human life, but not as the foundation for reliable knowledge.”

[ **simula** . research laboratory ]

## Research paradigms (based on the Briony Oates' text-book)

- Positivism
  - Controlled experiments, surveys, case studies, action research
- Interpretive research
  - Ethnography, case studies, action research, surveys
- Critical research
  - Action research, ethnography, case studies

**NB:** The above paradigms focus on theory building and testing. In addition, we may add “constructive research”. This type of research includes many (most?) software engineering research papers and aims at constructing products or methods scientifically.

[ **simula** . research laboratory ]

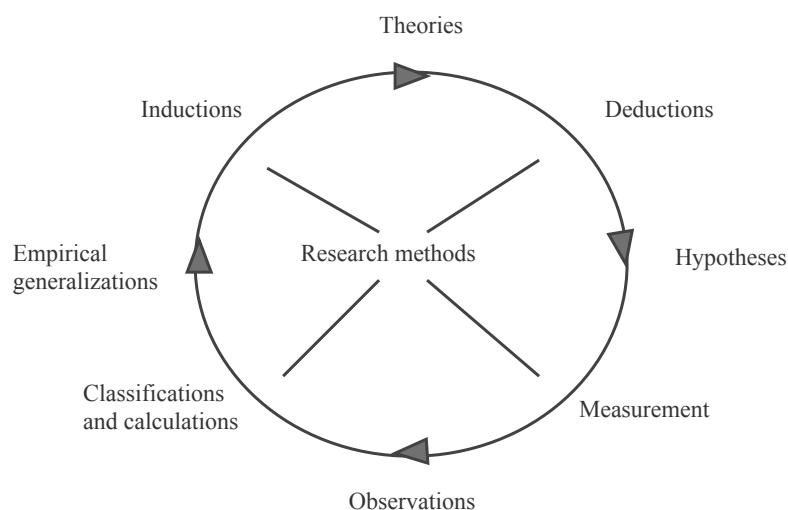


## Positivism

- Originally developed for the use in natural science, i.e., not studies of human behavior.
- Knowledge generation through Wallace's cycle (see next slide).
- Based on reductionism, repeatability and refutation (falsification, ref. Popper).
- Assumptions:
  - Our world is ordered, not random
  - We can investigate the world objectively (Well, at least achieve an acceptable degree of "inter-subjectivity".)
- Goal: Discover patterns.
- Criteria:
  - Objectivity (or at least inter-subjectivity)
  - Reliability
  - Internal validity (= the extent to which a study evaluates the intended hypotheses, i.e., that it is not likely that rival hypotheses explains the findings)
  - External validity (= the extent to which the results of a study extend beyond the limited sample used in the study)

[ **simula** . research laboratory ]

## Classical Research Process (Wallace's model)



[ **simula** . research laboratory ]

## Interpretive Research

- *“Interpretive research in IS and computing is concerned with understanding the social context of an information system: the social processes by which it is developed and construed by people and through which it influences, and is influenced by, its social setting.”* (p 292, in Briony J. Oates)
- Try to identify, explore and explain (“rich understanding”) how factors in a particular social setting are related and interdependent. Case studies are typically preferred.
- Characteristics:
  - Multiple subjective realities
  - Dynamic, socially constructed meaning
  - Researcher reflexivity (researchers should reflect on their own assumptions, beliefs and actions and their impact on the research process)
  - Study of people in their natural social setting (typically, case studies)
  - Qualitative data analysis
  - Multiple interpretations

[ **simula** . research laboratory ]

## Interpretive Research

- Criteria (somewhat forced into a positivistic framework):
  - Trustworthiness (more general concept than validity?)
  - Confirmability (analogue to objectivity - can we follow the arguments from the raw data to the interpretation?)
  - Dependability (analogue to reliability and repeatability – is the research process well documented?)
  - Credibility (analogue to internal validity – is it valid to draw the conclusions based on the data collected?)
  - Transferability (analogue to external validity – is it possible to transfer the findings to other cases?
    - **NB:** This is frequently not a goal in interpretive research. An interesting case is an interesting case, even when not transferable to other cases.

[ **simula** . research laboratory ]

## Critical Research

- “*Critical research in IS and computing is concerned with identifying power relations, conflicts and contradictions, and empowering people to eliminate them as sources of alienation and domination.*” (p. 296, in Briony J. Oates)
- Characteristics:
  - Emancipation (The goal is not only to understand, but free people from being dominated etc.)
  - Critique of tradition (It is essential to question status quo)
  - Non-performative intent (Critical to research with a focus on managers’ need for control and profit)
  - Critique of technological determination (People should be in control of technology development)
  - Reflexivity (Strong focus on own beliefs and values as researcher)

[ **simula** . research laboratory ]

## Which research method and paradigm is best?

- Wrong question! Most research methods and paradigms have their strengths and weaknesses.
- It is the relation between the research method, paradigm and research question (goal of a study) that matters.
- In practice, however, the choice of research method and paradigm is very much determined by personal preference and/or set of personal values (ideals).
  - This has the consequence that the choice of research method may be value based instead of selection of the best suited research methods.
    - Researchers belonging to “interpretive research” may not like to use statistics on people, which is essential among positivists.
    - Researchers belonging to “positivism” may not like the lack of pre-made analysis structure typical for interpretive research.

[ **simula** . research laboratory ]

## (Controlled) Experiment

- Belongs to the positivistic tradition.
- Manipulation of at least one variable, i.e., the “treatment”.
  - Example: Treatment A = Use of XP, Treatment B = Use of the Waterfall model
- Testing of hypotheses.
  - Productivity of XP is higher than productivity of Waterfall model.
  - Independent variable = Development method (XP or Waterfall)
  - Dependent variable = Productivity (“depends” on the development method)
- Strong on cause-effect relationships (mainly when treatment is randomized)
  - Without randomized treatment we have quasi-experiments where we have to argue that there are no alternative explanations.
  - Example: The developers are not randomly assigned to the use of XP or the Waterfall model. Perhaps are those using XP more motivated or more competent?

[ **simula** . research laboratory ]

## Experiment

- Typical process:
  - Hypothesis generation (e.g., derived from theory).
    - For example: Treatment A leads to higher X than treatment B.
  - Design a study where the hypothesis can be tested.
    - Study may, for example, be designed to demonstrate the existence of an effect of treatment, to examine effect size of treatment in realistic settings, or to test the robustness/generalizability of the effect of an treatment.
    - Study may be conducted in a particular context, have certain task and certain participants. These may be representative, extreme, randomly selected, or, selected by convenience.
  - Allocation of treatment to participant
    - Randomly (eases the cause-effect analysis), self-selected, ...
  - Execution of study, measurement and collection of data
  - Statistical analysis of data.
    - For example: Is the difference in effect statistically significant?
  - Interpretation of results should be done in light of previous results!

[ **simula** . research laboratory ]

## Evaluation of experiments

- Internal validity (Are there alternative explanations that can explain the results?)
  - Events other than the treatment that could have impacted the outcome?
  - Fatigue confounded the effect of the treatment?
  - Hawthorne effect occurred?
  - Measurement problems?
  - Statistical regression?
  - Biased selection of subjects, or biased allocation of subjects to treatment
  - Different loss of participants in different treatment groups

[ **simula** . research laboratory ]

## Internal validity – Exercises

- In an experiment, the effect of rewards on students' academic test results was evaluated. The hypothesis was that if the students were rewarded for good performance they would be more motivated and perform better on the next tests.
- The experiment was designed as follows:
  - Completion of Test A by 100 students.
  - The 10 best students were rewarded (given \$100) for their good performance
  - Completion of Test B by the same 100 students
- Results:
  - The 10 best students on Test A reduced, on average, their performance on Test B. The other students slightly improved their performance on Test B compared to Test A.
- Conclusion:
  - Rewards does more harm than good for students' performance.
- **Question:** Are there problems with the internal validity of this study?

[ **simula** . research laboratory ]

## Evaluation of experiments

- External validity:
  - Are the samples representative for the population of interest (the one we want to generalize to)?
    - Participants (When are students representative for software professionals?)
    - Tasks (What can we say about real world tasks based on results from smaller tasks?)
    - Contexts (What can we say about real-world effects from effects in laboratory settings?)

[ **simula** . research laboratory ]

## External validity - Exercise

- Design a study were two teaching techniques for learning OO-programming are compared.
  - A: Start early with the concept of “classes”
  - B: Learn simpler concepts, like if-then, while, .... first. Then, learn the OO-stuff.
- Formulate a sufficiently precise research question.
- Design an experiment with an acceptable level of both internal and external validity.

[ **simula** . research laboratory ]

## Famous experiments ...

- Frederik Winslow Taylor (Scientific management, Taylorisme)
  - <https://www.youtube.com/watch?v=8PdmNbqtDdl> (from 1:17)
- Hawthorn plant (ref. Hawthorne-effect)
  - <https://www.youtube.com/watch?v=lxZoxN5ljFE>
- Miligram (Obedience?)
  - <http://www.youtube.com/watch?v=yr5cjyokVUs>
- Fisher (Randomizing, p-verdier, "A lady tasting tea")
- False memories (How much can we trust witnesses)
  - [https://www.youtube.com/watch?v=PQr\\_IJvYzbA](https://www.youtube.com/watch?v=PQr_IJvYzbA)

[ **simula** . research laboratory ]

Example of use of an experiment as  
part of teaching:

**[tinyurl.com/innsbruck-exp1](https://tinyurl.com/innsbruck-exp1)**

[ **simula** . research laboratory ]

## Survey

- Data collection typically through questionnaires and/or interviews.
- Obtain the same data from a large group of people or organizations in a structured way.
- If not extensive (all IT-developers), the results should be possible to generalize.
- Frequently, a cheap and simple way to get information. Many MSc students apply this in their master thesis work.
- Easy to conduct, very difficult to conduct high quality surveys!!!!

[ **simula** . research laboratory ]

## Evaluation of surveys

To be able to generalize (with confidence) the survey should:

- Be based on a proper sampling technique
  - Random or stratified sampling are two methods to enable generalization. It is, however, common to use self-selection or convenience-based selections!
- Have a high response rate:
  - Are the non-responders different from those who responds?
- Ensure that questions are interpreted similarly by all respondents.
  - Is, for example, the term “agile” interpreted similarly?
- Ensure that misunderstandings are avoided and that the respondents have the necessary competence.

[ **simula** . research laboratory ]



## Design a Survey: Exercise

**Purpose:** What separates successful and failed IT-projects

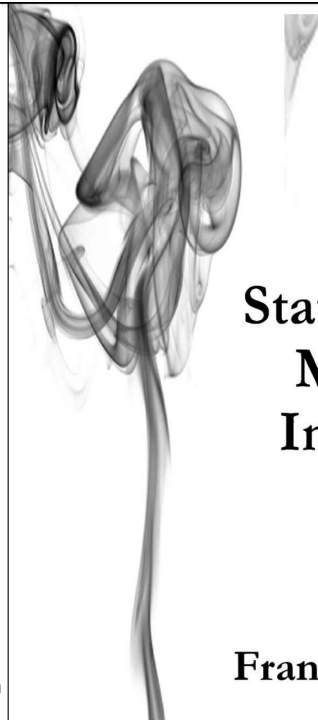
**Hypotheses:** ?? (or just exploratory)

**Sample:** ??

**Question:** ??

**Analysis:** ??

[ **simula** . research laboratory ]



## Statistics of Mental Imagery

Francis Galton

[ **simula** . research ]

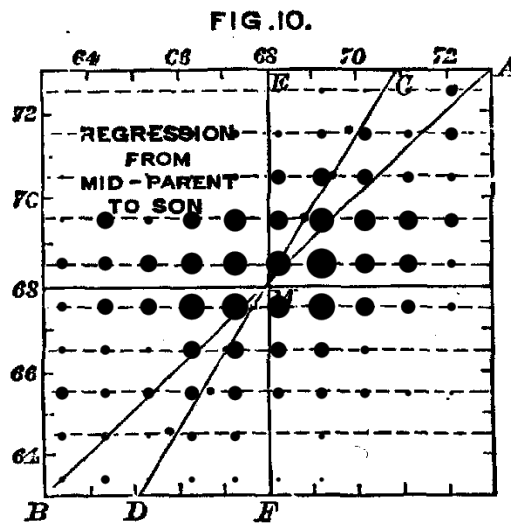
### **Galton:**

In many ways the inventor of questionnaires for scientific purposes and use of regression/ correlation type of analyses

## Sir Francis Galton's law of "filial regression to mediocrity" (one of the first scientific surveys)

Shows that children of tall parents are expected to be lower than their parents.

If this regression was a biological force, all people would soon be average!



*Natural inheritance.*  
Francis Galton,  
London,  
Macmillan and  
company. 1899.

...and how can  
(by reversing the  
regression)  
parents of tall  
children at the  
same time be  
expected to be  
lower than their  
children!

[ simula . research lab

[ simula . research laboratory ]

## Research Methods: Measurements and Statistics



**Learning goals:** Improved ability to assess the validity of software development-related measures (construct validity).

**Supporting texts:**

[www.moffitt.org/moffittapps/ccj/v4n5/article4.html](http://www.moffitt.org/moffittapps/ccj/v4n5/article4.html)

**Software quality measurement**, M. Jørgensen, *Advances in Engineering Software* 30(12):907-912, 1999.

## Introduction to Measurement Theory

- *When you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge of it is of a meager and unsatisfactory kind; it may be the beginning of knowledge, but you have scarcely, in your thoughts, advanced it to the stage of science. (Kelvin)*



- BUT, if you don't know much about it, it is not meaningful to measure it (and learn from the measurements)! So, here we have a problem.

**Exercise 1:** Which of the above two statements are more correct? If both are correct, how is measurement possible? What does this tell us about the nature of measurement?

**Exercise 2:** Why do we easily accept some measures (like the measure of length in meters), while others not (like the measure of intelligence thorough IQ-tests)?

[ **simula** . research laboratory ]

## Measurement Theory

**Def. Empirical Relational System:**  $\langle E, \{R_1 \dots R_n\} \rangle$ , where  $E$  is a set of entities and  $R_1 \dots R_n$  the set of empirical relations defined on  $E$  with respect to a given attribute.

**Def. Formal (numerical) Relational System:**  $\langle N, \{S_1 \dots S_n\} \rangle$ , where  $N$  is a set of numerals or symbols, and  $S_1 \dots S_n$  the set of numerical relations defined on  $N$ .

**Def. Measure:**  $M$  is a measure for  $\langle E, \{R_1 \dots R_n\} \rangle$  with respect to a given attribute iff:

1.  $M: E \rightarrow N$
2.  $R_i(e_1, e_2, \dots, e_k) \Leftrightarrow S_i(M(e_1), M(e_2), \dots, M(e_k))$ , for all  $i$ .

So, what does complex formalism really mean?

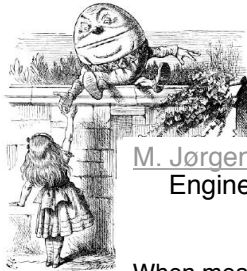
[ **simula** . research laboratory ]

## Illustration 1: Why is «meter» a meaningful measure of the height of a person?

- We have an “empirical relational system”.
  - There exists a commonly accepted understanding of the meaning of «height of a person» and of height-relations and operations, such as «person A is taller than person B».
- We have a “formal relational system”.
  - Numbers, relationships, logic, ...
- We have a mapping (function) that connect “height” and numbers so that all relationships in the “real world” are present in the “formal world”, AND, all relationships in the “formal world” are present in the “real world”.
  - For example (*A,B,C,D are persons*) and *h* our measure of height:
    - $A$  is taller than  $B$  in the real world  $\Rightarrow h(A) = 1.92 \text{ meter} > h(B) = 1.80 \text{ meter}$
    - $h(C) = 1.88 \text{ meter} > h(D) = 1.87 \text{ meter} \Rightarrow C$  is taller than  $D$  in the real world.
- In addition, have acceptable methods for the measurement process!

[ **simula** . research laboratory ]

## Illustration 2: Measurement of software quality



M. Jørgensen. Software quality measurement, Advances in Engineering Software 30(12):907-912, 1999.

When measuring complex phenomena like software quality we frequently have to choose between two evils:

- Use of a definition of software quality close to people’s intuition of what software quality is (e.g., “how well software meet the software development stakeholders needs”), which is good for communication purposes, but impossible to measure.
- Use of a definition that enables measurement of software quality (e.g., “errors per lines of code”), but only partly connected to the way the term software quality is used.

[ **simula** . research laboratory ]

## Exercise

- Assume that:
  - The management of an organization wants to know whether an introduced process change (e.g., change of development method) have had a positive effect on software maintainability (one possible aspect of software quality) or not.
  - You are the person in charge of measurement of this!
- How would you proceed?

[ **simula** . research laboratory ]

[ **simula** . research laboratory ]

## Research Methods: Qualitative studies, Case studies

Magne Jørgensen  
*magnej@simula.no*

## Introduction

### Learning goals of this lecture:

- Understand the principles of qualitative research.
- Better ability to evaluate the validity of results from qualitative studies.

### Recommended reading:

- Briony J Oates, *Researching information systems and computing*, SAGE Publications.
- David Silverman, *Interpreting Qualitative Data*, SAGE Publications
- Cynthia K Russell and David M Gregory, *Evaluation of qualitative research studies*, *Evid. Based Nurs.* 2003;6;36-40

## Quantitative vs qualitative research

- Possible differences:
  - Statistical analysis vs interviews and text analyses?
  - Experiments vs case studies?
  - Positivistic vs interpretivistic?
  - "Natural science" vs "Social science"?
- The subject of study should decide the selection of research method, e.g.
  - a study of the connection between education and salary levels may be hard to carry out without measurements.
  - a study of how people perceive the power structure at universities may be hard to carry out without talking to people and analysing their answers and/or observing behavior.
- There are no good or bad research methods, only good and bad research and different degrees of fit between research method and research problem.

## Main types of qualitative research methods

- Observation
  - Example: Observation of people's behavior at meetings
- Analysing texts and documents
  - Example: Analysing project experience reports and minutes from status meetings
- Interviews
  - Example: Interviews with experienced managers about the reasons of estimation errors.
- Recording and transcribing
  - Example: Videorecording of team work. Categorizing the communication according to types of statements.

[ **simula** . research laboratory ]

## Example: Observation of Effort Estimation in Teams

- **Research Problem:** What is the processes used when estimating effort in teams.
- **Setting:** Seven teams from the same company estimated the effort of the same software application in a close to 100% realistic setting.
- **Quantitative research methods applied:**
  - Video recording of the team discussions
  - Repeated observation of the teams' verbal and non-verbal communication
  - Transcription of the discussions (with information about non-verbal communication)
  - Coding of the discussion elements
  - Combining quantitative (the estimates) and qualitative information
  - Interviews with all the participants after the team estimation process was completed

[ **simula** . research laboratory ]

## An example of a transcribed dialogue

Verbal communication	Description of actions
<p>1. <b>DB:</b> The first thing we have to do at any rate is that we have to implement what it says here, then we'll see if it's right (). And of course that takes some time, so we have a one-off job here. We'll convert what we have from Oracle to an SQL server. I don't understand why they are going to do this, but it is (5 sec). The challenge here is that. It says at the back here, that they don't have, they don't have any Oracle installations themselves. So the question is whether we can assume that they have access to an existing Oracle installation so that we can get it over, access the database directly, or whether we have to get it on files and define the file format.</p>	DB looks at the process flow diagram in the requirement specification. PM is picking up a pen, looking in DB's requirement specification. D looks in her requirement specification.
<p>2. <b>D:</b> Of course, there's not supposed to be any online interface at all.</p>	
<p>3. <b>DB:</b> No, and then we have to</p>	
<p>4. <b>D:</b> Everything will just go on files.</p>	
<p>5. <b>PM:</b> I see it also quite clearly as the transfer of historic data. Then () Someone whether or not it is a part of this, I am not quite sure at the moment, which then extracts it from the Oracle, but of course the most important part of the job here is to get the data put into the database. So that you. The format is different, so you can't just plop it in.</p>	PM looks in the requirement specification while talking.
<p>6. <b>DB:</b> Yeah, yeah, because it says that they are different formats and that it is a one-off job. It's something you do just once. This is a bit difficult since I don't know how much data is supposed to be transferred. It doesn't say very much about it.</p>	
<p>7. <b>PM:</b> In my opinion, it needs to be interpreted based on that we have defined, or we are now defining the database we want. Then we will need some data, and we have to take that from the old one, which we will actually have to use as a starting point. What we want to include and assume will be available in one form or another in the old one.</p>	All three team members look in the requirement specification.
<p>8. <b>D and DB:</b> mm</p>	
<p>9. <b>DB:</b> As I interpret the text here, there is at least wholesale data that we are retrieving from there. () Getting a comma separated file is not a big job. If we, that is, if we assume they can retrieve it for us, and if not, we will need people who know both Oracle and Outsider. No, excuse me, SQL Server. () Because then you need to be able to export it from there, and it must be possible to make an import program in the other database.</p>	DB looks in the requirement specification while talking. PM starts taking notes.

[ **simula** . research laboratory ]

## Excerpt of the analysis of the transcribed analysis ...

- What happens in this interplay between elaborations and clarifications is that assumptions drive the interaction forward and creates possibilities to pinpoint and narrow down the elaborations that takes place. This narrowing down is achieved through an articulation of the main challenges which close the elaboration and establishes a common ground of understanding, making it possible for the team to continue in their work. This is what happens with the project manager's utterance in line 5.
- The articulation also leads to another aspect of the estimation discussion, namely planning. Planning the development of the software system is what activates the future-oriented dimension in the estimation discussion, which is of vital importance for assigning a number of work hours needed for the development. Moreover, the articulation creates the connection between the two aspects, problem solving and planning, making switching between them possible.

*<excerpt from a paper by Kristin Børte and Monika Nerland>*

[ **simula** . research laboratory ]



## The coding of the team discussion elements

Team 1	Team 2	Team 3	Team 4	Team 5	Team 6	Team 7
Process	Process	Process	Process	Process	Process	Process
Understand	Understand	Understand	Understand	EstimTot	Understand	Understand
Process	Process	EstimAna	EstimAna	Process	EstimProg	Process
Understand	EstimProg	Process	EstimDes	Understand	Understand	Understand
Process	Understand	EstimProg	EstimProg	EstimAna	EstimProg	EstimProg
Understand	EstimProg	Understand	EstimOther	Understand	Understand	Understand
EstimDes	U n d e r s t a n d	SearchPart	EstimTot	EstimDes	EstimProg	EstimProg
EstimProg	EstimProg	Understand		EstimProg	Understand	EstimTot
Understand	U n d e r s t a n d	EstimProg		EstimOther	EstimProg	Process
EstimProg	EstimProg	Understand		EstimTot	Understand	EstimTot
Process	U n d e r s t a n d	EstimProg		EstimOther	EstimProg	
Understand	EstimProg	Process		Understand	Understand	
EstimProg	Understand	EstimProg		EstimOther	EstimProg	
Understand	Process	EstimOther		EstimTot	Understand	
EstimProg	EstimDesign	EstimDes			EstimProg	
Process	Understand	EstimAna			EstimTot	
EstimProg	EstimProg	EstimOther				
EstimTot	U n d e r s t a n d	EstimTot				
	E s t i m P r o g					
	Understand					
	EstimProg					
	EstimOther					
	EstimTot					

[ simula . research laboratory ]

## Examples of results

- There seems to be two types of estimation processes in use:
  - Sequence
  - Inside-out
- There was a surprising lack of reference to previous experience
  - Instead there was a sort of negotiation between beliefs.
- One of the team started with an "estimation anchor", which seems to have made their estimate less accurate than the other teams' estimates.
- One of the teams ended up agreeing on an effort estimate that all team members thought were too low.

[ simula . research laboratory ]

## Case study: Selection of cases

- Some good strategies for selecting cases:
  - Random selection (to avoid systematic biases)
  - Stratified selection (to ensure representativeness)
  - Typical cases
  - Extreme or exceptional cases
  - Maximum variation cases
  - Maximum generalizability cases (if it's the case here, it will be the case in many other cases)
  - Falsifying/critical cases (if it's not the case here, the theory is strongly weakened)
  - Educational cases
- The typical strategies (which are not that good): Convenience cases and confirming cases

[ **simula** . research laboratory ]

## Example: Case study

- Embedded knowledge and offshore software development, by Brian Nicholson and Sundeep Sahay.
- Longitudinal and interpretive case study methodology conducted during 1998–2000.
- *“An interpretive approach assumes that the knowledge of reality is gained only through social constructions such as the use of language, attitudes and shared meanings of actors, structure and form of documents, and the use of tools, technologies and other artefacts (Klein & Myers, 1999). Interpretive research does not predefine independent and dependent variables and determine causal relationships between them. Instead, the aim is to understand the complexity of human sense making processes, and the processes by which inter-subjectivity is obtained as the situation is constantly changing. An implication of this interpretive perspective in our research was that our aim was not to try to correlate the problems of knowledge with the success or failure of a global software development relationship. Instead, the aim was to provide insights into the processes contributing to the complexity of embedded knowledge in offshore software development settings, and the contextual conditions that contribute to this complexity.”*

[ **simula** . research laboratory ]

## Example: Case study

- Motivation:
  - “While migratory knowledge resides in “mobile packages” such as books, formulas and machines, embedded knowledge tends to be non-migratory and “resides primarily in specialised relationships among individuals and groups and in the particular norms, attitudes, information flows and ways of making decisions that shape their dealings with each other”. Thus, knowledge residing in organising principles, routines and standard operating procedures may be non-migratory due to embeddedness of knowledge in context.”
- Research questions:
  - What is the nature of embedded knowledge in offshore development?
  - How do individuals, teams, and organisations manage this complex problem of embedded knowledge in offshore development?

[ **simula** . research laboratory ]

## Example: Case study

- Information sources: Semi-structured interviews, observations and document analysis.
- Research process: Evolutionary
  - Observe, develop concepts/theory, observe more, adjust concepts/theory, ....
- Example of result:
  - Demonstrations on how knowledge embedded in one culture may lead to communication problems. [and, after 2 years, to a shut down of the company's subsidiary in India.]
  - *“The case emphasizes that outsourcing is not merely about managing the economics, but also developing cultural sensitivity and empathy.”*
    - Informally: It may not be a good idea to create “little England” in India.

[ **simula** . research laboratory ]

## Evaluation of quantitative research

### *Control questions*

#### **What are the findings?**

1. Is the description of findings thorough?

#### **Are the findings valid?**

1. Is the research question clear and adequately substantiated?
2. Is the design (the research method) appropriate for the research question?
3. Was the method of sampling (e.g., case selection) appropriate for the research question and design?
4. Were data collected and managed systematically?
5. Were the data analysed appropriately?

[ **simula** . research laboratory ]

## Evaluation of quantitative research

#### **How can I apply the findings?**

1. What meaning and relevance does the study have for my practice?
2. Does the study help me understand the context of my practice?
3. Does the study enhance my knowledge about my practice?

[ **simula** . research laboratory ]

## Collection and Evaluation of Practice-based Evidence

**Learning goals of this lecture:**

- Better ability in identifying, collecting and evaluating relevant practice-based evidence

**Recommended reading:**

- <http://web.cs.wpi.edu/~jburge/thesis/kematrix.html>
- Reasons for Software Effort Estimation Error: Impact of Respondent Role, Information Collection Approach, and Data Analysis Method, Magne Jørgensen & Dag Sjøberg

## Experience vs expertise and skill

- *“Yet in nearly every study of experts carried out within the judgement and decision-making approach, experience has been shown to be unrelated to the empirical accuracy of expert judgements”* (Hammond 1996, p. 278).
- The amount of “deliberate practice”, i.e., activities especially designed to improve specific aspects of an individual’s performance seems to be more closely related to skill than amount of experience (Ericsson, Krampe et al. 1993).

*“What we learn from history is that people don’t learn from history.”*  
(George Bernard Shaw)

[ **simula** . research laboratory ]

## Experience vs expertise and skill

- The reasons why the quality of professionals’ judgements may not improve much through experience are according to (Brehmer 1980):
  - We try to confirm theories, rather than reject incorrect hypotheses.
  - The fact that we are able to find a rule is sufficient to believe that we have a valid rule even though we have no experience indicating that the rule is valid. In other words, the confidence in own knowledge increases with the ability to find rules regardless of the validation of these rules.
  - In cases where we act on the experience based judgement there will be a number of additional factors that prevent us from detecting that our judgement is incorrect, e.g. self-fulfilling prophecies.
  - We tend to prefer deterministic rules even if the relationships between variables are probabilistic. If we find no deterministic rules, we tend to assume that there is no rule at all and start guessing.

[ **simula** . research laboratory ]

## Exercise

- Studies repeatedly shows that the actively managed mutual funds are not more profitable than their reference index (see for example [www.ub.uit.no/munin/bitstream/10037/2118/1/thesis.pdf](http://www.ub.uit.no/munin/bitstream/10037/2118/1/thesis.pdf)).
  - In addition, there is no (or a slightly negative) correlation between previous and future performance. This means that there are no persistence in the performance, either. Otherwise, we could select only the "good" funds.
  - The model explaining the performance best is a pure "by chance" model (random walk).
- Why, do you think, so many people do not see this and instead follow more profitable ways of investing their money, e.g., by buying so-called index funds?
  - In other words: Why do most people think the "experts" managing the mutual funds have the skill we should be looking for, i.e., better predictions of the stock market than the reference index, when the reality is that they clearly don't have it?

[ **simula** . research laboratory ]

## Learning problem 1: We see what we expect to see



[ **simula** . research laboratory ]

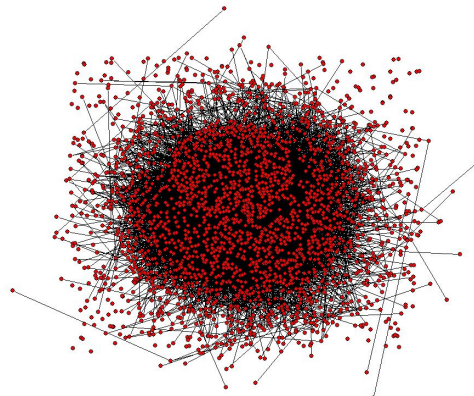
## Learning problem 2: “We won” - “they lost”

- We sincerely believe that we succeeded because we are skilled and failed because we had bad luck.
- The need for a high level of self-esteem makes learning sometimes difficult.
- Example:
  - Software developers systematically point at reasons outside their control to explain failures, and reasons the control as reasons for success.

[ **simula** . research laboratory ]

## Learning problem 3: Lack of the total picture

- **Local interpretation:** In a company, most project leaders agreed on that the most important reason for overruns was lack of clear and precise requirements.
- An analysis of the projects suggested the opposite. The advantage of vague requirements (increase of flexibility) was larger than the disadvantage of the lack of clarity.
- Exercise: Why didn't the project leaders discover this?



[ **simula** . research laboratory ]



## Learning problem 4: Superficial Learning

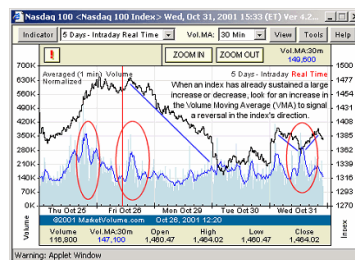
- Most people stop when they have believed they have found the direct causes, and do not look for indirect and contributory reasons.
  - A reason for problem failure is, for example, frequently "unexpected events".
  - BUT, unexpected events are quite common and should not be unexpected.
  - The important cause may be why they weren't sufficiently prepared for unexpected events.
- Children are in many ways good learning examples for deeper learning.



[ simula . research laboratory ]

## Learning problem 5: We see patterns where there are none

- HOT HAND?
  - "Basketball players and fans alike tend to believe that a player's chance of hitting a shot are greater following a hit than following a miss on the previous shot. However, detailed analyses of the shooting records of [reference to several studies and a controlled shooting experiment] provided no evidence for a positive correlation between the outcomes of successive shots." (Gilovich, COGNITIVE PSYCHOLOGY 17, 295-314, 1985)
- Frequently the same problem in IT-projects. If B follows A two times in a row, we have a rule.
- Stock market analysis is heavily based on finding patterns where there are none.



[ simula . research laboratory ]

## Learning problem 6: Hindsight bias

- In a survey we gave the software professionals real and invented project outcomes. Regardless of the version they received, most of them thought that the outcomes were as expected.
- We do this, even when we (at least on behalf of others) are aware of the hindsight bias effect



[ **simula** . research laboratory ]

## Learning problem 7: Falsification

- Several studies show that we tend to confirm what we believe and are very poor at looking for and emphasizing non-conforming evidence.
- The consequence is that we may end up believing strongly in incorrect or strongly uncertain knowledge.



[ **simula** . research laboratory ]

## Learning problem 8: A strong focus on learning may make things worse

- In particular, when the desire is not connected with the opportunities to learn
  - F. I. Steele: Organizational overlearning, Journal of Management Studies, 1971.
- Example: Governmental reports on the reasons for failed, mega-large IT-projects.
  - Interpretations based on highly incomplete argumentation
  - The causal chain is clearly too simplistic. There are, for example, many cases where the same chain led to success.
- Paradox: The learning itself frequently makes the learning less relevant.

[ **simula** . research laboratory ]

## Results from a study

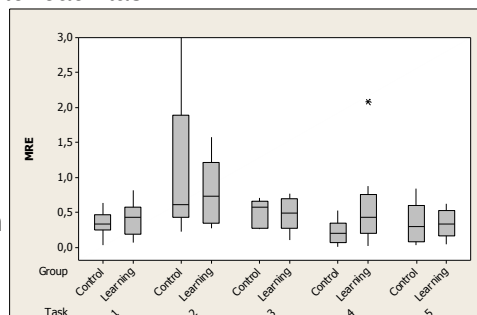
### Design:

- 20 experienced software developers, randomly allocated a learning and a control group
- All of them estimated and complete the same five development tasks
- Those in the learning group, but not those in the control group were instructed to spend at least 30 minutes on the identification, analysis and summary of experience and learning after each task

### Results:

- Those in the learning group did not improve the estimation accuracy, and were more over-confident in the estimation accuracy. This may have been due over-estimation of how much they had learned.

[ **simula** . research laboratory ]



## **OK, it's difficult to learn from experience. BUT, how should we collect reliable knowledge?**

**Guidelines:** Check relevance, combine perspectives, triangulate of methods, be critical, design processes that go for the deeper cause-effect relationships

- Check the relevance of the experience. Remember that:
  1. Relevance of knowledge and skill can be very narrow.
  2. Experience is not the same as knowledge. Preferably, to transfer from experience to relevant and reliable knowledge, the following conditions should be met by the persons's learning situation:
    - Learning-friendly conditions. Preferable situations where only few changes takes place and there are systematic effect measurement in place.
    - Unbiased interpretations. A person responsible for selecting a new tool is, as an illustration, not the best one to assess it's impact on quality and productivity.

[ **simula** . research laboratory ]

## **How should we collect reliable knowledge?**

- If unbiased, complete pictures from one person is difficult, try to collect information from more than one perspective, background and role.
  - Preferably, the informants should have formed their knowledge independent of each other.
- Example of knowledge collection technique:
  - Observations of on-the-job work
  - Interviews
  - Observations in controlled contexts with verbal protocols (thinking-aloud)
  - Study of written material (emails, experience reports, etc.)
  - Statistical modeling
  - Concept mapping
  - Sessions of analysis of cause-effects (Root Cause Analysis, Ishikawa, Post Mortem Analyses, ...)

[ **simula** . research laboratory ]

## Types of cause (X) – effect (Y) relationships

- There is a *direct* causal link between X and the Y, i.e., X is a *direct reason* for Y.
- X leads to events that, in turn, lead to Y, i.e., X is an *indirect reason* for Y. If the events leading to Y started with X, we may call X the *root reason* or the *trigger reason*.
- The events actually leading to Y would have been harmless if X had not been present, i.e., X is an important *contributory reason*, or *necessary condition* for Y.
- The strength of Y always increases when X is present, i.e., X is a *deterministic* reason.
- The presence of X increases the probability of Y, i.e., X is a *probabilistic* reason.
- Mainly the very high (or low) Y values are caused by X, i.e., X is mainly a *large effect* reason.

[ **simula** . research laboratory ]

## An example of data collection triangulation

- **Study:** Reasons for Software Effort Estimation Error: Impact of Respondent Role, Information Collection Approach, and Data Analysis Method
- **Motivation:** How to collect practice-related experience that can enable reduced estimation error
- Experience collection methods:
  - Semi-structured interviews with employees in different roles
  - Examination of 68 written experience reports
  - Statistical analysis

[ **simula** . research laboratory ]

TABLE 1  
Questionnaire-Based Studies on Reasons for Software Estimation Error

Study	Population	Study Design	Results
Phan et al. [2]	Software professionals (80% of them were project managers or developers) in 191 organizations.	Four pre-defined categories: Long duration, over-optimism, poor analysis and design, and frequent changes.	The two most important reasons were "unrealistic over-optimism" and "frequent changes".
Van Genuchten [3]	Project managers responsible for the estimation of 160 activities in six development projects within one department.	Pre-defined classification of reasons for error. The six project managers marked one (or more) of these for each activity.	Most frequent reasons were "more time spent on other work than planned" and "complexity of application underestimated".
Lederer and Prasad [4]	Estimation responsible (mainly project managers and developers) personnel in 112 organizations.	Pre-defined list of reasons where general importance for estimation error was marked with a value from 1 to 5.	Most important reasons were "frequent requests for changes by users", "users lack of understanding of their own requirements", and "overlooked tasks".
Standish Group - 1994 <sup>3</sup>	"IT executive managers" (mainly project managers?) from 365 organizations.	Pre-defined classification of reasons.	The three most important reasons for estimation overruns were "lack of user input", "incomplete requirements and specifications", and, "changing requirements and specifications".
Subramanian and Breslawski [5]	Project managers in different companies representing 45 projects.	Reasons classified by the authors based on responses from the project managers.	Most important reasons were "requirement change/addition/deletion", "programmer or team member experience, turnover", and, "design changes, scope, complexity".

Previous studies show a strong tendency to emphasize direct reasons and reasons outside one's own control.

[ **simula** . research laboratory ]

## The personnel interviewed

- The manager of the technical personnel (M-Tech).
- The manager of the human-computer-interaction personnel (M-HCI).
- The manager of the graphic design personnel (M-Graph).
- The most senior project manager (PM-Sen). This project manager was frequently used to review other project managers' estimates.
- Two project managers with technical background (PM-Tech1 and PM-Tech2).
- A project manager with human computer interaction background (PM-HCI).
- A project manager with graphic design background (PM-Graph).

[ **simula** . research laboratory ]

## The interviews

Results:

- The responses depended very much on the reasons provided
- General managers provided more general reasons.
- Little critique of own role, e.g., the project managers did not think their project management ability was a problem, while the general managers thought this.
- Only one respondent mentioned "contributory reasons".
- The chain of reasons were not well explained and mainly based on beliefs.
- All reasons were described deterministically, in spite of that a probabilistic description would have been more correct in most contexts.

[ **simula** . research laboratory ]

Interviews are well suited to get access to indirect reasons, but may need special attention to get to the deep-level causes.

Subject	Reasons
M-Tech ( <i>Manager of the software developers</i> )	No systematic feedback to enable learning (→→). Insufficient time on estimation and planning (→→), leads to overlooked tasks (→).
M-HCI ( <i>Manager of the HCI personnel</i> )	Lack of processes enabling learning from experience (→→). Insufficient focus on HCI in the estimation process (→→). Lack of client realism in HCI-requirements (→→). Poor project planning (→→). Poor project management (→→).
M-Graph ( <i>Manager of the graphical designer personnel</i> )	Project managers are not skilled in planning multi-disciplinary projects (→→), which leads to insufficient focus on graphic design in the estimation process (→→), and inefficient allocation and use of graphic design resources (→). No systematic feedback to enable learning (→→). Insufficient tool support for project management (→→). Poor project management (→→). Customer requirements difficult to interpret (→→).
PM-Sen ( <i>Senior project manager with extensive experience from project bidding and planning</i> )	Insufficient focus on the project manager role (→→), leads to insufficient training and feedback (→→). Insufficient standardization of planning and development processes (→→). The experience database of previous projects is not used (→→). Inefficient allocation of project resources (→→).
PM-Tech1 ( <i>Project manager with technical background</i> )	Clients unable to deliver a good requirement specification (→→), leads to unplanned re-work (→). Lack of requirement change control processes (→→). Insufficient time spent on estimation and planning (→→). Not sufficient focus on learning from experience (→→).
PM-Tech2 ( <i>Project manager with technical background</i> )	Projects are frequently different from earlier projects (→→), leads to lack of relevant experience when estimating (→), because of lack of checklists (→) and experience database (→). Incomplete requirement specifications (→→).
PM-HCI ( <i>Project manager with HCI background</i> )	HCI is involved too late (→→), which leads to unrealistic expectations by clients (→→), and unplanned activities (→). Project manager has insufficient knowledge about HCI (→→). Not sufficient focus on learning from experience (→→).
PM-Graph <sup>3</sup> ( <i>Project manager with graphic designer background</i> )	Insufficient focus on graphic design in the estimation process (→→). No systematic feedback to enable learning (→→). Estimate strongly impacted by price-to-win (→→). Lack of justification of estimates (→→).

[ **simula** . research laboratory ]

## The Experience Reports

- Experience reports from 68 projects/tasks
- Classification scheme for the reasons for accurate and inaccurate estimates
- Includes measures of estimation accuracy per project/task

[ **simula** . research laboratory ]

Id.	Reason	Reported in Project	Mean MRE	Mean RE	Proportion of Over Median Large Projects
1	Unexpected events and overlooked tasks (→)	5, 8, 10, 11, 15, 21, 25, 26, 30, 31, 35, 43, 47, 49, 50, 51, 52, 58, 60, 61, 62, 63, 64, 65, and, 66	0.32	0.32	60%
2	Change requests from clients or "functionality creep" (→)	5, 7, 9, 14, 15, 16, 18, 22, 23, 31, 47, 48, 61, and, 67	0.35	0.32	71%
3	Simpler task or more skilled developer than expected (→)	13, 34, 36, 42, 57, and, 59	0.54	-0.54	17%
4	Resource allocation problem (→→)	8, 28, 43, and, 47	0.32	0.32	50%
5	Poor requirement specification or problems with communication with the client (→→)	4, 8, 18, 22, 25, 26, 31, 43, 44, 45, 48, 54, 59, 63, and, 67	0.42	-0.26	73%
6	Too little effort on estimation work (→→)	63	0.70	0.70	100%
7	High priority on quality, cost accuracy not of high importance (→→)	17, 18, 22, and, 30	0.32	0.33	75%
8	More reuse than expected from other projects (→)	4, and, 57	0.61	-0.61	50%

[ **simula** . research ]



## Experience reports

- Mainly direct reasons were reported.
- Success was described as due to the respondents' own skill and choices, failures were attributed events outside their control.
- Some obvious reasons were not reported, e.g., reasons related to the "political estimation games".
- A more structured process for experience reporting may have led to more reliable reports.

[ **simula** . research laboratory ]

## Statistical analysis

- $MRE = 0,14 + 0,13 \text{ Company Role} + 0,13 \text{ Participation} + 0,13 \text{ Client Priority}$ ,  
( $p=0.03$ ) ( $p=0.08$ ) ( $p=0.07$ ) ( $p=0.09$ )
- $RE = 0,12 - 0,29 \text{ Company Role} + 0,27 \text{ Previous Accuracy}$   
( $p=0.05$ ) ( $p=0.004$ ) ( $p=0.01$ )
  - Company Role: The project was estimated by a software developer = 1. The project was estimated by a project manager = 0.
  - Participation: The estimator estimated the work of others = 1. The estimator participated in the estimated project = 0.
  - Client Priority: The client prioritized time-to-delivery= 1. The client had other project priorities than time-to-delivery, i.e., cost or quality = 0.
  - Previous Accuracy: The estimator believed that he/she had estimated similar tasks with an average error of 20 percent or more = 1; less than 20 percent error = 0.

[ **simula** . research laboratory ]

## The Results Summarized

<b>Interviews</b>	<b>Experience Reports</b>	<b>Statistical analysis of MRE</b>
No systematic feedback to enable learning	Unexpected events and overlooked tasks	Project estimated by a software developer (as opposed to a project manager)
Poor project planning and management	Change requests from clients or “functionality creep”	Project estimated by a person not participating in the project
Poor requirement specification	Simpler task or more skilled developer than expected (reason for effort under-run)	Client prioritizes time-to-delivery, not cost or quality

- Different respondents and collection methods lead to different results.

[ **simula** . research laboratory ]

## Exercise

- Assume that your task is to analyse whether your company should introduce pair-programming. You know a couple of other companies that have used pair-programming and want to interview them about their experience, i.e., you want to get practice-based evidence about pair-programming relevant for you own company.
- Outline the design of the interview? (including preparation, selection of respondents, questions and request for other material that could be used to quality assure the interview-based responses – method triangulation)

[ **simula** . research laboratory ]

## Review and Synthesis of Evidence

### Review and synthesis

*Review* - the process of bringing together a body of evidence from different sources

*Systematic review*: a review which tries to adhere to a set of 'scientific' methods to limit error (bias) mainly by attempting to locate, appraise and synthesize (attempt to reconcile) all relevant evidence (from research or more widely) to answer a particular question(s)

*Synthesis* - stage of a review in which evidence extracted from different sources is *compared* to identify patterns & direction in the findings, or *integrated* to produce an overarching, new explanation/theory which attempts to account for the range of findings

*Meta-analysis*: Use of statistical techniques to synthesize results into a single quantitative estimate of an effect.

## The purpose

- To weight the strength and direction of the published evidence in relation to a question
- To identify the areas of uncertainty
- To identify gaps in knowledge (in general and in a particular context)
- For a treatment (method, process, tool, ...)
  - To identify what is effective/cost-effective and to reduce uncertainty in estimates of effectiveness in general
  - To identify what is likely to be effective in particular populations and institutional contexts
  - To help develop new interventions which may work
- This should be synthesized so that it provides valuable evidence on which specific decisions can be based.

[ **simula** . research laboratory ]

## Desirable features

- Systematic (no bias, all relevant studies included, up-to-date)
- Rigorous
- Explicit (transparent methods)
  - Search
  - Evaluation criteria
- In practice, reviews will be iterative and not completely explicit. In your case, the search may not be fully systematic and rigorous.

[ **simula** . research laboratory ]

## How to synthesize

- No mechanical process available (other than for meta-analysis based synthesis)
- Typical process:
  - Preliminary synthesis of individual results to organize findings, get a sense of patterns and develop understanding of effects
  - Exploration of relationships of findings wrt:
    - Similarity of results
    - Variation in results
    - Contradictions, context dependencies
  - Formulation of general results consistent with the individual results – and the robustness/trustworthiness/limitations of the general results
  - Identify research gaps

**NB: Remember that the synthesis should be relative to the research question! Clarify the purpose of the synthesis in the beginning of your report.**

[ **simula** . research laboratory ]

## Other things to remember ...

- Preferably, the synthesis should be formulated so that it can easily be used to guide a decision.
  - This requires an understanding of organizational politics, user needs, etc.
  - Synthesis conclusions should therefore often be written in a language used by the decisions makers.
- Assessment of publication bias
  - Is it likely that some results (e.g., no difference in effect) are not likely to be published?
- Synthesis is similar to “pattern matching”.
  - Avoid seeing patterns that are not there – ref. earlier presentations

[ **simula** . research laboratory ]

## Example

- What do we know about agile development?
  - IEEE Software, Dybå and Dingsøy

[ **simula** . research laboratory ]

## Exercise

- Research question: When does pair programming pays off?
- Searches using google scholar/ISI web of knowledge
  - Automatic search, manual search, snowball search ...
  - Search domains:
    - Exploratory testing vs test-case based testing
    - Expert estimation vs model-based estimation

[ **simula** . research laboratory ]

EXTRA (If time permits ...)

## **EXAMPLE OF USE OF EVIDENCE-BASED PRINCIPLES (WHEN SELECTING A COMPANY FOR YOUR PROJECT)**

[ **simula** . research laboratory ]



**How much is a great  
developer worth?**

## Research on productivity differences ...

- First study in 1966, with 12 experienced programmers (Sackman, Erickson & Grant):
  - Effort difference 1:16 and 1:25
  - Size difference 1:6 and 1:5
- Summary of individual programming productivity from 61 experiments (5-36 persons) (Prechelt, 1999)
  - Typical difference between best and worst about 1:15
  - Typical difference between one in "slower quarter" and one in "faster quarter" about 1:5
- Four companies developing the same system (Anda, Sjøberg et al., 2009)
  - Effort difference of about 1:3 (including client effort)
  - Size difference of about 1:2

[ **simula** . research laboratory ]

## Own research: The 6 best companies out of 16 companies bidding for our project

	Comp. A	Comp. B	Comp. C	Comp. D	Comp. E	Comp. F
Price	Very low	Low (2x)	Medium (3x)	High (5x)	Very high (12x)	Very high (14x)
Est. effort	Very low	Low (1.5x)	Medium (3x)	High (8x)	Medium (4x)	Very high (8x)
CV	OK	OK	Good	Good	Good	OK
Refs.	Very good	Very good	Very good	Very good	Very good	Very good
Proposal	OK	OK	Good	OK	OK	OK
Country	Finland	Malaysia	India	India	Canada	US

Which company would you select?

[ **simula** . research laboratory ]



## Before I give you the results ... It is not easy to be a client.

- As a client you have to decide whether a very low price or effort estimate (such as the one by Company A) indicates:
  - High productivity and skill (great developer)
  - High degree of over-optimism, leading to unrealistic plans
  - Low skill (the Dunning-Kruger effect, where those unskilled are less aware of their lack of skill)
  - Lower expected quality of the product
  - More problematic process with the provider (typical when fixed price projects and a bidder with low price is selected)
- In short, should we take the risk of selecting Company A with its low price and low effort estimates?

[ **simula** . research laboratory ]

## Our study of more than 800.000 projects at freelancer.com shows that

- Clients tend to avoid companies/developers with unusually low price, **even when the companies document the the same level of competence as the one selected!**
  - Experience from Norwegian software industry indicates that this does not necessarily hold for large scale projects costing millions, where they are more likely to select low price bidders ...
- A fear of low price is, to some extent rational. Our data shows that:
  - Low price makes, on average, good companies perform worse (due to overoptimistic estimates)
  - Low price correlates with higher risk of project failure

[ **simula** . research laboratory ]

## The study also showed that:

- The best predictor of non-failing projects, was "previous successful collaboration with the client"
  - Can be seen as a very realistic test of the provider
- Client skill was almost as important as the skill of the provider to predict project failure
- Systematic and large differences between project failure rates in different outsourcing countries.
- Among the larger outsourcing countries:
  - Lowest failure rates: Argentine, Eastern European countries
  - Highest failure rates: South Asia (India, Pakistan, Bangladesh)

[ **simula** . research laboratory ]

## So, the clients may be rational, BUT ...

**Clients avoiding companies with low price or low effort estimates may also avoid the companies with low price due to great developers!**

Let's go back to our 6-company study ...

[ **simula** . research laboratory ]

## Repetition: The six good looking companies

	Comp. A	Comp. B	Comp. C	Comp. D	Comp. E	Comp. F
Price	Very low	Low (2x)	Medium (3x)	High (5x)	Very high (12x)	Very high (14x)
Est. effort	Very low	Low (1.5x)	Medium (3x)	High (8x)	Medium (4x)	Very high (8x)
CV	OK	OK	Good	Good	Good	OK
Refs.	Very good	Very good	Very good	Very good	Very good	Very good
Proposal	OK	OK	Good	OK	OK	OK
Country	Finland	Malaysia	India	India	Canada	US

[ **simula** . research laboratory ]

## We selected all six ... Here is how they performed

	Comp. A	Comp. B	Comp. C	Comp. D	Comp. E	Comp. F
Actual effort	Very low	Low (3x)	High (6x)	High (8x)	Very high (18x)	Very high (16x)
Error fixing effort	Very low	High (4x)	Medium (2.5x)	High (4x)	Very high (8x)	Extr. high (20x)
Maintenance effort	Very low	High (6x)	Very high (11 x)	High (8x)	Extr. high (26x)	Extr. high (20x)
Lines of code	Very low	Low (2x)	Low (1.5x)	Medium (3x)	High (4x)	Low (1.5x)

Company A had a great developer, but we would probably not have chosen that company in the normal case when selecting only one developer. Simply too risky without knowing more about the competence. Middle is more safe ...

[ **simula** . research laboratory ]

## What can we learn?

- Huge differences in software development productivity, quality and maintenance cost for even simple systems
- Not easy to identify great developers from CVs, satisfaction of previous clients and quality of proposals
- The real differences will typically remain unknown to the clients, the managers of the developers and probably to the developer themselves, as clients select only one provider

[ **simula** . research laboratory ]

## Consequences

- The salaries of developers and payment by clients are not even close to reflecting the real differences in performance
- We need better ways to assess the competence of developers and companies

[ **simula** . research laboratory ]

## Hiring : State of practice

- Many employers are currently using suboptimal selection methods for hiring of software developers
  - Both for consultants and permanent employment
- Studies shows that when recruitment personnel are updated on relevant academic research, their companies perform better economically

[ **simula** . research laboratory ]

## Interviews (unstructured)

- Often used for hiring developers
  - Cheap and straight-forward method
  - Interviewer are often over-confident in their interviewing skills
- Research has repeatedly documented that this is a poor selection mechanism
  - Over-emphasize irrelevant information and contextual knowledge
  - Difficult to compare candidates
  - Probably even worse for selection in offshoring contexts



[ **si** ]

## “Clouds Make Nerds Look Better”



- We know that interviews are influenced by the candidate's weight, attractiveness, speed of speech, etc
- Study of university applicants:
  - 12% higher chance when sunshine compared to worst cloud cover
  - Sunshine means more focus on social skills, cloudy means more focus on academic skills
    - Nerd-factor measured as academic rating divided by social rating (e.g., leadership).

[ **simula** . research laboratory ]

155

## Structured interviews

- Good selection method for hiring top performers
  - Unlike unstructured interviews
- How does structured interviews differ?
  - Questions determined by a careful analysis of the job in question
  - Usually the same questions to all candidates
  - Predefined scoring of responses and rules for candidate evaluation
- In practice, structured interviews are very similar to testing of candidates

[ **simula** . research laboratory ]

## Biographical information (CV)

- Useful for initial screening
- Research suggest that you should emphasize:
  - University grades
  - Past job performance (preferable from similar jobs)
  - Relevance of experience / education
- ...and don't emphasize:
  - Years of experience
  - Knowledge of specific technologies & frameworks
  - "Buzzword compliance"
  - Activities unrelated to work
- Research shows that strong candidates benefit from excluding less relevant information in their CVs

[ **simula** . research laboratory ]

## References

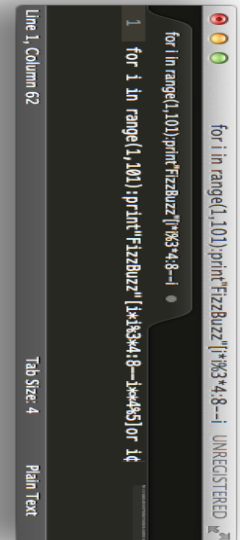
- Checking references and networks may be a valuable if you can trust them to give you honest and complete information
  - NB! Most people find it difficult to reveal negative information
  - NB! Sometimes job performance is strongly dependent on job environment
  - NB! Expertise can be surprisingly narrow

[ **simula** . research laboratory ]

## Test of programming skills

- Useful for filtering out candidates that lack programming skills
  - Lots of tests are available
  - Typically "programming puzzles"
- ... but remember that many other factors also impact software development performance
  - Ability to share/reuse code
  - Team work / communication
  - Requirement engineering skills
  - Etc

[ si



```
for i in range(1,101):print "FizzBuzz" if %3==0 and %5!=0 else i
1 for i in range(1,101):print "FizzBuzz" if %3==0 and %5!=0 else i
```

Line 1, Column 62 Tab Size: 4 Plain Text

## Work sample tests

- Highly recommended method
  - Better than general programming tests
  - Typically small, but complex tasks
  - The more representative tasks, the better results - context-specific problems
  - Examples: Fix a bug in the system, design a new feature
- Take measures to avoid cheating
  - Change tasks frequently
  - Use pair-programming/blackboards

[ simula . research laboratory ]



## General Mental Ability (GMA)

- Intelligence at work is not wholly different from intelligence at school
  - Intelligent people acquire job knowledge faster and acquire more of it
  - Inexpensive tests are available, e.g. Wonderlic tests
- Research shows that GMA nicely complements structured interviews and tests
  - ...but GMA is rarely used for hiring of software developers
  - Prejudice against high IQ (bad at communication, etc)
  - We may, wrongly, assume small differences within the same profession (e.g. software developers)

[ **simula** . research laboratory ]

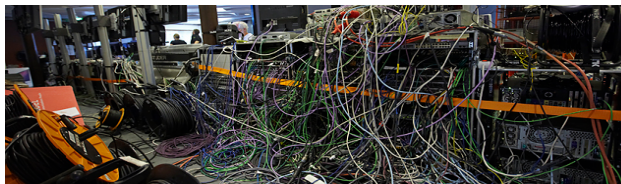
## NB! Selection of top performers is not the only way to increase productivity

- We can also increase productivity by, e.g.
  - Reducing system complexity
  - Improving software development tools and methods
  - Improving the work environment
  - Improving processes analysis and specification work
- Hiring top performers is not even always wanted
  - Sometimes bad for team dynamics
  - Issues with cost and competition
  - Lack of challenges / more easily bored

[ **simula** . research laboratory ]

## Reduce system complexity

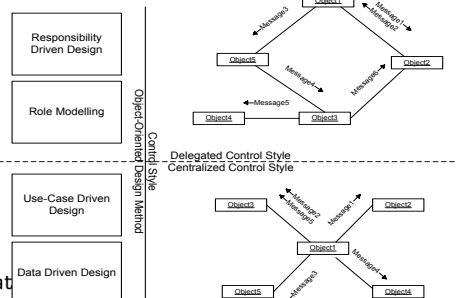
- Productivity differences correlates with job complexity
  - Reduce system complexity -> reduce differences
- Many well-known approaches to reduce complexity
  - Modularization
  - Consistency
  - Conventions and Documentation
  - Simple, easy to understand, design patterns



[ simula . research

## Centralized vs. Delegated design

- In a study by Erik Arisholm, 500+ performed maintenance tasks on two alternative designs of the same system
- Purpose: Study the effect of centralized vs. delegated design (the latter often considered better)



[ simula . research laborat

## Results

- In the delegated design, the maintenance tasks took more time and had more errors
- Only the most experienced developers seemed to have the necessary skills to utilize the more elegant delegated design

[ **simula** . research laboratory ]

## Summary

- Productivity differences are huge among software developers
  - Even for developers with similar CV, experience, education, etc
- It is hard to select the top performers
- Recommended: GMA test in combination with either structured interviews and work-samples
  - The huge economic benefits of selecting top performers makes up for the additional costs of these selection methods

[ **simula** . research laboratory ]