Control versus Realism in Software Engineering Studies: Experiences from Simula Research Laboratory

Professor Dag Sjøberg

Simula Research Laboratory/

University of Oslo

Plan for the talk

1. Simula Research Laboratory
2. Increasing the realism of experiments
3. Increasing the control of case studies
4. Support environments and resources
5. Conclusion

Simula Research Laboratory

1991: Decision to close the airport at Fornebu, Oslo

2000: The Parliament decides that IT-Fornebu shall develop a Knowledge Park at the old airport

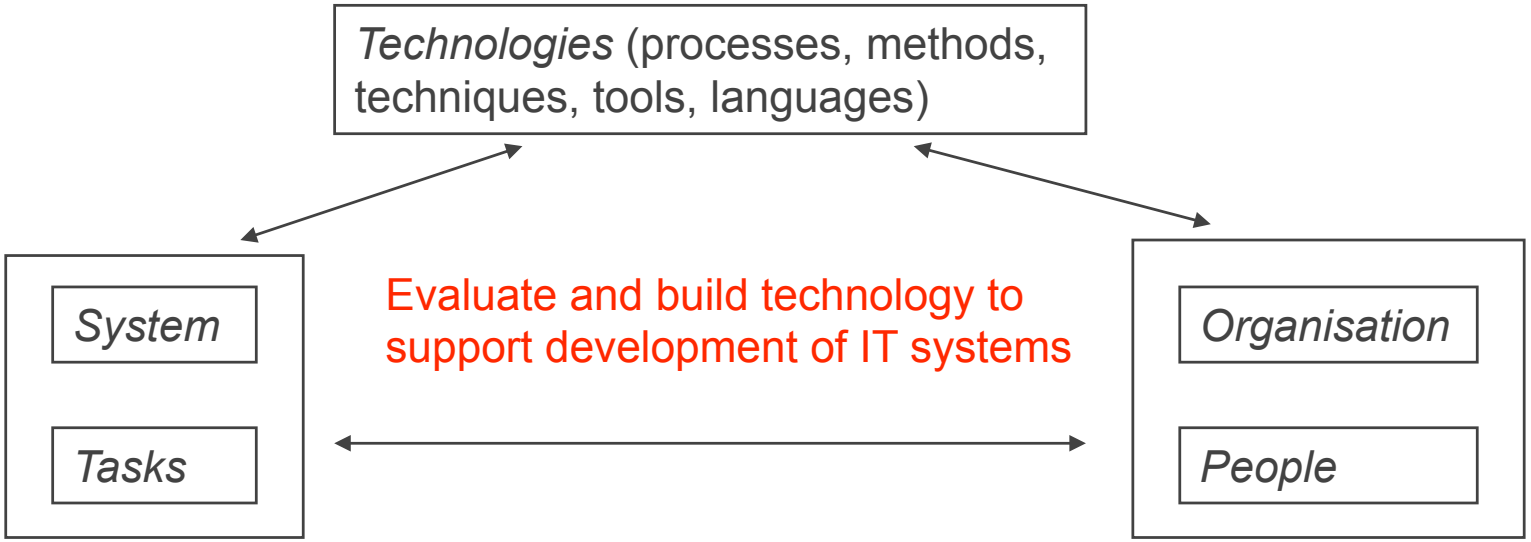
2000: Three research groups selected from 17 Norwegian university groups

2001: Simula established

2009:

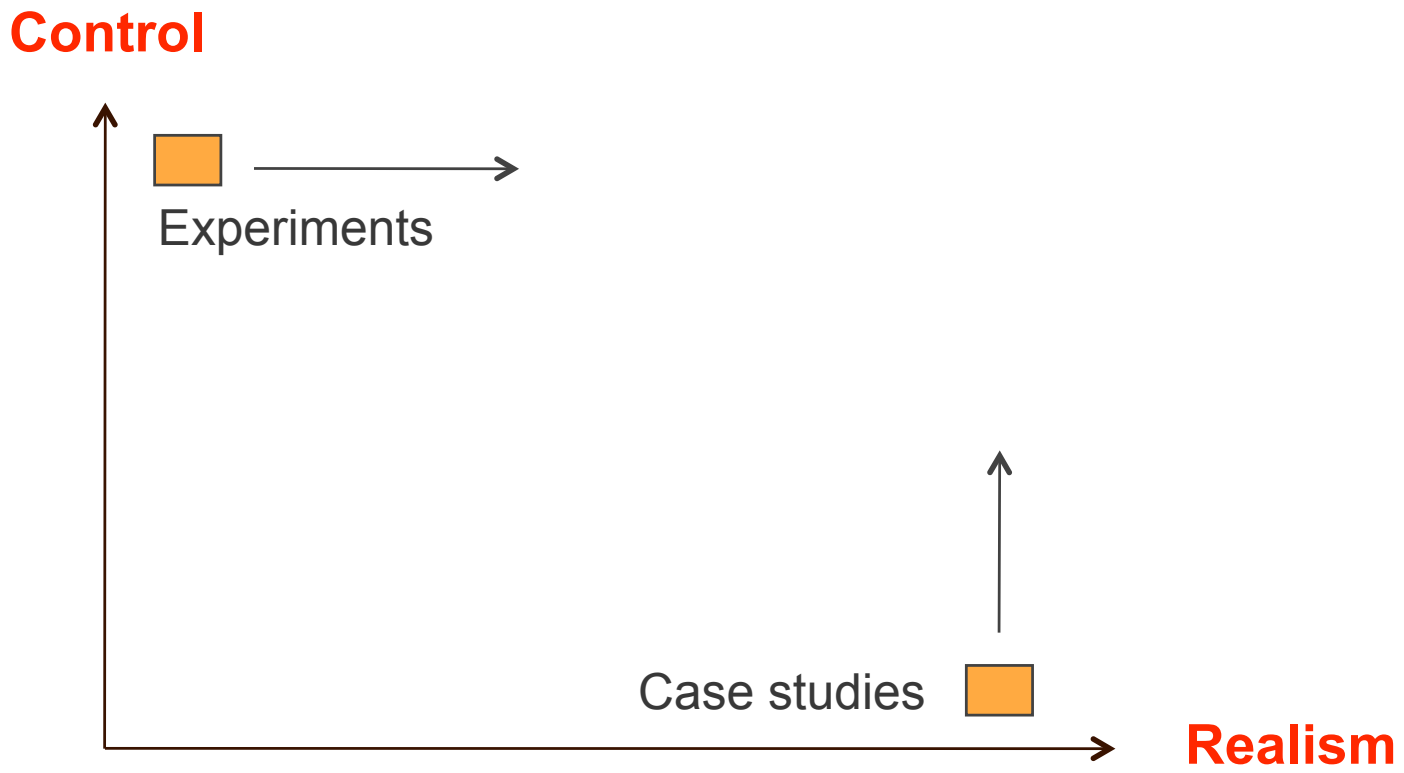
- 100 employees
- Shareholding company (Norwegian state: 80 %, Sintef and Norwegian computing centre: 20 %)
- Research departments
  - *Networks and Distributed Systems*
  - *Scientific Computing*
  - *Software Engineering*
- Simula Innovation

SE research at Simula



- How should the industry (and others who build software) judge which software development technologies are useful when?

Control vs. Realism



How can we scientifically conclude from our empirical studies and how can we convince engineers and managers in industry that the results/conclusions are relevant to them?

Definition of experiment

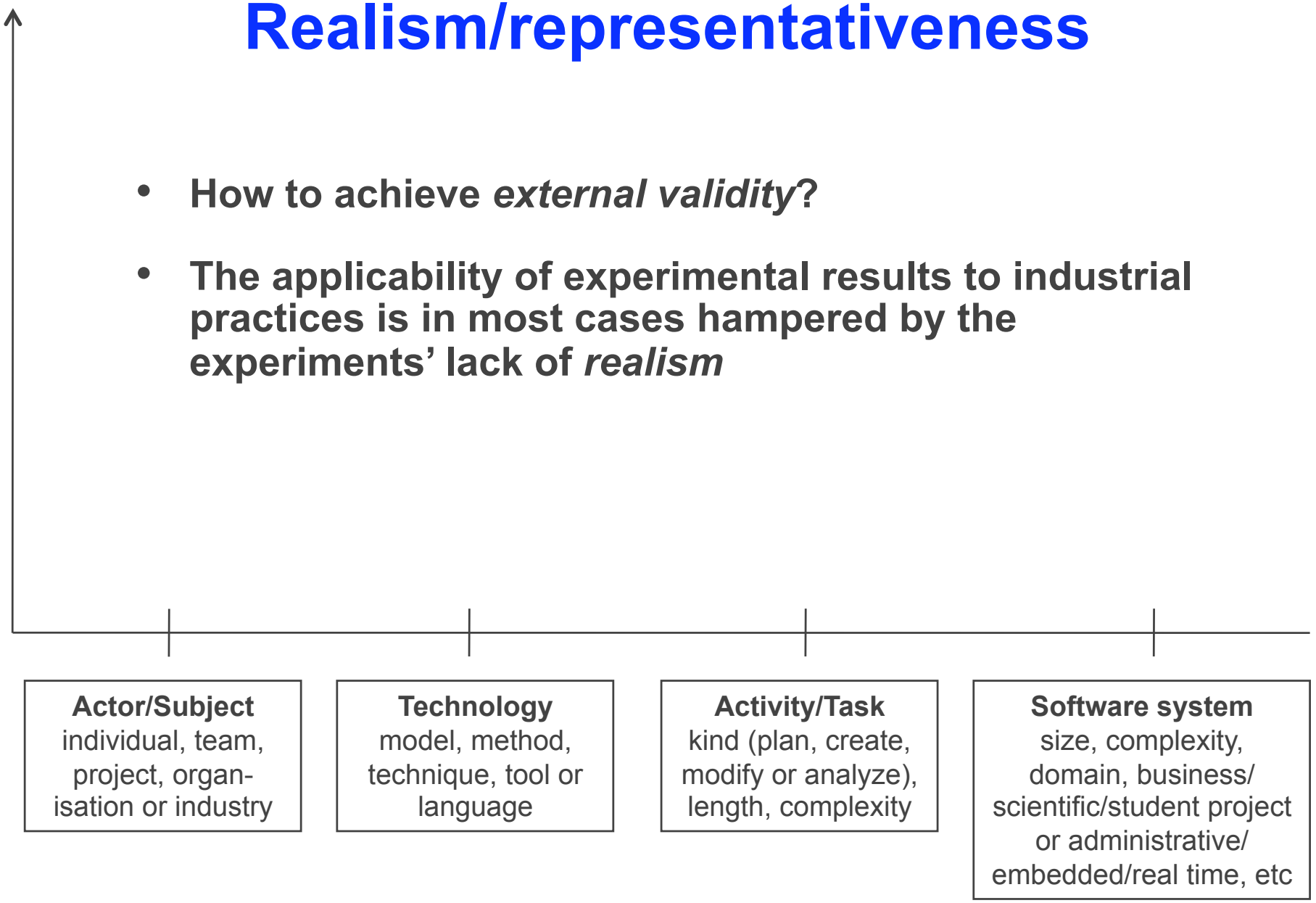
Controlled experiment in software engineering is a study in which individuals or teams (the experimental units) conduct one or more software engineering tasks for the purpose of comparing different treatments (populations of subjects, processes, methods, techniques, languages, or tools)

Control to help ensure Internal Validity

The *internal validity* of an experiment is "the validity of inferences about whether observed co-variation between *A* (the presumed treatment) and *B* (the presumed outcome) reflects a causal relationship from *A* to *B* as those variables were manipulated or measured" [Shadish, 2002]. Changes in *B* may have alternative causes than the manipulation of *A*.

Realism/representativeness

- **How to achieve *external validity*?**
- **The applicability of experimental results to industrial practices is in most cases hampered by the experiments' lack of *realism***



Actors/subjects

Subject Category	Reported Subject Types	N	%
Undergraduates	Undergraduates, Bachelors, Third and fourth-year students, Last-year students, Honors and Majors.	2969	54.1
Graduates	Graduate students, Students following graduate courses or Master's programs, MSc and PhD students.	594	10.8
Students, type unknown	Students in computer science, Students.	1203	21.9
Professionals	Developers, Practitioners, Software engineers, Analysts, Domain experts, Business managers, Facilitators, Professionals.	517	9.4
Scientists	Professors, Post-doctorates, Staff members of educational institutions.	74	1.3
Unknown		131	2.3
Total		**5488**	**100**

Actor/Subject
individual, team, project, organ-isation or industry

Technology
model, method, technique, tool or language

Activity/Task
kind (plan, create, modify or analyze), length, complexity

Software system
size, complexity, domain, business/ scientific/student project or administrative/ embedded/real time, etc

State of the art in SE experimentation

*Sjøberg *et al.*, A survey of controlled experiments in software engineering, IEEE Transactions on Softw. Engineering 31(9) (2005), pp. 733–753.

		Articles reporting controlled experiments	
Journal/ Conference	Total no. of articles investigated	N	Row %
EMSE	124	22	17.7
ISESE	20	3	15.0
METRICS	177	10	5.6
JSS	886	24	2.7
TSE	687	17	2.5
ICSE	520	12	2.3
IST	745	8	1.1
SME	186	2	1.1
IEEE SW	532	4	0.8
TOSEM	125	1	0.8
IEEE Comp	780	0	0
SP&E	671	0	0
All	**5453**	**103**	**1.9**

- **Does it matter? Isn't the relative performance of a technology the same independently of the type of subjects?**
- **Is a helicopter better than a bike?**
- **Is pair programming better than solo programming?**

A (quasi) experiment on pair programming

295 *junior*, *intermediate* and *senior* professional Java consultants from 29 companies were paid to participate (one work day)

99 individuals (conducted in 2001/2002)

98 pairs (conducted in 2004/2005)

- Norway: 41
- Sweden: 28
- UK: 29

The pairs and individuals performed the same Java change tasks on either:

- a ”*simple*” system (centralised style) or
- a ”*complex*” system (delegated style)

We measured duration (elapsed time), effort (cost) and correctness

Why that many subjects? Power analysis

Research question:

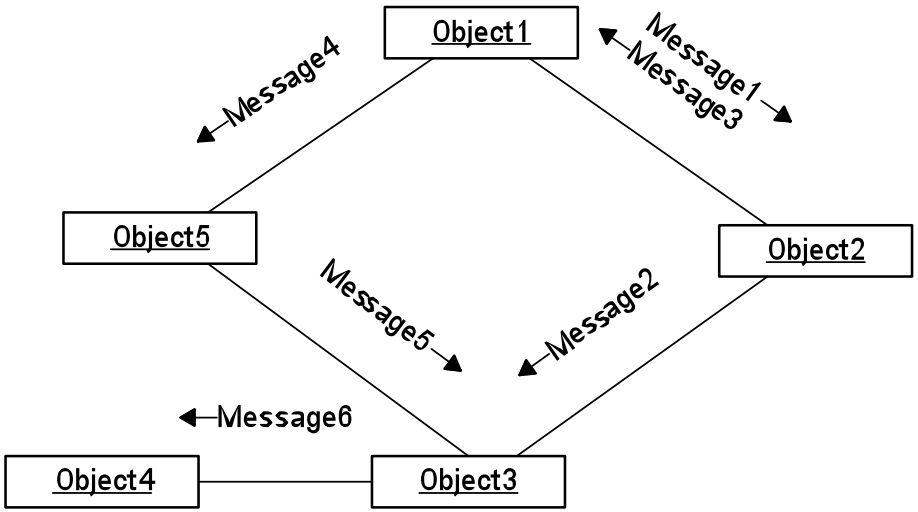
What is the effect regarding duration, effort and correctness of pair programming for various levels of system complexity and programmer expertise when performing change tasks?

2x2x3 fixed-effect analysis of covariance:
pair programming (two levels), control style (two levels) and expertise (three levels), resulting in twelve levels/groups

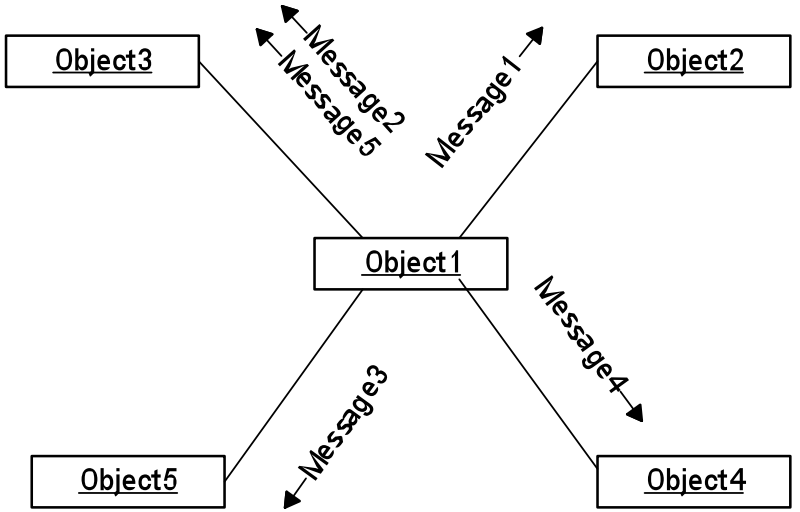
***N* = 170 (85 individuals and 85 pairs)**

N = 14 in each of the 12 groups

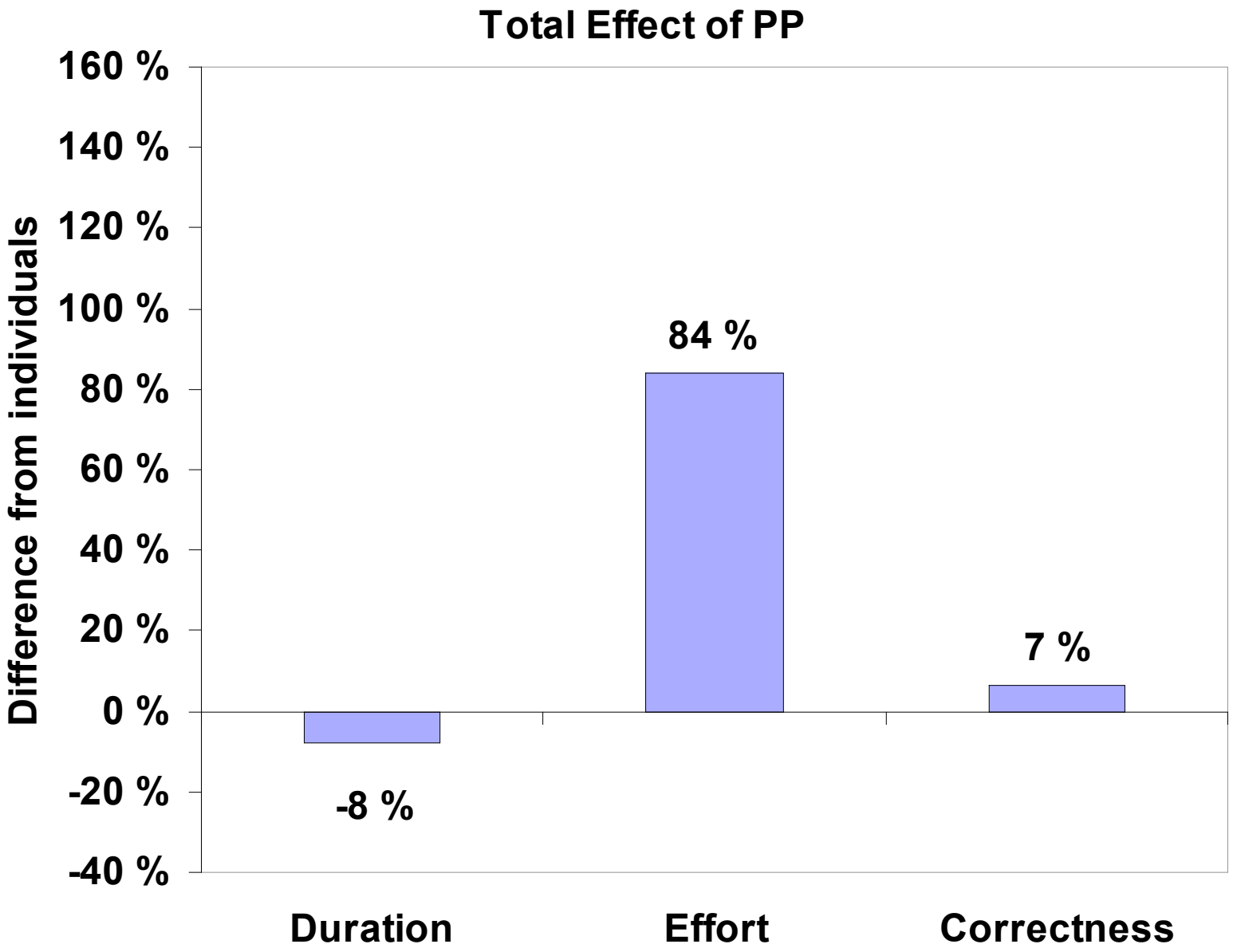
Object-Oriented Design Styles

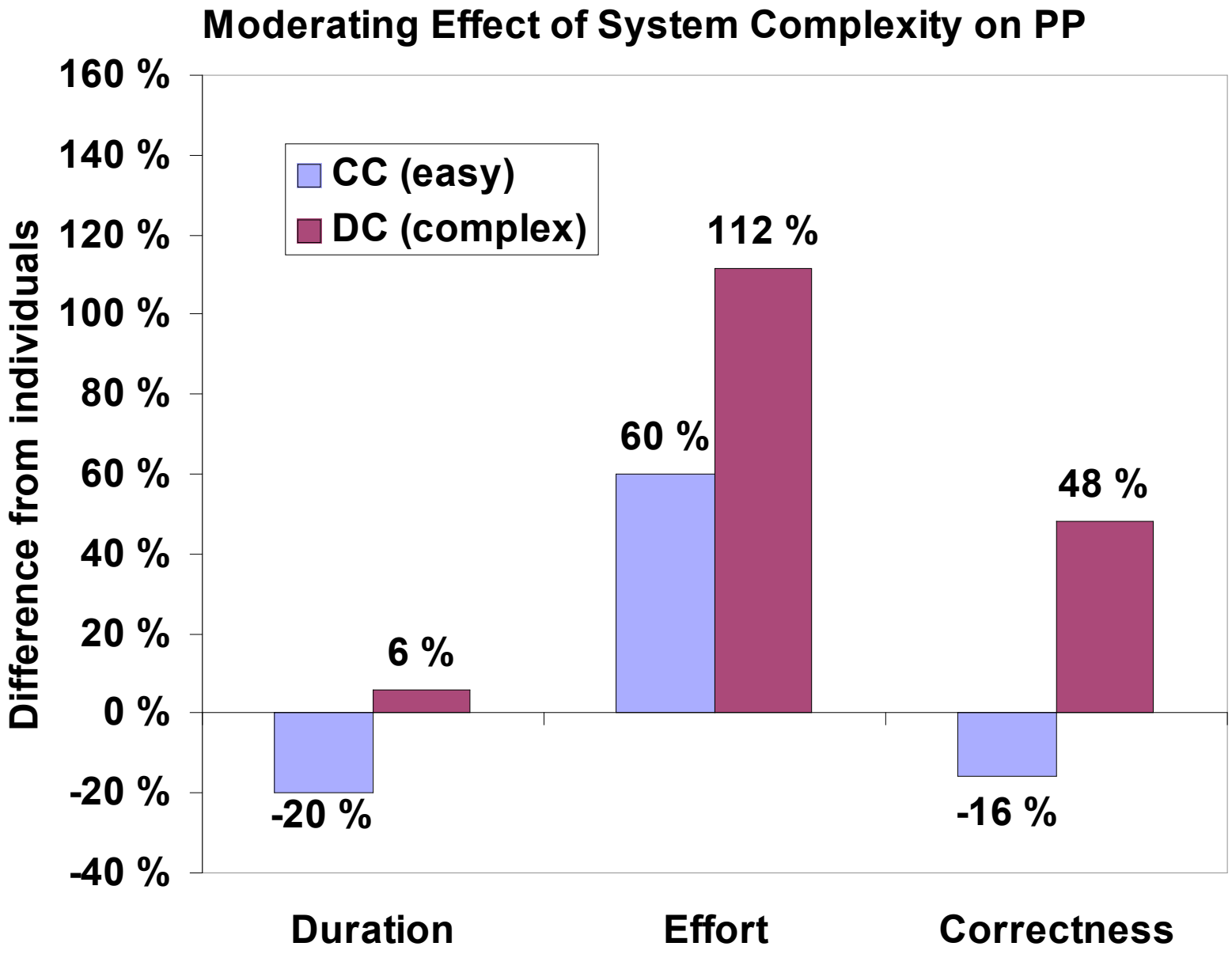


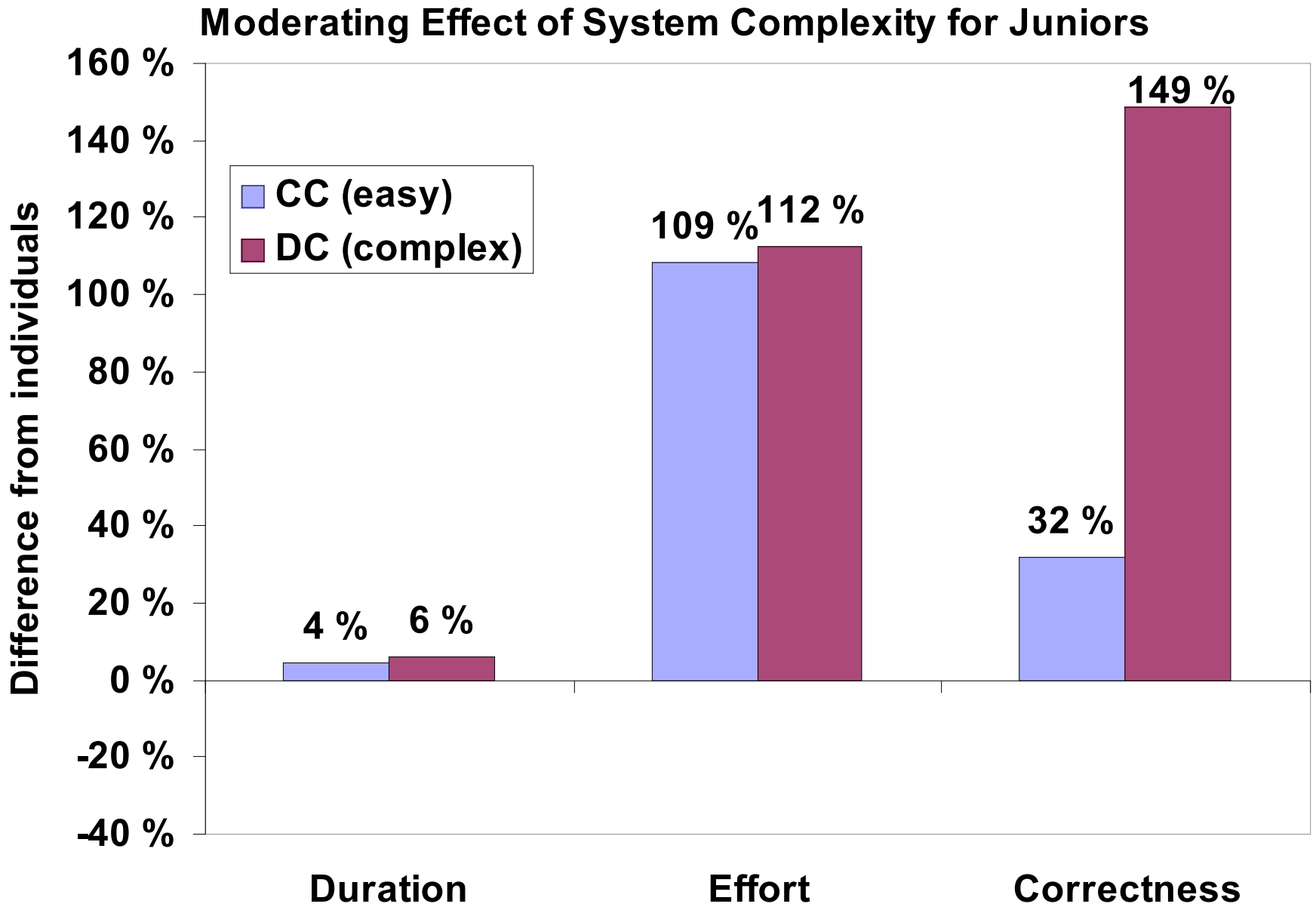
Delegated Control Style

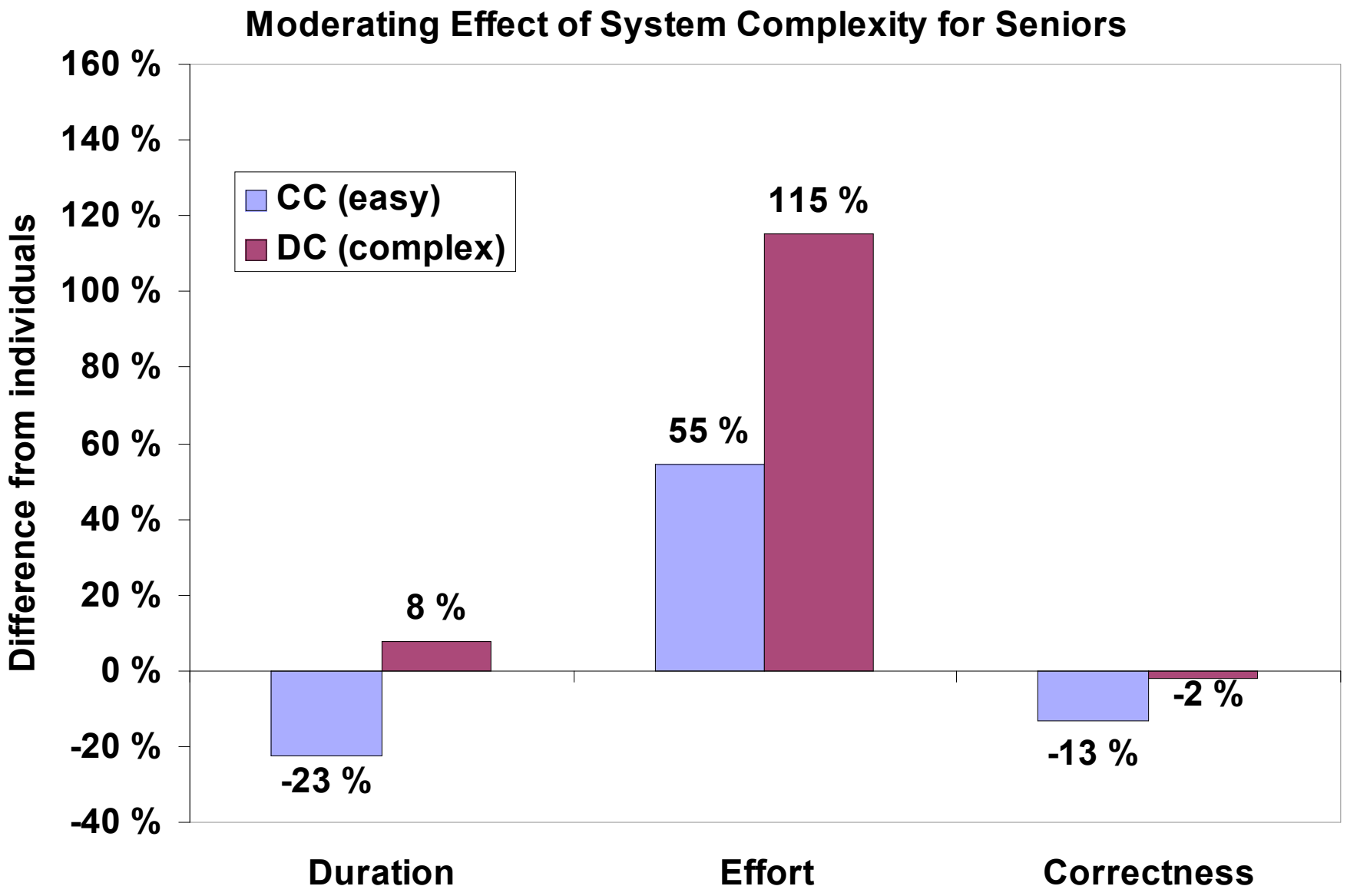


Centralized Control Style


Total Effect of PP
Difference from individuals
160 %
140 %
120 %
100 %
80 %
60 %
40 %
20 %
0 %
-20 %
-40 %
84 %
7 %
-8 %
Duration
Effort
Correctness


Moderating Effect of System Complexity on PP
Difference from individuals
160 %
140 %
120 %
100 %
80 %
60 %
40 %
20 %
0 %
-20 %
-40 %
CC (easy)
DC (complex)
-20 %
6 %
60 %
112 %
-16 %
48 %
Duration
Effort
Correctness


Moderating Effect of System Complexity for Juniors
Difference from individuals
160 %
140 %
120 %
100 %
80 %
60 %
40 %
20 %
0 %
-20 %
-40 %
CC (easy)
DC (complex)
4 %
6 %
109 %
112 %
32 %
149 %
Duration
Effort
Correctness


Moderating Effect of System Complexity for Seniors
Difference from individuals
160 %
140 %
120 %
100 %
80 %
60 %
40 %
20 %
0 %
-20 %
-40 %
CC (easy)
DC (complex)
-23 %
8 %
55 %
115 %
-13 %
-2 %
Duration
Effort
Correctness

The effect of PP "depends on"

Programmer expertise	Task complexity	Use PP?	Comments
Junior	Easy	Yes	Provided that increased quality is the main goal
	Complex	Yes	Provided that increased quality is the main goal
Intermediate	Easy	No	
	Complex	Yes	Provided that increased quality is the main goal
Senior	Easy	No	
	Complex	No*	

* Unless you are sure that the task is too complex to be solved satisfactorily even by solo seniors

The performance of the various categories may depend on their relevant education, work experience, the actual task and system, development technology, etc.

In the survey of 113 experiments, 7 involved both students and professionals. Only 3 measured difference in performance: partly no difference, partly professionals better.

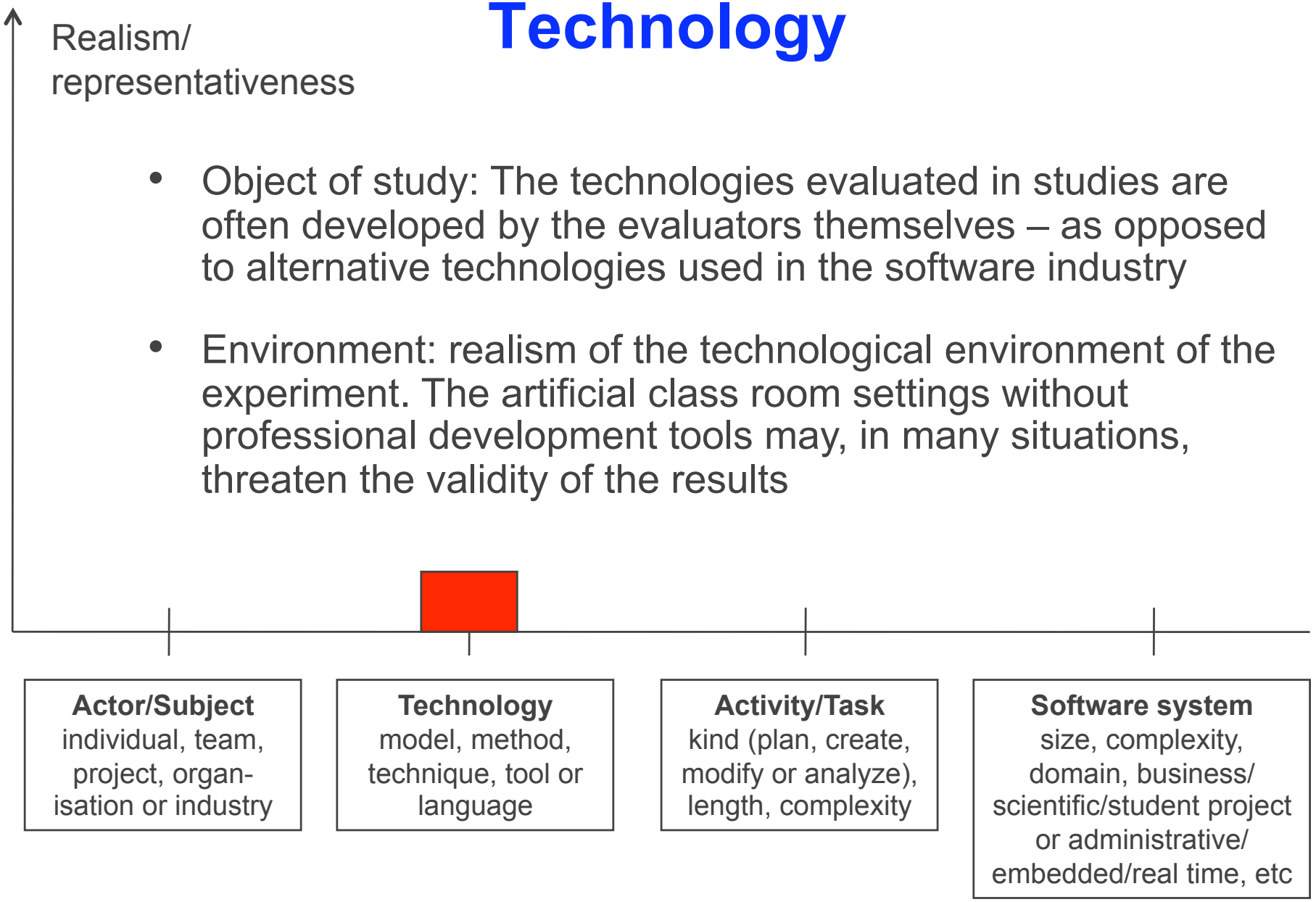
Why is scale important?

- Easier to obtain a representative sample of the target population.
  - One of 113 experiments reported sampling from a well defined target population
- Many aspects of the complexity of software engineering only manifest themselves in controlled experiments if the experiments involve a sufficiently large number of subjects and tasks, for example, differences among subgroups of subjects

Realism (representativeness) of technology, tasks and systems

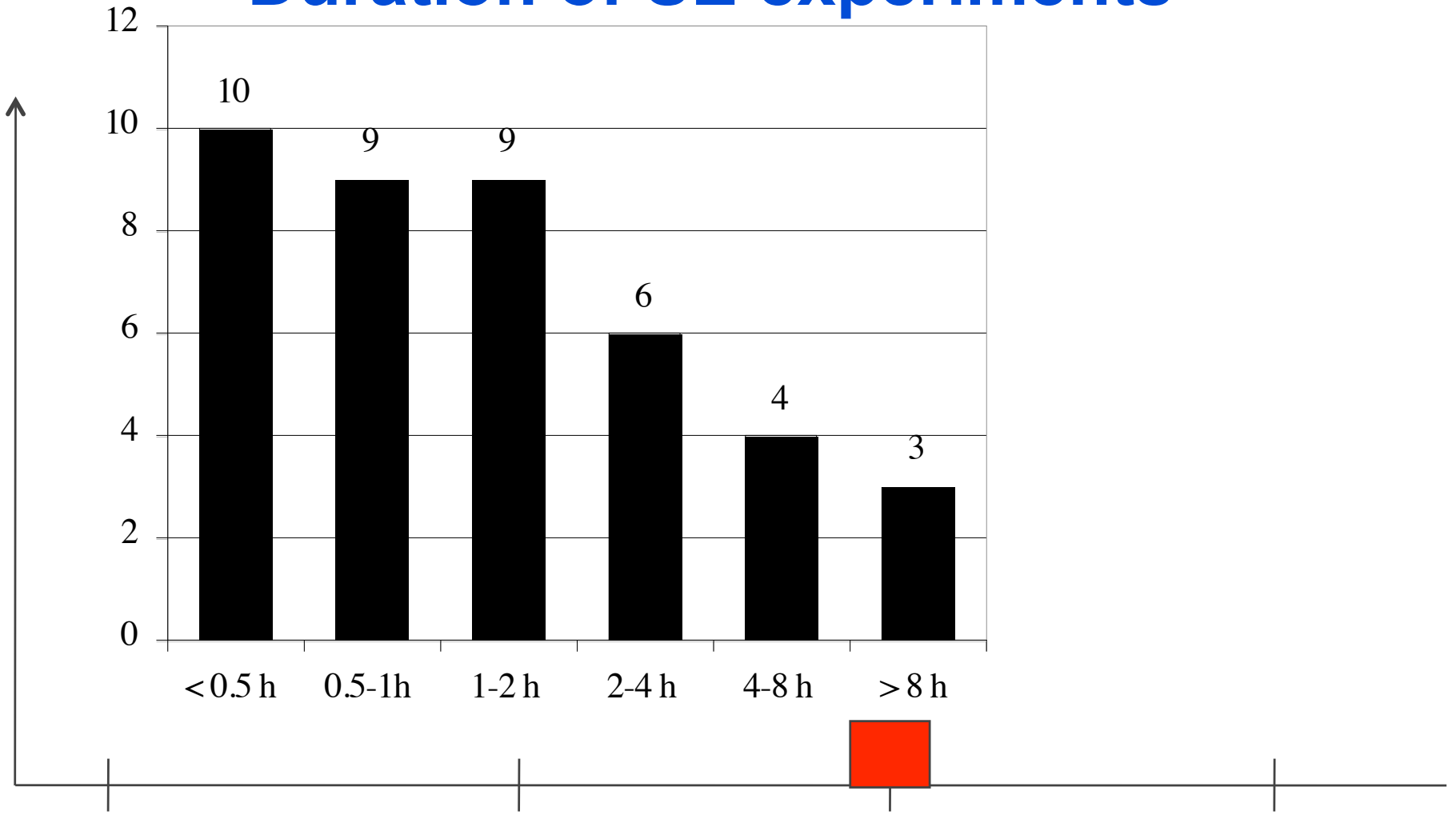
- A grand challenge in SE experimentation is how we generalise from the specific technology, tasks and systems of SE experiments
- Not aware of suitable taxonomy or classification of these aspects for SE
- Nevertheless, development tasks in industry usually take longer and are often more complex than is the case in most experiments – as is the case with technology and systems

Technology



- Object of study: The technologies evaluated in studies are often developed by the evaluators themselves – as opposed to alternative technologies used in the software industry
- Environment: realism of the technological environment of the experiment. The artificial class room settings without professional development tools may, in many situations, threaten the validity of the results

Duration of SE experiments



Actor/Subject
individual, team, project, organ-isation or industry

Technology
model, method, technique, tool or language

Activity/Task
kind (plan, create, modify or analyze), length, complexity

Software system
size, complexity, domain, business/ scientific/student project or administrative/ embedded/real time, etc

Systems

Application type	N	%
Constructed	80	70.8
Commercial	16	14.2
Student project	5	4.4
Open source	0	0.0
Unclear	12	10.6
Total	**113**	**100**

Actor/Subject
individual, team, project, organ-isation or industry

Technology
model, method, technique, tool or language

Activity/Task
kind (plan, create, modify or analyze), length, complexity

Software system
size, complexity, domain, business/ scientific/student project or administrative/ embedded/real time, etc

Experiments/studies at Simula

99 consultants from 8 companies

- **one-day** experiment that compared two different object-oriented control styles

295 consultants from 29 companies in Norway, Sweden and the UK

- **one-day** experiment that tested the effect of pair programming

39 consultants from 11 companies

- **Three-day** experiment on design patterns

20 programmers from 13 companies

- worked individually from **one to two weeks** in an experiment on UML **(real system)**

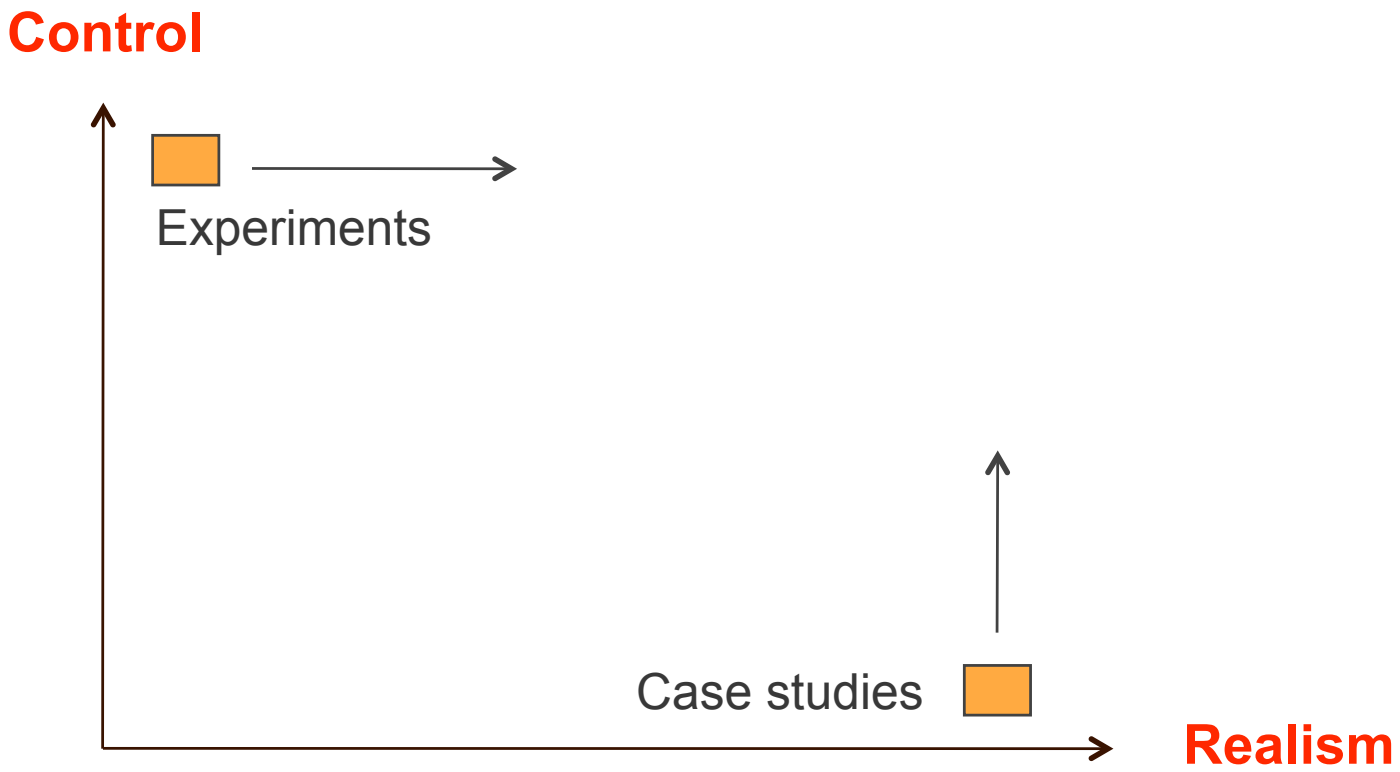
35 companies presented bids for a web-based system that we needed

- 4 were selected to build the system **(real system)** independently of each other.
- The teams (2-3 developers from each company) spent from **7 to 25 person-weeks** each

30 companies from 11 countries in Europe and Asia presented their bids.

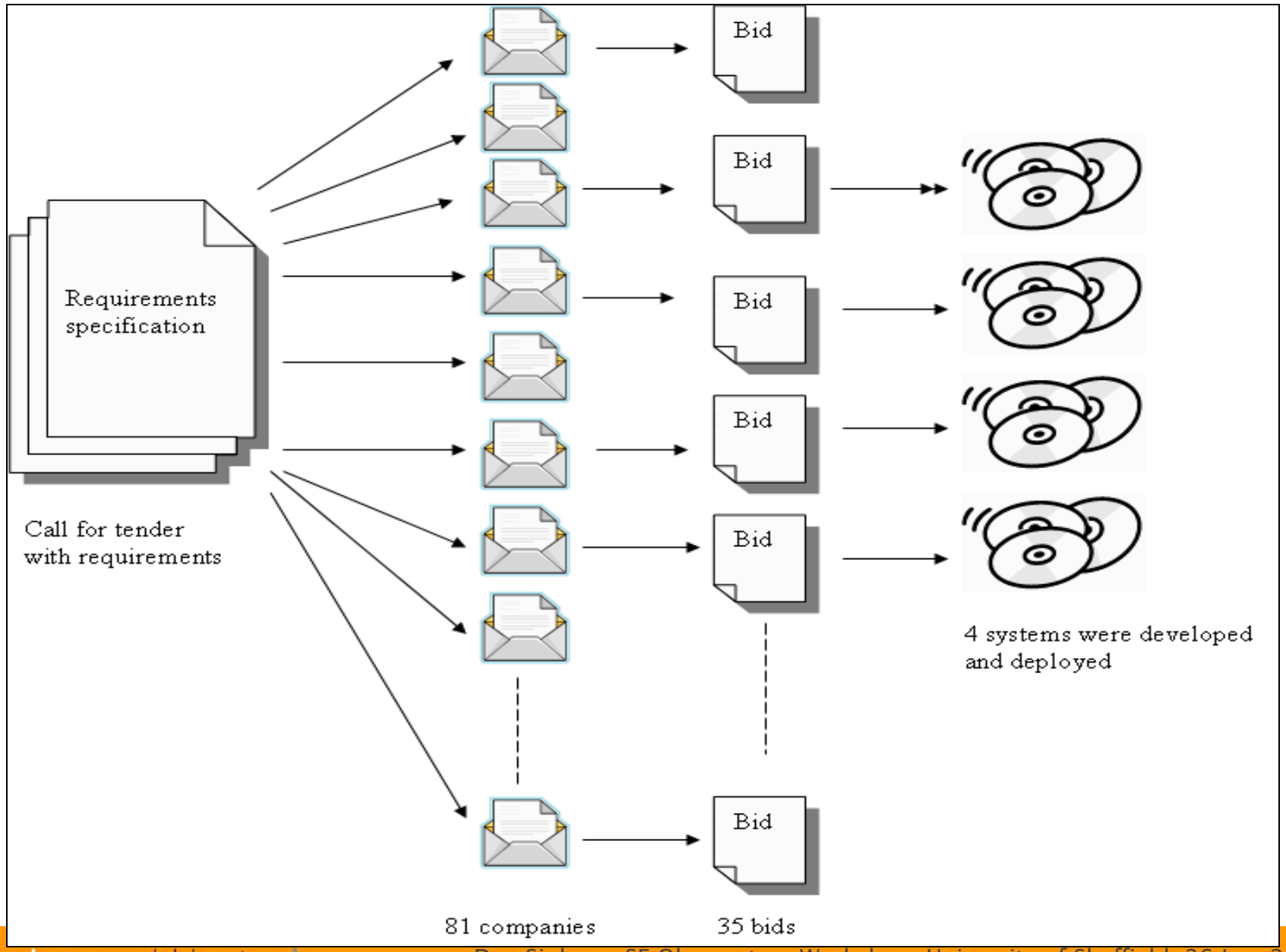
- 4 companies built the system **(real system)**
- each spent from **10 to 20 person-weeks**

Control vs. Realism



- **How to increase control in case studies?**

A field experiment + multiple-case study



Part 1: Full realism experiment on bidding

Two phases in bidding process:

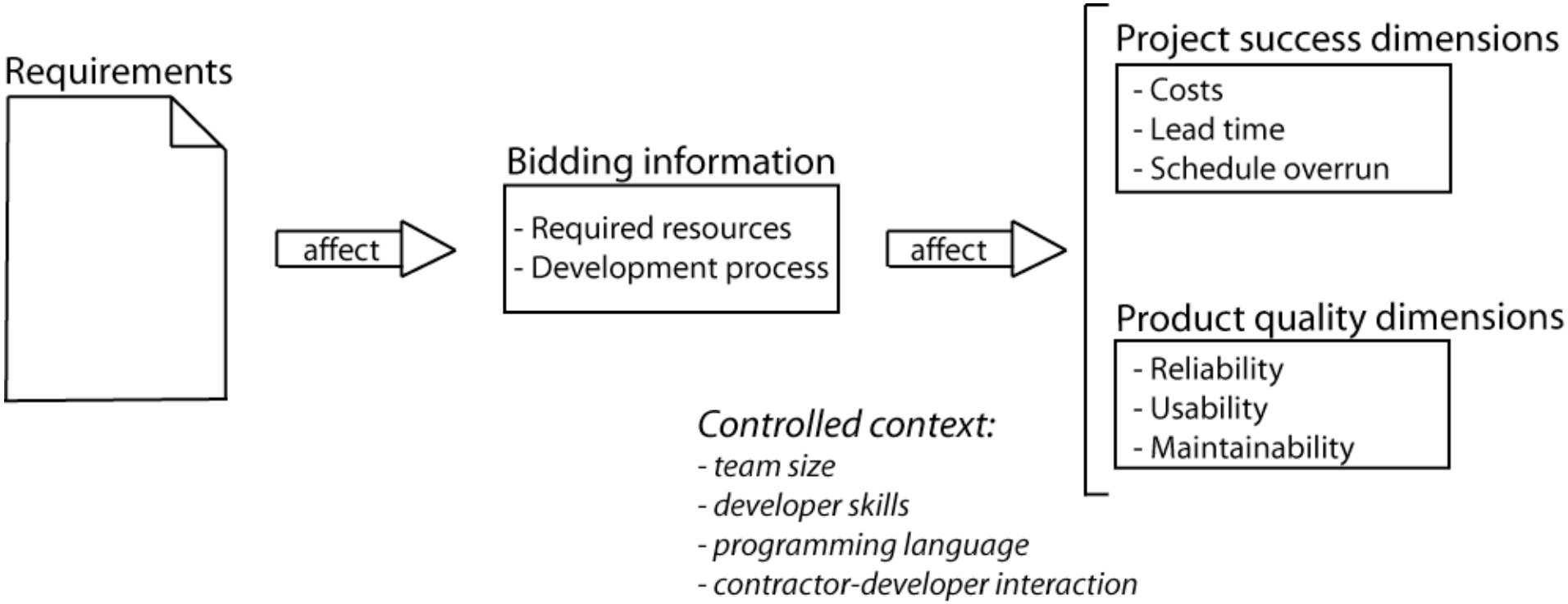
- In pre-study phase, 17 of the 35 bidding companies indicated price based on an incomplete description of user requirements
- In the bidding phase, all 35 companies provided bids based on a more complete requirement specification with substantially more functionality than the system indicated in the pre-study phase

The 17 companies involved in the pre-study phase presented bids 70% higher than the bids of the other companies.

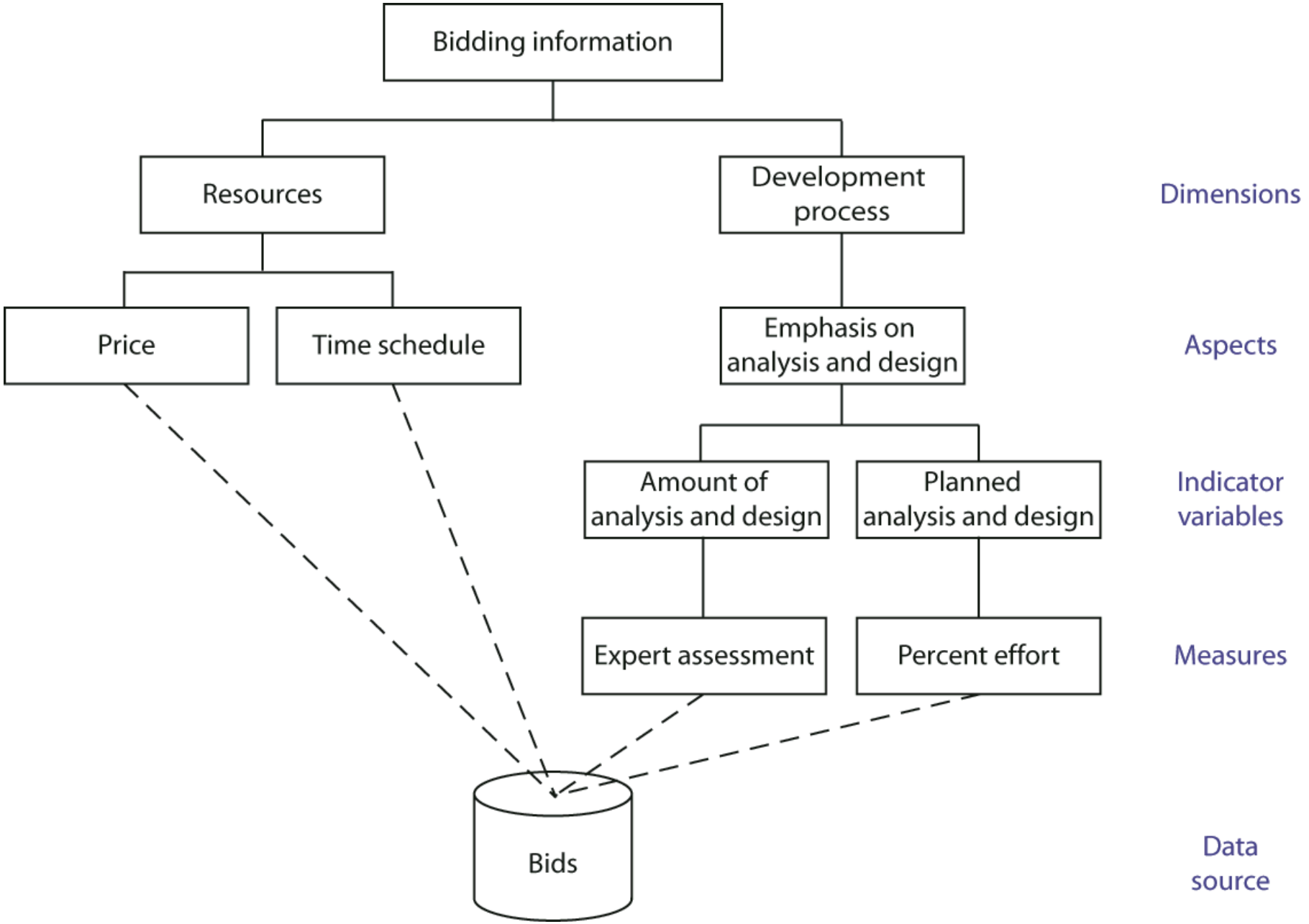
Preliminary theory:

1) Software clients tend to achieve better prices, when the requirement uncertainty perceived by the bidders is low.
2) Software clients should not request early price indications based on limited and uncertain information when the final bids can be based on more complete and reliable information

Part 2: Multiple-case study with controlled context*

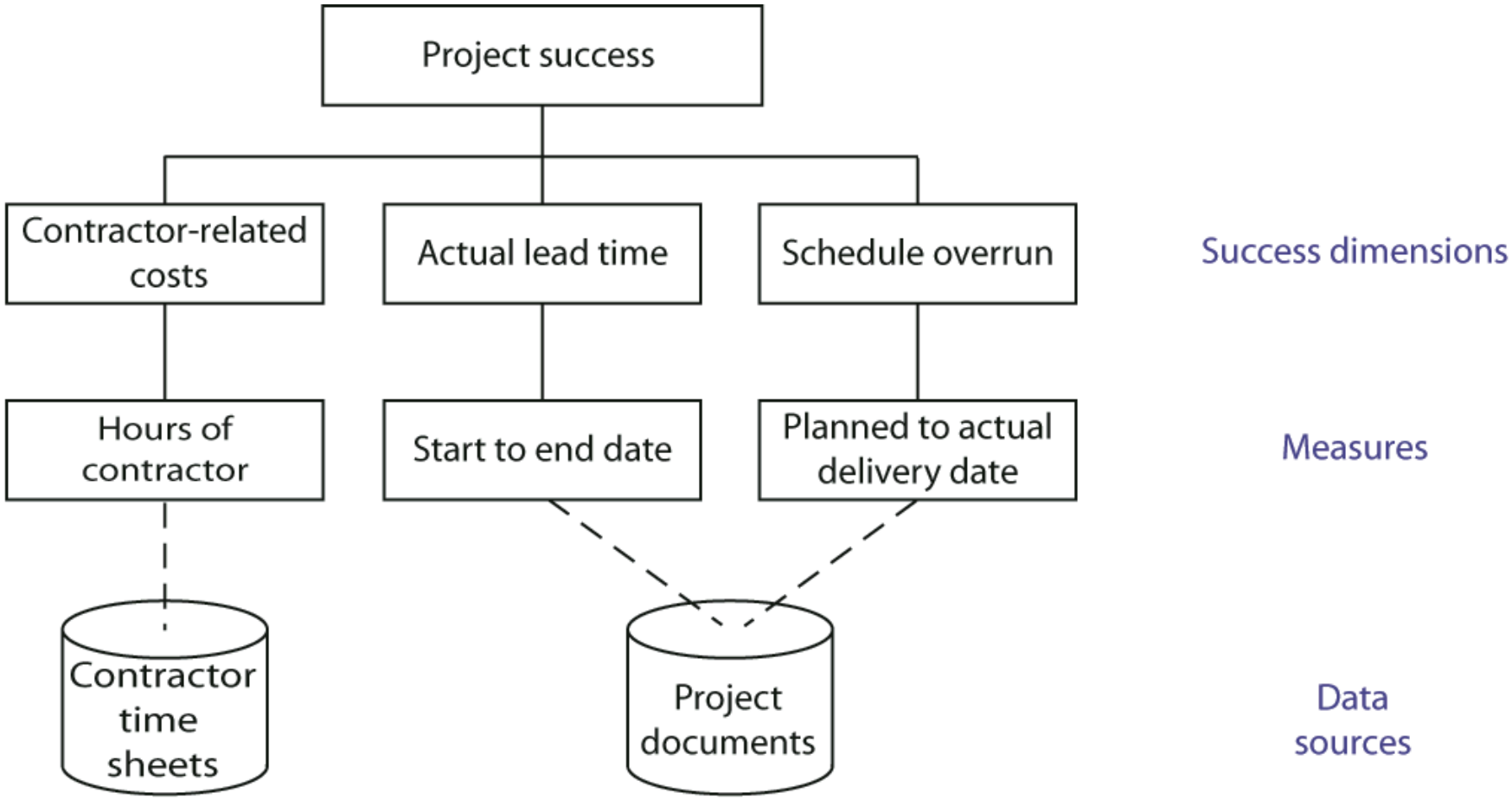


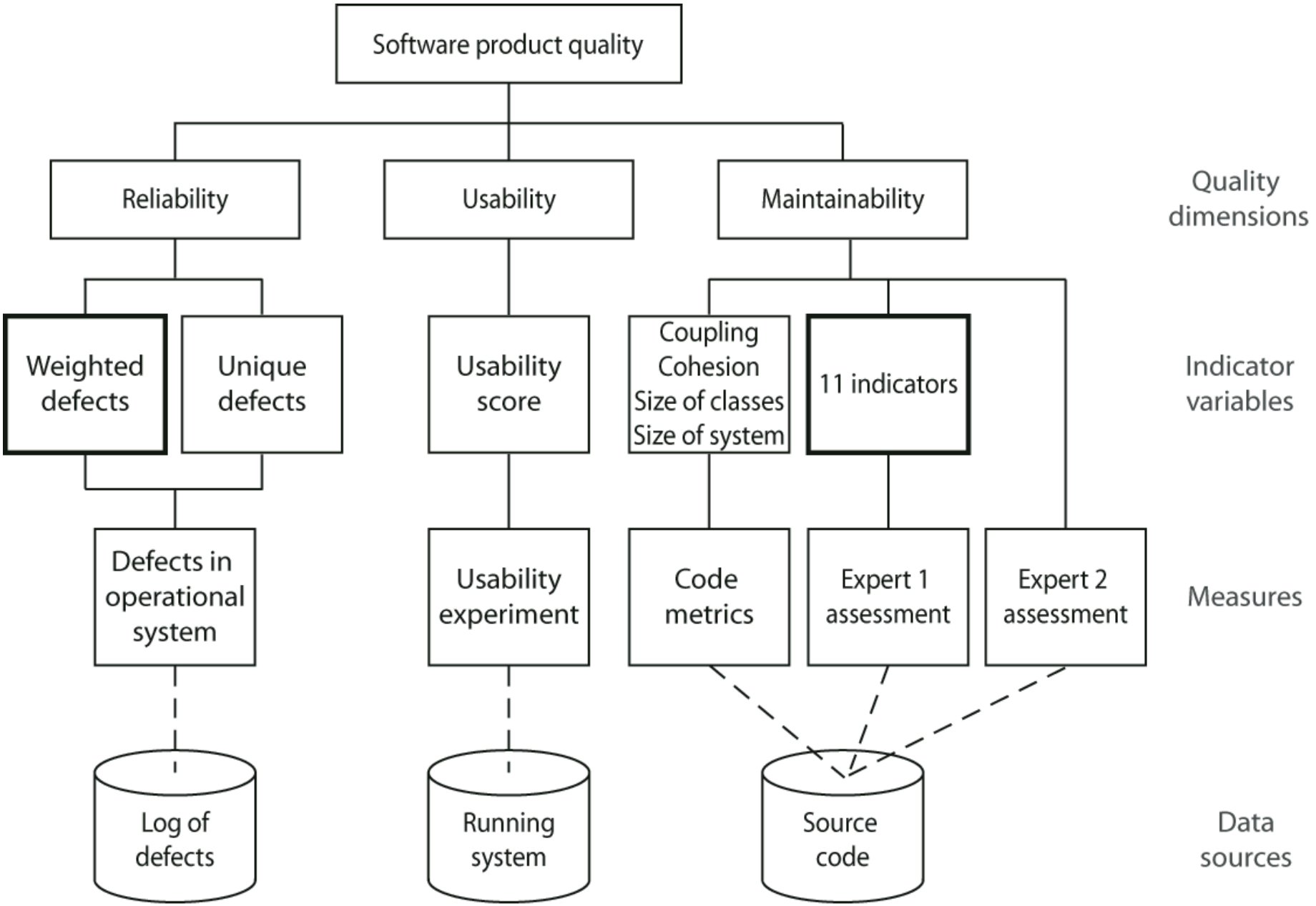
*B. Anda, D.I.K. Sjøberg, and A. Mockus. Variability and Reproducibility in Software Engineering: A Study of four Companies that Developed the Same System, IEEE Trans. on Software Engineering (to appear)


Bidding information
Resources
Development process
Price
Time schedule
Emphasis on analysis and design
Amount of analysis and design
Planned analysis and design
Expert assessment
Percent effort
Bids
Dimensions
Aspects
Indicator variables
Measures
Data source

The 4 companies

	Company A	Company B	Company C	Company D
Nationality	Norwegian	Norwegian	Norwegian	International
Ownership	Private	By employees	By employees	Listed on exchanges
Location	Oslo	Oslo	Bergen	Oslo + 20 countries
Size	Appr. 100	Appr. 25	Appr. 8	Appr. 13,000 worldwide
Firm price	€20,000	€45,380	€8,750	€56,000
Agreed time schedule	55 days	73 days	41 days	62 days
Planned effort on A&D	28%	20%	7%	23%


Project success
Contractor-related costs
Actual lead time
Schedule overrun
Hours of contractor
Start to end date
Planned to actual delivery date
Contractor time sheets
Project documents
Success dimensions
Measures
Data sources


Software product quality
Reliability
Usability
Maintainability
Quality dimensions
Weighted defects
Unique defects
Usability score
Coupling
Cohesion
Size of classes
Size of system
11 indicators
Indicator variables
Defects in operational system
Usability experiment
Code metrics
Expert 1 assessment
Expert 2 assessment
Measures
Log of defects
Running system
Source code
Data sources

Quality of project and product

Dimensions		Comp. A	Comp. B	Comp. C	Comp. D
Project	**Contractor-related**	90 hours	108 hours	155	85 hours
	Actual lead time	87 days	90 days	79 days	65 days
	Schedule overrun	58%	23%	93%	5%
Product	**Reliability**	Good	Good	Poor	Fair
	Usability	Good	Fair	Fair	Good
	Maintainability	Good	Poor	Poor	Good

Construct validity – defining and measuring quality

- **Measure to understand − but need to understand to measure. What can be measured meaningfully in SE?**
- **For example: Quality = number of errors? And what kind of errors, found where, found when? Compared with what? What about functionality, usability, maintainability, etc.**
- **In our study, we were surprised by the lack of measures that we could use to compare the quality of the 4 systems**

"At Sheffield University, students formed multiple small teams that built systems for commercial clients [41]. For each client, several teams competed to build the system that the client judged to be the best. However, little information was reported on the actual quality of the resultant products and how quality was measured."

[41] M. Holcombe, T. Cowling and F. Macias, "Towards an Agile Approach to Empirical Software Engineering", *Proc. Workshop on Empirical Studies in Software Eng.*, pp. 33-48, 2003.

Organisation, separation of roles

Developer companies

Project Management

Customer/vendor roles

Contract management

Functional specifications

Research Scientists

Research questions

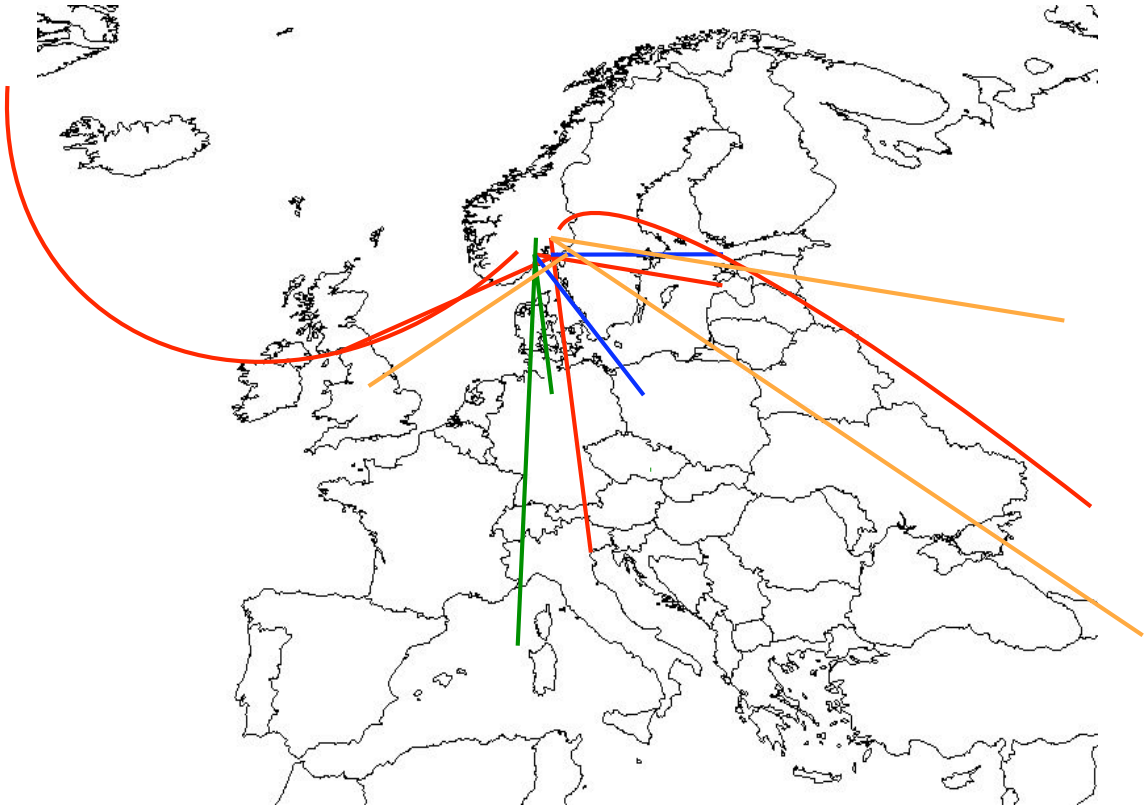
Data collection

Data source	Description
Access logs	The logs from Simula's web server were collected during the two years the systems were operational.
Bids	The companies' offered firm price was included in all the bids. The time schedule of the project, the planned development process, and the analysis and design of the product were included in many of the bids.
Contractor time sheets	Simula's contractor team recorded the time they spent on the project.
CVS	At the completion of the development and testing, the researcher team received complete CVS bases from the projects.
E-mail	All the e-mail communication between Simula's project manager and the development projects was recorded.
Interviews	The projects' team members were interviewed weekly about their work on the project and about the possible effects of being the object of research. The interviews were semi-structured and based on an interview guide in which some questions were the same each week, while others varied depending on the status of their project.
Issue tracker	The companies registered their questions and needs for clarification in Bugzero; see http://www.websina.com/bugzero/. The Simula contractor team registered their responses. Later on, the Simula team registered defects (classified according to severity) that were detected in the acceptance tests.
Log of defects	This is the log of the defects found after the systems became operational.
Project documents	These are documents related to overall project management, such as time schedules, design descriptions, acceptance test logs, and technical documentation.
Running Systems	The four systems.
Snapshots	Snapshots of all documents (including code) were sent to the research team weekly.
Source code	The source code of the four systems

Prerequisites for Simula's studies

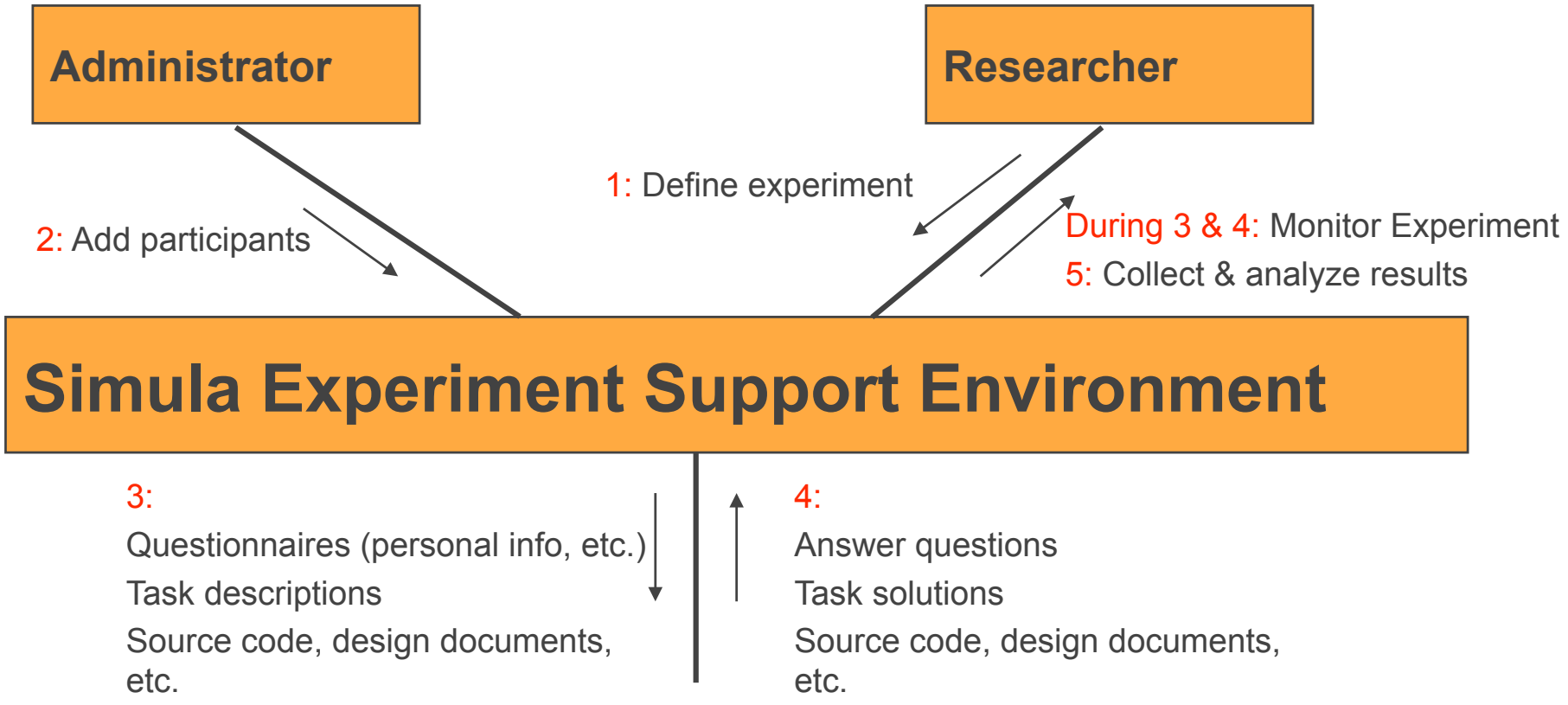
- Support environments
- Money

Empirical studies with professionals – a global activity



Country	Companies	People
Norway	216	3427
India	18	110
Russia	17	45
Sweden	14	161
Ukraine	7	20
Pakistan	7	14
UK	5	60
Romania	5	57
Nepal	4	101
Belarus	4	45
Bulgaria	4	8
Denmark	3	79
Vietnam	3	77
Germany	2	80
Ukraina	2	80
Poland	2	23
Czech Rep.	2	21
Moldovia	2	15
Italy	1	11
Finland	1	10
Lithuania	1	10
China	1	2
Phillipines	1	2
Serbia	1	2
Slovakia	1	2
Thailand	1	2
Canada	1	1
27	**326**	**4465**

Web-based tool support (SESE)



SESE is also used for surveys

Key functionality of SESE

- Real-time monitoring of the experiment
- Time is automatically taken
- Flexibility of defining new kinds of questions and measurement scales
- Automatic recovery of experiment sessions
- Automatic backup of experimental data
- Multi-platform support for downloading experimental materials and uploading task solutions

SESE is built on top of a commercial human resource management system, and is partly being developed by an external company

[E. Arisholm, D. I. Sjøberg, G. J. Carelius and Y. Lindsjørn. A Web-based Support Environment for Software Engineering Experiments, Nordic Journal of Computing 9(4):231-247, 2002.]

In the first 8 years of Simula, 262 companies from 24 countries have taken part with 2730 professionals in 75 experiments

How do Simula recruit professionals to take part in studies?

Type of study	Incentive	N companies	N persons
Experiments	Simula pay industry	158	1094
	Simula give seminar/increased knowledge	125	1636
Case studies	Simula pay industry	8	25
Action research/ case studies	Simula offer expertise, Research council pay Simula's time (40%), industry spend own time (60% of total costs) to improve business processes	8	83

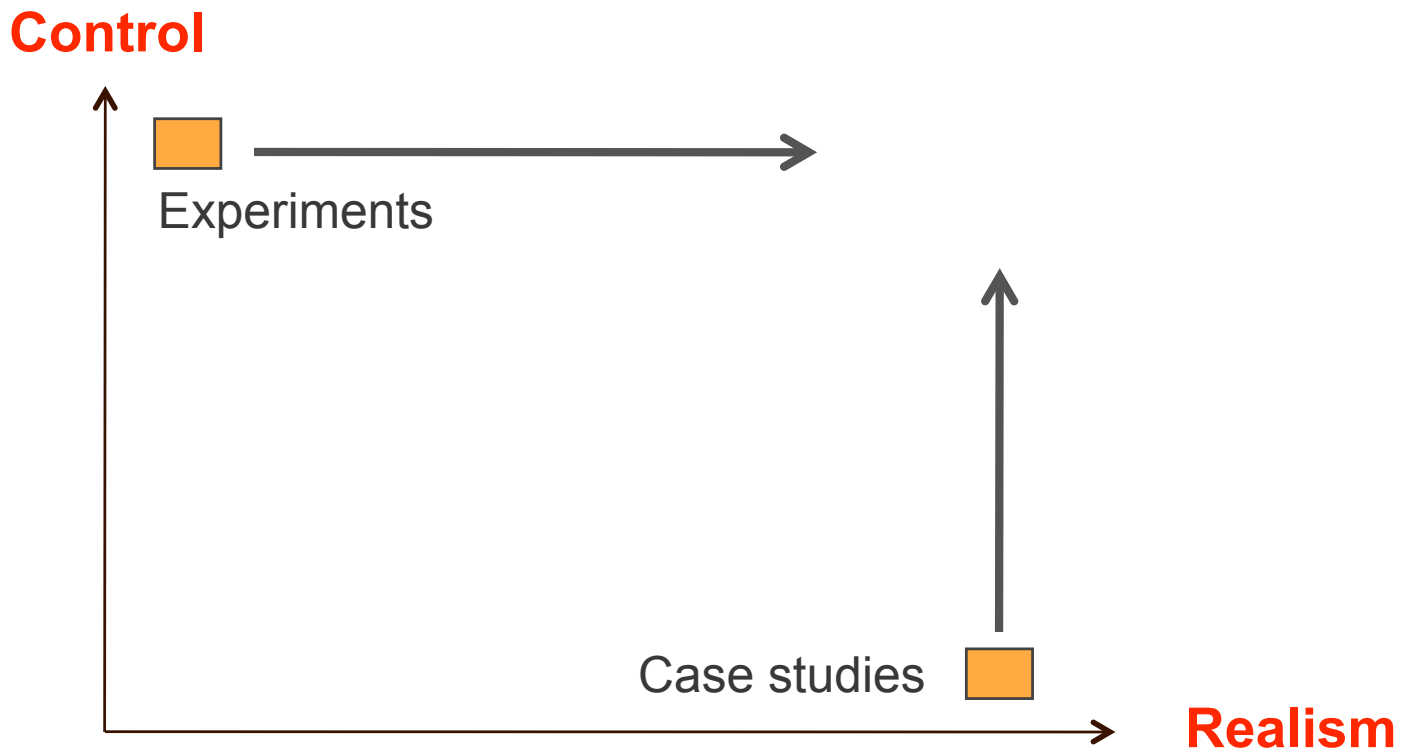
Hiring consultants

- Simula's experiments has cost up to €200,000
- We paid the companies ordinary consultancy fees for individuals or fixed price for a whole project, like any other ordinary customer.
  - The companies have routines for defining (small) projects with local project management, resource allocation, budgeting, invoicing, providing satisfactory equipment, etc.
- Difficult to find subjects employed in an in-house software development company because the management will typically prioritize the next release of their product

Large-scale empirical work requires a great amount of resources

- At Simula we used to spend about 25% of budget on empirical studies, including employing a professional project and data manager mainly at the expense of more researchers.
- In research grants applications, one budgets for money for positions, equipment and travel; why not include money for conducting empirical studies?
- Given the importance of software systems in society, why should research projects in SE be less comprehensive and cost less than large projects in other disciplines, such as physics and medicine? The U.S. funding for the Human Genome Project was $437 million over 16 years. If related activities are included, the total cost rises to $3 billion! CERN's annual budget is about $800 million.
- An ambitious, long-term goal would be to establish a research programme in SE similar to the Human Genome Project.

Conclusion



We have made some progress on how to evaluate which software development technologies are useful when, but joint effort in the SE community is needed to further increase the realism regarding subjects, technology, tasks, and software systems. And we need more case studies and more control in them.