Scientific Publication in SE, Seminar 1:

Science, Engineering, Community, Sciento- and Bibliometrics

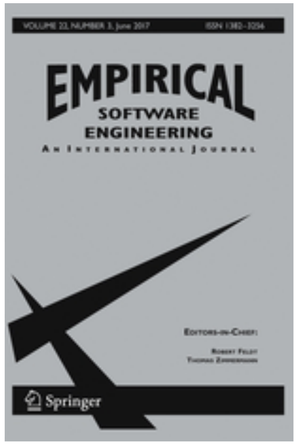
20231116, NTNU, Trondheim

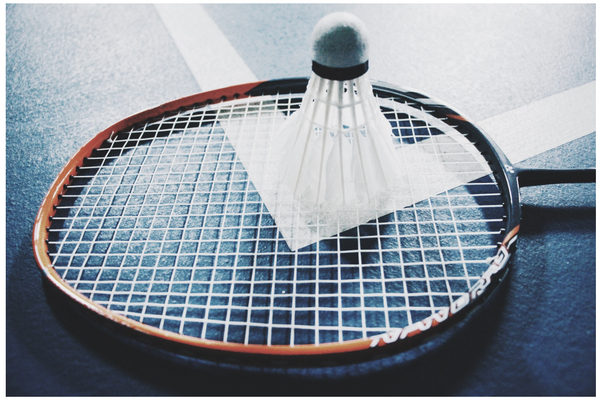
Robert Feldt

About me

CHALMERS
UNIVERSITY OF TECHNOLOGY

BLEKINGE TEKNISKA HÖGSKOLA · BTH ·





Course is focused on seminars and discussion to help the participants form their own views and answers, both on a teoretical and a practical level, to key questions like:

- What is a good scientific study in software engineering (SE)?
- What is a good SE paper?
- What is the difference between publishing in a journal and at a conference?
- Do different publication venues have different requirements and status?
- Where should I target my papers?
- How does review processes work?
- How do I review papers?
- What is this thing called science?
- Is software engineering a science?

The course is based on seminars to which PhD students prepare and after which they write down their views in (a) brief report(s). The idea is to give hands-on practical advice and show examples of how the software engineering research community functions. Since the course is an initial course, limited in size and scope, we go broad rather than in depth. But, at least, participants get a grounding in key activities and practices and can then go deeper as they see fit.

Course goals

After completion of the course the participants are expected to be able to:

- Read SE research papers and describe their motivation, the methods used and the results achieved
- State the strong and weak points of an SE study, the threats to validity as well as how the study can be improved
- Identify the research paradigm and method used in a SE study and compare it to alternative methods
- Write good and constructive reviews of scientific papers in software engineering
- Apply review templates objectively even on their own research studies and papers
- Select relevant publication targets for their research
- Describe (strong) publication targets for SE in general as well as for their sub-area
- Write and update publication strategies for their research
- Define bibliometrics used to rank publication targets, journals and researchers, as well as their benefits and drawbacks

Trondheim course 2023-11-16

The course is organized as a two-day event with 2 seminars per day:

- Seminar 1: Science, engineering, community, and bibliometrics
- Seminar 2: Publication, it's processes and your publication strategy
- Seminar 3: Reading and reviewing papers
- Seminar 4: Research methods and validity threats

Ground Rules / Format

- We are colleagues having a discussion to broaden our views
- Everyone is active in discussions, everyone counts
- There is seldom a single (correct) answer
- No question is dumb; we learn by questioning, reflecting & sharing
- Format:
  - Initial discussion on prep questions
  - Robert shares some slides and we discuss the questions further as we go along
  - Polish is less important than content & repetition often good; slides are gonna be messy :)

Disclaimer

- I'll give examples based on my experience and my views
  - might not be the views of all faculty / your supervisor
- I'm sorry if I sometimes come across as cynical
  - sometimes hard to stay "idealistic" after a work life in academia

A key divide: Ideal vs Pragmatic View

Acknowledgements

While this is based on courses I have given at Chalmers and BTH before I got most of it from others and colleagues over the years.

In particular I want to thank Daniel Mendez (prof at BTH/Sweden & at Fortiss in Munich/Germany) for his slides on science, Empirical SE & methods

What is science? What is engineering? Importance of community?

Let's discuss!

Prep questions

PQ1. What is science?

PQ2. How is science different from engineering?

PQ3. How is science different from a process improvement project in a company (for example, trying to find a better way to capture bugs found during early stages in software projects)?

PQ4. What kind of science is Software Engineering?

PQ5. Why is publication important in science?

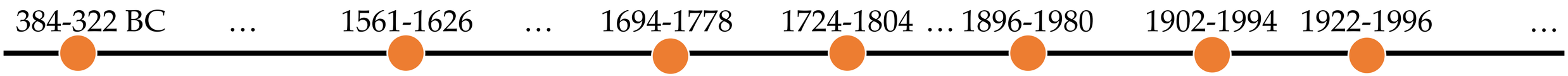
PQ6. Why is a scientific/research community central for scientific processes?

PQ7. What are the two most common bibliometric measures out there and how are they calculated?

“Science” wasn’t built in a day…

> Science is understood as the human undertaking for the search of knowledge (through systematic application of scientific methods)

→ Needs to be considered in a historical context
→ Increased understanding of scientific practice (and what science eventually is)



Aristoteles

- Search for laws and reasoning for phenomena
- Understanding the nature of phenomena

Bacon

- Progress of knowledge of nature (reality)
- Draw benefits from knowledge growth

Voltaire

- Emancipation from gods and beliefs

Kant

- System of Epistemology (theory of knowledge)

Piaget

- Era of constructivism

Popper

- Falsification as demarcation criterion
- Birth of null hypothesis testing

Kuhn

- Paradigm shifts
- Scientific Revolutions

“Science” wasn’t built in a day…

1922-1974 — **Lakatos**

- Competing Research Programs
- Progressive RP = predicts novel facts
- Degenerative RP = protects it’s core assumptions

1924-1994 — **Feyerabend**

- Against (single) Method
- Epistemological Anarchism (Pluralism & Creativity)

1941- — **Van Fraassen**

- Constructive Empiricism = Empirical Adequacy
- Focus on models
- Theories = family of models

1944- — **Cartwright**

- Pragmatic focus
- Causation as basis

How the laws of physics lie
N Cartwright, E McMullin
American Journal of Physics 52 (5), 474-476

1944- — **Longino**

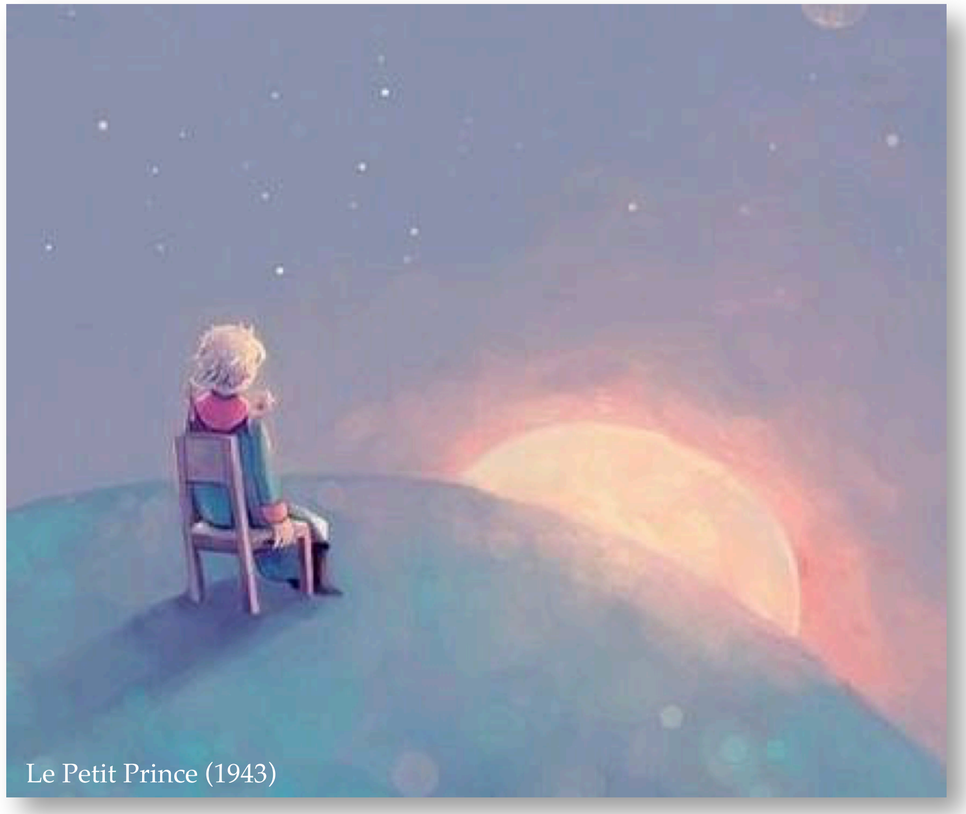
- Social Nature of Science/Knowledge
- Inclusive Objectivity

…

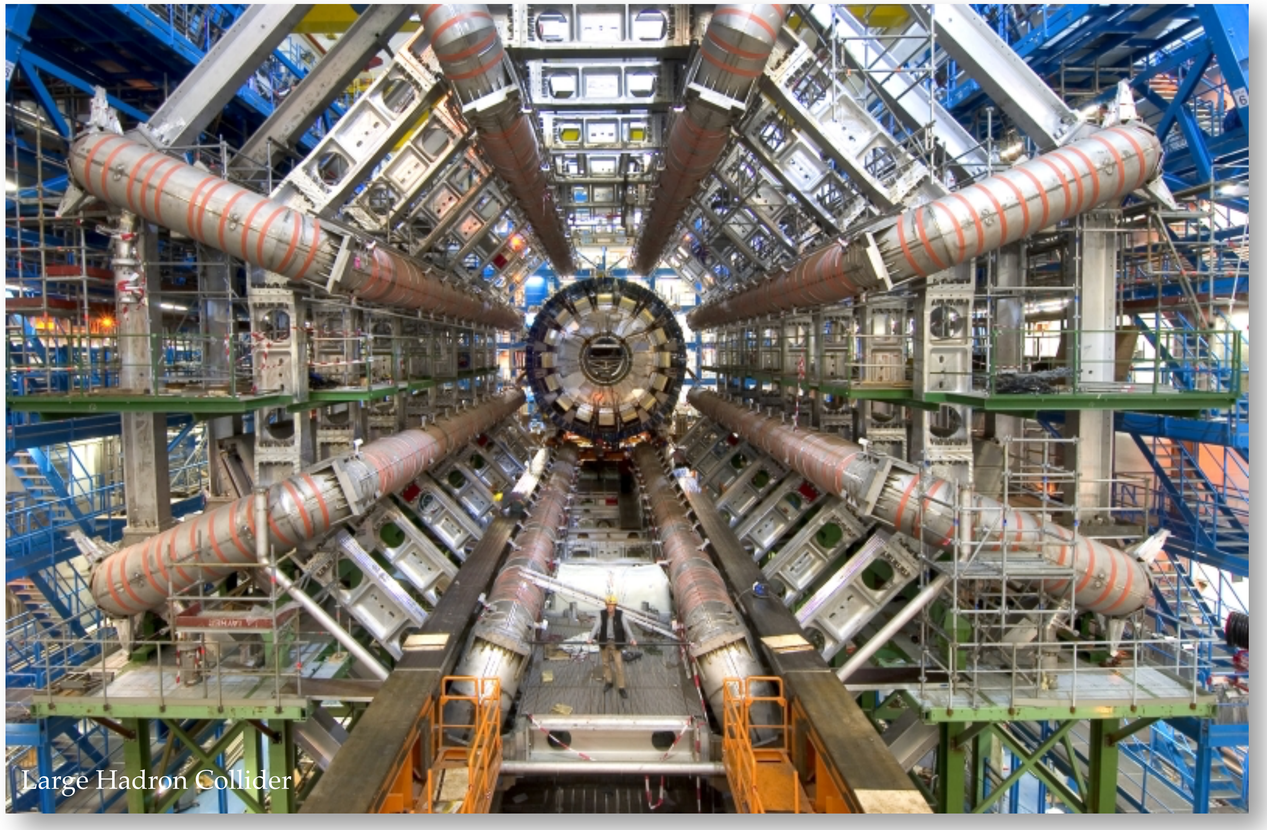
Scientific practices and research methods have changed over time, the role of empiricism* not

384-322 BC

Today



Le Petit Prince (1943)



Large Hadron Collider

* Gaining knowledge through sensory experiences

Scientific knowledge and practice

> Scientific knowledge is the portrait of
> our understanding of reality (via scientific theories).

Necessary postulates for scientific practice (selected):

- There are certain rules, principles, and norms for scientific practices
  - Rationalism: Reasoning by argument / logical inference / mathematical proof
  - Empiricism: Reasoning by sensory experiences (case studies, experiments,…)
- There is nothing absolute about truth
- There is a scientific community to judge about the quality of empirical studies

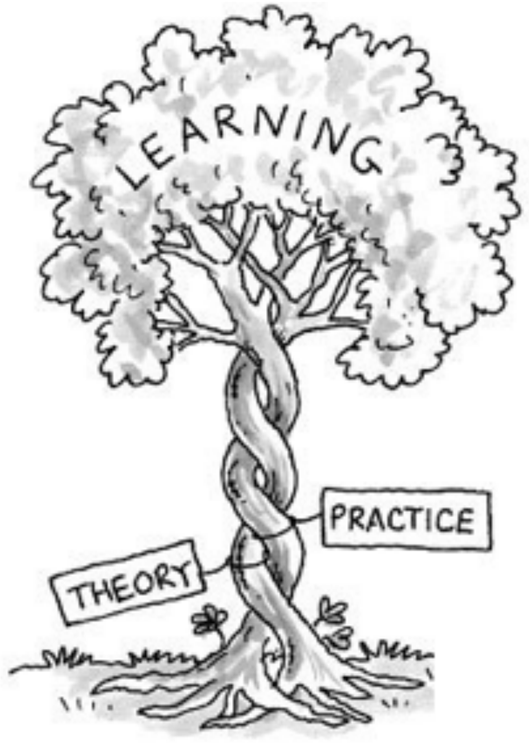
One view:

Empirical Software Engineering

> The ultimate goal of Empirical Software Engineering is to advance our body of knowledge by building and evaluating theories.

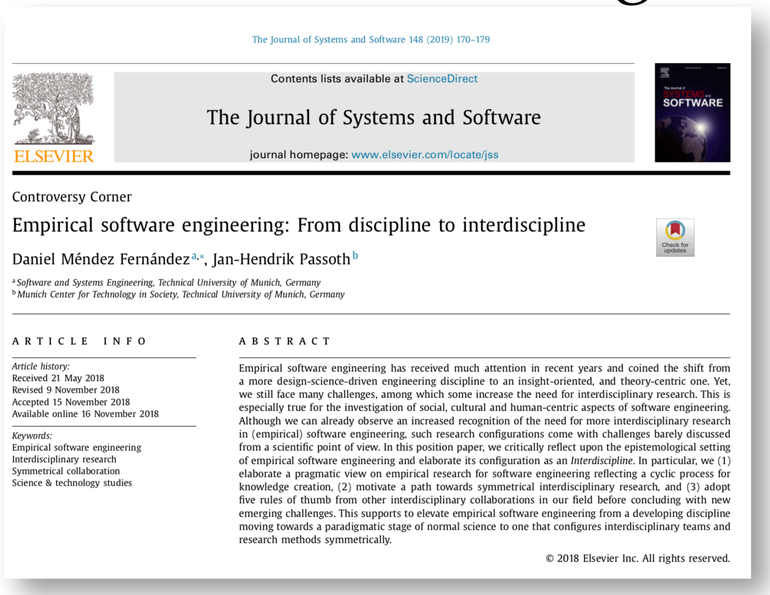
Relevance from a theoretical and practical perspective:

- Reason about the discipline and (e.g. social) phenomena involved
- Recognise and understand limitations and effects of artefacts (e.g. by evaluating technologies, techniques, processes, models, etc.) in their practical contexts



From real world phenomena to theories and back: The empirical life cycle

Further reading and outlook



The Journal of Systems and Software 148 (2019) 170–179

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

Controversy Corner

Empirical software engineering: From discipline to interdiscipline

Daniel Méndez Fernández[a,*], Jan-Hendrik Passoth[b]

[a] Software and Systems Engineering, Technical University of Munich, Germany
[b] Munich Center for Technology in Society, Technical University of Munich, Germany

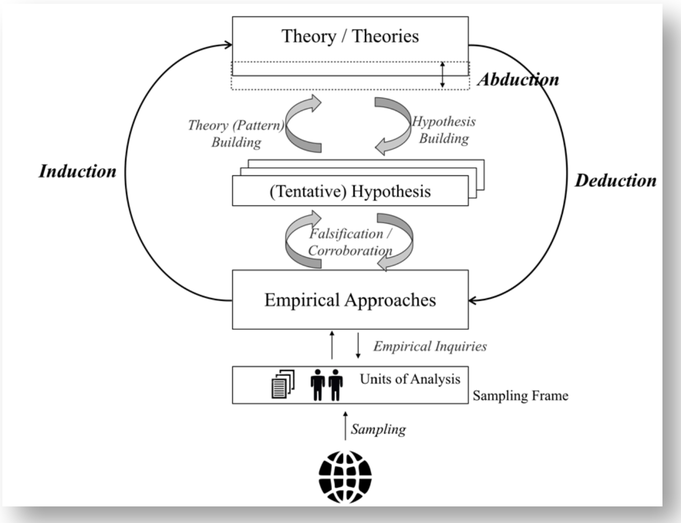
ARTICLE INFO

Article history:
Received 21 May 2018
Revised 9 November 2018
Accepted 15 November 2018
Available online 16 November 2018

Keywords:
Empirical software engineering
Interdisciplinary research
Symmetrical collaboration
Science & technology studies

ABSTRACT

Empirical software engineering has received much attention in recent years and coined the shift from a more design-science-driven engineering discipline to an insight-oriented, and theory-centric one. Yet, we still face many challenges, among which some increase the need for interdisciplinary research. This is especially true for the investigation of social, cultural and human-centric aspects of software engineering. Although we can already observe an increased recognition of the need for more interdisciplinary research in (empirical) software engineering, such research configurations come with challenges barely discussed from a scientific point of view. In this position paper, we critically reflect upon the epistemological setting of empirical software engineering and elaborate its configuration as an *Interdiscipline*. In particular, we (1) elaborate a pragmatic view on empirical research for software engineering reflecting a cyclic process for knowledge creation, (2) motivate a path towards symmetrical interdisciplinary research, and (3) adopt five rules of thumb from other interdisciplinary collaborations in our field before concluding with new emerging challenges. This supports to elevate empirical software engineering from a developing discipline moving towards a paradigmatic stage of normal science to one that configures interdisciplinary teams and research methods symmetrically.



Preprint: https://arxiv.org/abs/1805.08302

- Epistemological setting of Empirical Software Engineering
- Theory building and evaluation
- Challenges in Empirical Software Engineering

duction

(Creative) Synthesis of an explanatory case from a general rule and a particular result (observation)

Deduction

Application of a general rule to a particular case, inferring a specific result

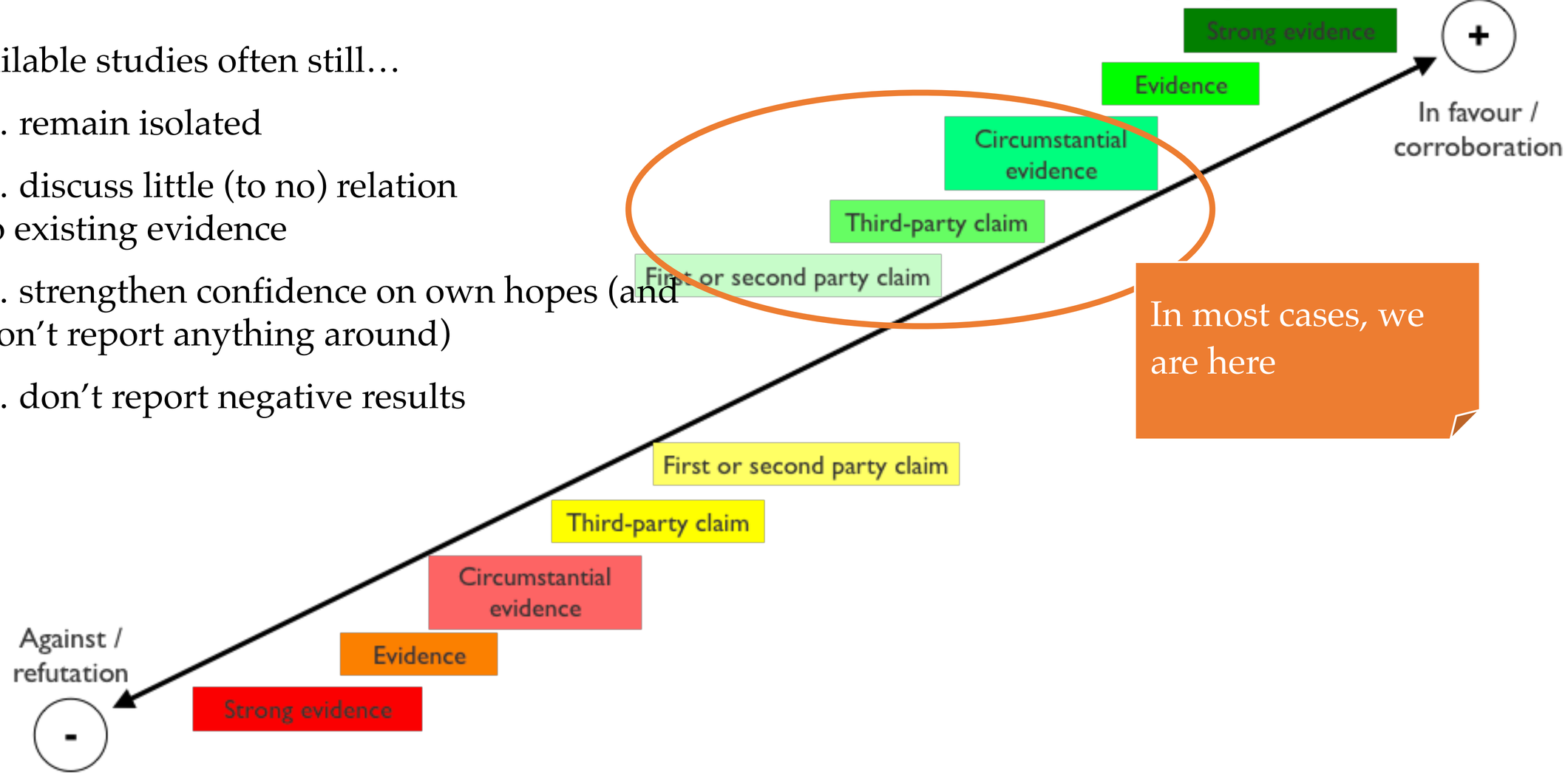
pling Frame

Source: Méndez and Passoth: Empirical Software Engineering: from Discipline to Interdiscipline, 2018.

Current state of evidence in Software Engineering

Available studies often still…

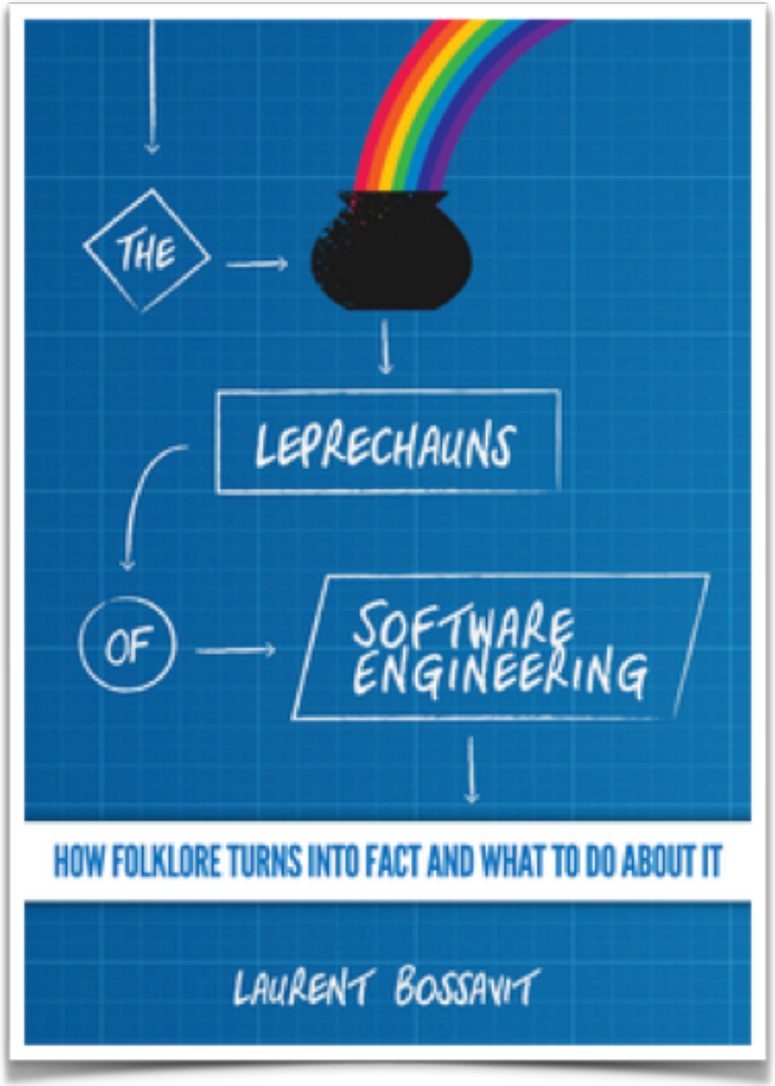
- … remain isolated
- … discuss little (to no) relation to existing evidence
- … strengthen confidence on own hopes (and don't report anything around)
- … don't report negative results



Source (levels of evidence): Wohlin. An Evidence Profile for Software Engineering Research and Practice, 2013.

Conventional Wisdom in SE

- Emerge from times where claims by authorities were treated as “facts”
- Reasons manifold:
  - Lack of empirical awareness
  - Neglecting particularities of practical contexts
  - Neglecting relation to existing evidence
  - No proper citations (one side of the medal or over-conclusions)
  - Lack of data
  - …



Exemplary conventional wisdom:
Go To statements considered harmful

1968

Edgar Dijkstra: Go T

Go To Statement Consider

Key Words and Phrases: go to
branch instruction, conditional
itive clause, program intelligibil
CR Categories: 4.22, 5.23, 5.24

EDITOR:

For a number of years I have be
that the quality of programmers
density of **go to** statements in the
recently I discovered why the use
disastrous effects, and I became
ment should be abolished from al
languages (i.e. everything except,
At that time I did not attach to
covery; I now submit my consid
in very recent discussions in whic
been urged to do so.

My first remark is that, altho
ends when he has constructed a
taking place under control of hi
matter of his activity, for it is this
the desired effect; it is this proces
has to satisfy the desired specificat
been made, the "making" of the
gated to the machine.

2015

An Empirical Study of Goto in C Code from GitHub Repositories

Meiyappan Nagappan[1], Romain Robbes[2], Yasutaka Kamei[3], Éric Tanter[2], Shane McIntosh[4], Audris Mockus[5], Ahmed E. Hassan[6]
[1]Rochester Institute of Technology, Rochester, NY, USA; [2]Computer Science Department (DCC), University of Chile, Santiago, Chile; [3]Kyushu University, Nishi-ku, Japan; [4]McGill University, Montreal, Canada; [5]University of Tennessee-Knoxville, Knoxville, Tennessee, USA; [6]Queen's University, Kingston, Ontario, Canada
[1]mei@se.rit.edu, [2]{rrobbes, etanter}@dcc.uchile.cl, [3]kamei@ait.kyushu-u.ac.jp, [4]shanemcintosh@acm.org, [5]audris@utk.edu, [6]ahmed@cs.queensu.ca

ABSTRACT

It is nearly 50 years since Dijkstra argued that `goto` obscures the flow of control in program execution and urged programmers to abandon the `goto` statement. While past research has shown that `goto` is still in use, little is known about whether `goto` is used in the unrestricted manner that Dijkstra feared, and if it is 'harmful' enough to be a part of a post-release bug. We, therefore, conduct a two part empirical study - (1) qualitatively analyze a statistically representative sample of 384 files from a population of almost 250K C programming language files collected from over 11K GitHub repositories and find that developers use `goto` in C files for error handling (80.21±5%) and cleaning up resources at the end of a procedure (40.36 ± 5%); and (2) quantitatively analyze the commit history from the release branches of six OSS projects and find that no `goto` statement was removed/modified in the post-release phase of four of the six projects. We conclude that developers limit themselves to using `goto` appropriately in most cases, and not in an unrestricted manner like Dijkstra feared, thus suggesting that `goto` does not appear to be harmful in practice.

Harmful [11]. This is one of the many works of Dijkstra that is frequently discussed by software practitioners [25] and researchers alike (more than 1,300 citations according to Google Scholar and almost 4000 citations according to ACM Digital Library as of Aug 15, 2014). This article has also resulted in a slew of other articles of the type *global variables considered harmful* [32], *polymorphism considered harmful* [24], *fragmentation considered harmful* [16], among many others. In fact, Meyer claims that as of 2002, there are thousands of such articles, though most are not peer-reviewed [15].

Indeed, Dijkstra's article [11] has had a tremendous impact. Anecdotally, several introductory programming courses instruct students to avoid `goto` statements solely based on Dijkstra's advice. Marshall and Webber [19] warn that when programming constructs like `goto` are forbidden for long enough, they become difficult to recall when required.

Dijkstra's article on the use of `goto` is based on his desire to make programs verifiable. The article is not just an opinion piece; as Koenig points out [7], Dijkstra provides strong *logical* evidence for why `goto` statements can introduce problems in software.

1969

O Considered Harmful"
lered Harmful

ost-noted item ever pub-
in *Communications* was a
from Edsger W. Dijkstra
d "Go To Statement Con-
d Harmful" [1] which at-
ed to give a reason why the
statement might be harm-
though the argument was
nic and unconvincing, its
ems to have become fixed
mind of every programming
er and methodologist. Con-
tly, the notion that the
is harmful is accepted al-
niversally, without question
bt. To many people, "struc-
programming" and "GOTO-
ogramming" have become
synonymous.

1987

"'GOTO Considered Harmful' Considered Harmful" Considered Harmful?

I enjoyed Frank Rubin's letter ("'**GOTO** Considered Harmful' Considered Harmful," March 1987, pp. 195–196), and welcome it as an opportunity to get a discussion started. As a software engineer, I have found it interesting over the last 10 years to write programs both with and without **GOTO** statements at key points. There are cases where adding a **GOTO** as a quick exit from a deeply nested structure is convenient, and there are cases where revising to eliminate the **GOTO** actually simplifies the program.

"We conclude that developers limit themselves to using goto appropriately, [not] like Dijkstra feared, [thus] goto does not appear to be harmful in practice."

- Public exchange based on reasoning by argument (rationalist arguments)...
- … finally challenged by one single empirical study.

[1] Edsger Dijkstra . Go To Statement Considered Harmful. Communications of the ACM, 1968.
[2] Frank Rubin. "GOTO Considered Harmful" Considered Harmful. Communications of the ACM, 1969.
[3] Donald Moore et al. " 'GOTO Considered Harmful' Considered Harmful" Considered Harmful?" Communications of the ACM, 1987.
[4] Nagappan et al. An empirical study of goto in C code from GitHub repositories, 2015.

State of Evidence

- The current state of evidence in Software Engineering is still weak
  - Practical relevance and impact?
  - Potential for transfer into practice and adoption?

- But there is hope…
  - Importance of empirical research recognised
  - Growth of a strong research community over last two decades

Science vs Engineering

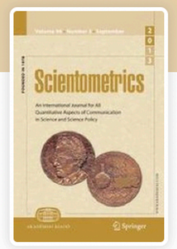
	Objectives	Methods	Outcomes	Nature of Work
Science	Understand World Discover & Explain & Predict Phenomena	Scientific Method Rigorous Testing & Validation of Ideas	New Knowledge & Understanding Published: Theories + Models + Findings	Exploratory & Theoretical Hypothesis & Experimentation Answer: How? Why?
Engineering	Create, Design, Build & Improve Implement Practical Solutions to Real Problems	Design process Consider constraints (Cost, Safety, Regulations, Env)	Tangible Tech, Systems, Tools, & Processes Serve practical purposes & Improve life	Project-based & Application-focused Specific Criteria & Needs

Definitions and background

- *Scientometrics* is the science of measuring and analyzing science.
- *Bibliometrics* is the science of measuring and analyzing publications.
- Scientometrics is often done by applying bibliometrics on scientific publications.

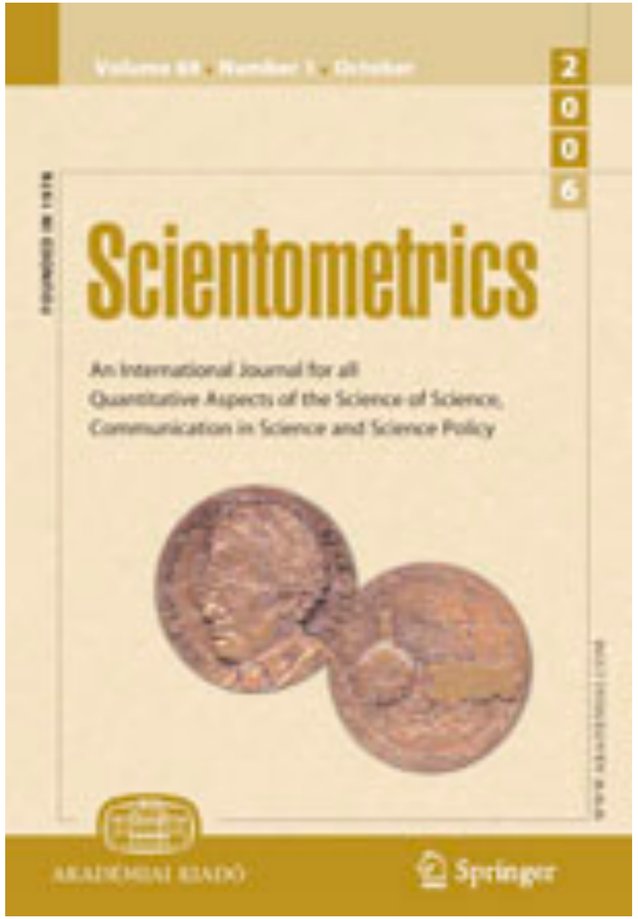
- There are of course scientific journals in this research area

Springer



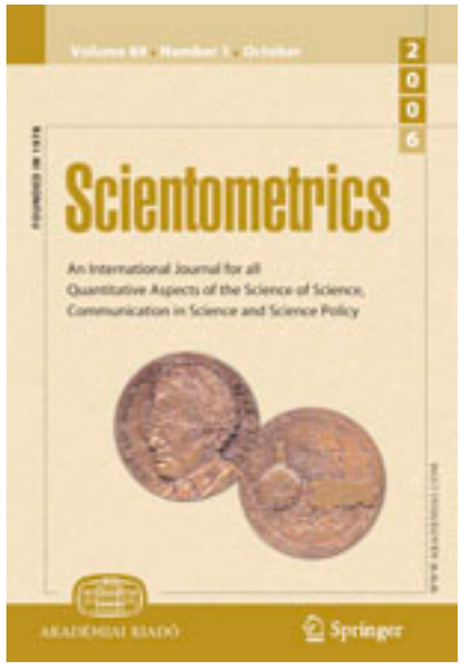
Scientometrics

An International Journal for all Quantitative Aspects of the Science of Science, Communication in Science and Science Policy



Example of papers in Scientometrics

- http://www.springerlink.com/content/0138-9130
- Bla: Claes Wohlin, "*A new index for the citation curve of researchers*", Journal of Scientometrics, Vol 81, Num 2, November 2009. ☺



Papers in Scientometrics 2012

- “A citation-analysis of economic research institutes”
- “Impact factor: Imperfect but not yet replaceable”
- “Driving factors of external funding and funding effects on academic innovation performance in university–industry–government linkages”
- “Creative accomplishments in science: definition, theoretical considerations, examples from science history, and bibliometric findings”

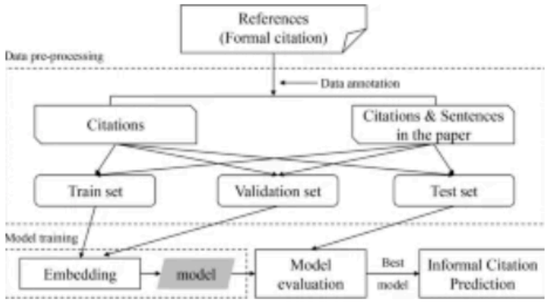
Papers in Scientometrics 2023

A study of BERT-based methods for formal citation identification of scientific data

Ning Yang, Zhiqiang Zhang & Feihu Huang

OriginalPaper | Published: 16 September 2023 |

Pages: 5865 - 5881



Do male and female authors employ different journal choice strategies?

Hayk Amirkhanyan, Michał Krawczyk & Maciej Wilamowski

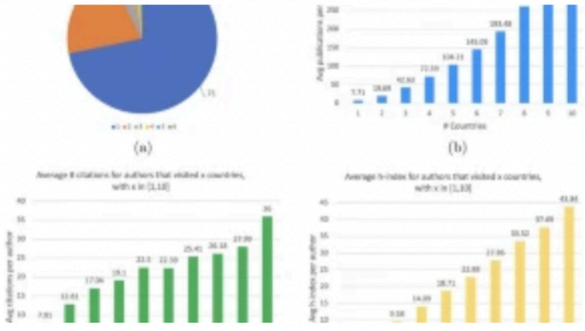
OriginalPaper | Open Access | Published: 10 October 2023 | Pages: 5905 - 5928

An analysis of international mobility and research productivity in computer science

Irene Finocchi, Andrea Ribichini & Marco Schaerf

OriginalPaper | Open Access |

Published: 25 September 2023 | Pages: 6147 - 6175



Motivation for metrics (University perspective)

- Several important reports during the last years:
  - ”Bibliometriska undersökningen”, Akademi Sydost
    - Used a number of different metrics and sources
  - ”Högskolerankingen”, Sydsvenska Industri- och Handelskammaren
    - One criterion (out of 8) was the number of publications (indexed by the ISI Web of Science)

Motivation for metrics (University perspective)

- ”Resurser för kvalitet”, Utbildningsdepartementet
  - Suggests that governmental research funds should be based on the quality of the research, measured by e.g. the number of citations
- Forskningsproppen 2012 and on
  - 10-20% of total research funding should be awarded based on “excellence”
  - Measured as both bibliometrics but also through expert assessment (international experts evaluate each major area)
  - Similar to UK’s REF which happens every 4-6 years

Motivation (University perspective)

- Swedish Science Council (VR) wants everyone to state number of citations for every listed paper, but also says:

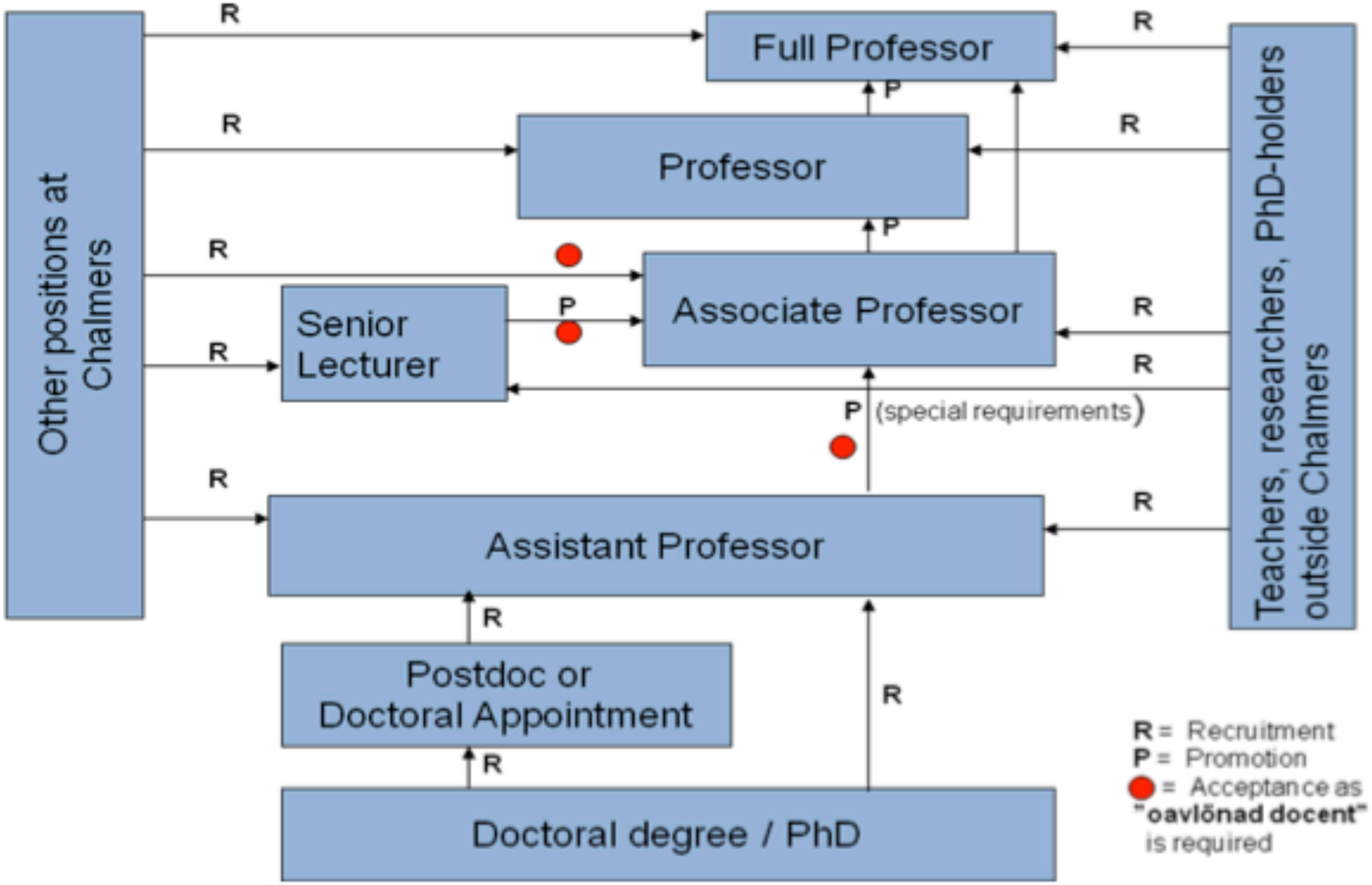
”Bibliometriska data, till exempel i form av det totala antalet citeringar för en sökande, bör användas med försiktighet i beredningsarbetet. Det har visat sig att sådana kan bli missvisande, om inte ett omfattande arbete med kvalitetssäkring utförts. Dock ska de sökande fr o m 2009 själva rapportera antalet citeringar för sina publikationer i ansökan.”

UK’s REF (Research Excellence Framework) often “inspiration"

Motivation (individual perspective)

- Better understanding of ”system”
  - How are researchers being evaluated/ranked?
  - How are journals and conferences ranked?
- Can be useful e.g. when
  - selecting where to publish your research results
  - improving the impact of your research
  - applying for positions
  - applying for project funding
- However, trend seems to be somewhat away from explicit use of metrics
  - But implicit use still abundant or even seen as the normal state!?

Motivation (individual perspective)



University appointment regulations for teaching and research faculty at Chalmers

2011

Motivation (individual perspective)

4.2 Fundamental qualification requirements

Scientific and pedagogical expertise is required for each academic position at Chalmers. Other experience and abilities are also required, depending on the needs of each department with regard to the subject content and responsibilities of the position. For some positions, the requirement for scientific expertise may be replaced by other specific alternative professional skills, for example artistic, pedagogical or leadership abilities, see Section 4.2.1 and Sections 6.1 and 6.2.

In addition to fulfilling the formal scientific and pedagogical qualifications, promotion requires compliance with *Chalmers' Strategies* with regard to responsibility and involvement, respect for others and equality of treatment.

University appointment regulations for teaching and research faculty at Chalmers

2011

Motivation (individual perspective)

4.2.1 Scientific expertise

Scientific expertise shall be demonstrated through one's own research and the planning and leadership of research. It can also be shown by the ability to achieve results through cooperation with other researchers inside or outside Chalmers. The leading international level of research in a specialist field shall be the reference point for assessment of scientific expertise.

In architecture and design, artistic expertise can be equivalent to scientific expertise.

University appointment regulations for teaching and research faculty at Chalmers 2011

Motivation (individual perspective)

4.2.2 Pedagogical expertise

Pedagogical expertise should be shown through one's own teaching and the ability to generate commitment and interest in the subject, to organise knowledge in well-structured and highly esteemed courses, to motivate students in their learning, and to communicate with students and other teachers. The abilities to hold a comprehensive view and to engage in renewal are valuable assets. Pedagogical expertise is founded on sound and extensive knowledge of the subject in question and a reflective attitude both towards one's own pedagogical work and student learning. Links with research in the subject are also important in relation to pedagogical expertise. Pedagogical qualifications should be documented in a Pedagogical Portfolio.

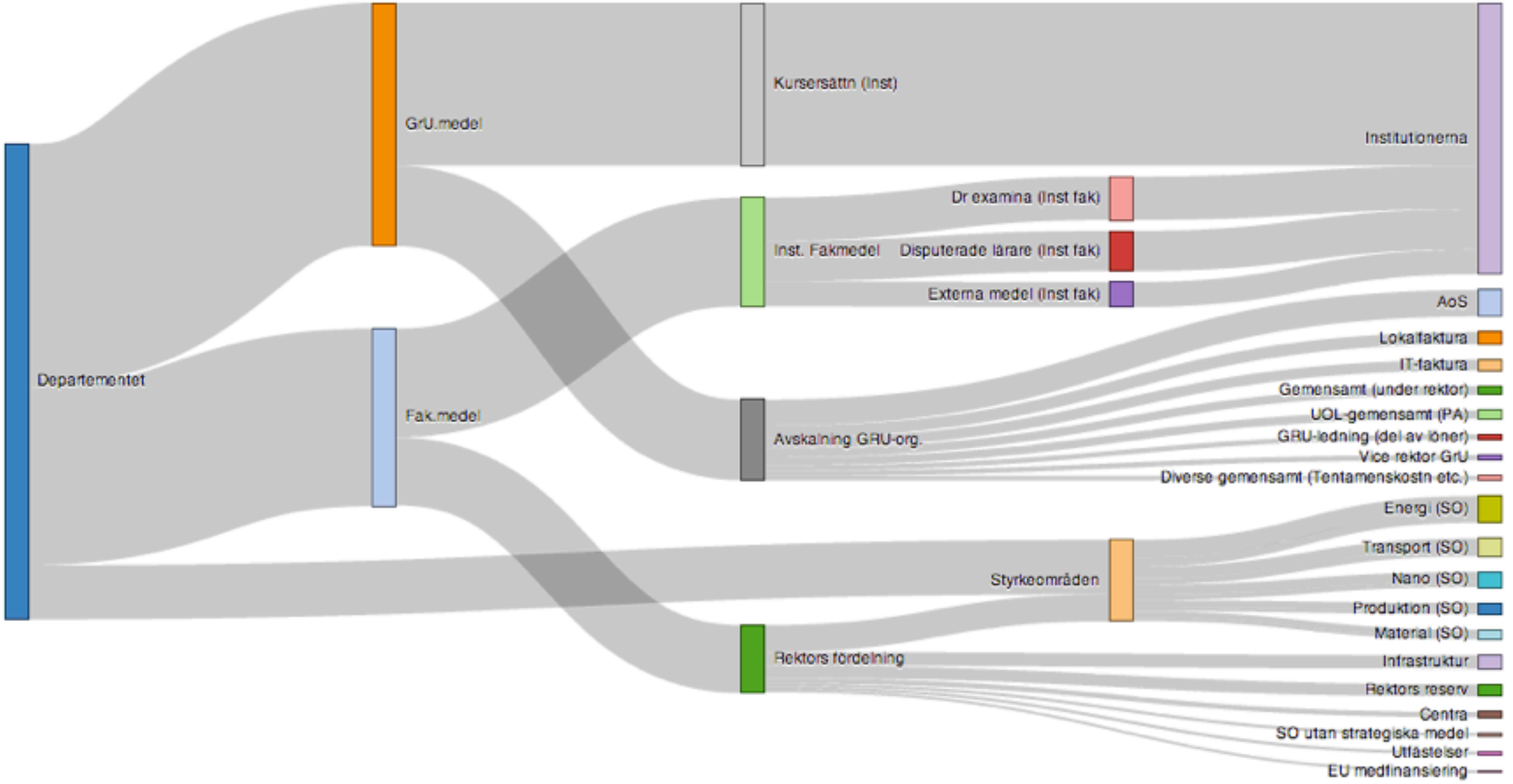
University appointment regulations for teaching and research faculty at Chalmers 2011

Motivation (individual perspective)

- Even internal money at Chalmers will be distributed partly according to success in publishing in the “right” fora
  - “per head” models are disappearing
  - “per (strong) paper” models are coming

Departementsmedel @ Chalmers

Oct 2, 2012



Metrics (scientists)

- Total number of publications
- Total number of citations
- Average number of citations per paper
- H-index (proposed by Hirsch in 2005)
  - An h-index of X means that a person have X publications that have X citations or more
  - This metric is useful because it discounts the disproportionate weight of highly cited papers or papers that have not yet been cited.
  - Combines *quantity* (number of publications) and *quality* (impact/citations to these publications)
- The mean number of citations to offline articles is 2.74, whereas the mean number of citations to online articles is 7.03!

  (Lawrence, 2001)

Main sources of bibliometric data

- ISI Web of Science (WoS)
  - only (a selection of) journals (about 9000)
  - favors areas like the medical and natural sciences
  - at the expense of areas where conference publications are important, like computer science although this has improved in later years
  - as well as, other areas which mainly publish books and/or reports, like the humanities
- Google Scholar (GS)
  - includes ”everything” available on the www
  - originally had problems with accuracy, but less of a discussion nowadays
  - many scholars first “goto”, despite much talk about “it shouldn’t be so”

Metric for publications: Impact Factor et al

- The average number of cites to articles in a journal/proceedings in the last 2 years (-1)
- PageRank (similar to Google's algorithm) is a recursive impact factor, to give citations from journals that have high impact greater weight than citations from low-impact journals
- The number of citations to papers in a particular journal does not really directly measure the true quality of a journal, much less the scientific merit of the papers within it.
- Cannot be compared directly between different research areas/disciplines. It depends on, e.g.:
  - the absolute number of researchers in the area,
  - the average number of authors on each paper,
  - the nature of results in different research areas,
  - citation habits between different disciplines, particularly the number of citations per paper
- Review/survey papers/journals always get a lot of citations
- Venue h-index (like GScholar's area pages) depend heavily on size of venue
- Luckily, trend seems to be away from venue-specific rankings
  - But many countries still have requirements on publication in specific "quality tiers"

Sources (Publications)

- Impact Factor
  - ISI Journal Citation Reports
  - CiteSeer Impact (discontinued)
  - www.eigenfactor.org (a PageRank-type measure, not updated since 2015)
- Acceptance rates
  - www.cs.ucsb.edu/~almeroth/conf/stats/ (networking confs though)
- Ranking / Prestige (not really metrics) often unofficial lists, e.g
  - Australia's CORE ranking: https://www.core.edu.au/conference-portal
    - Discontinued their journal ranking list in 2022
  - CSRankings (https://csrankings.org/#/index?all&us)
  - Journal Quality List, Econ+Management (https://harzing.com/resources/journal-quality-list)
  - "Finnish list" … (I'm sure there is a Norwegian one?!)

Sources for SE

- Tao Xie's "Software Eng. Conferences" and "SE Conf Map" (Not updated since 2015)
  - http://people.engr.ncsu.edu/txie/seconferences.htm
  - http://research.csc.ncsu.edu/ase/semap
- Australian Ranking of ICT Conferences
  - https://www.core.edu.au/conference-portal
    - Discontinued their journal ranking list in 2022
    - Many other countries rely/relied on their rankings though

Sources for SE

- Journal of Systems and Software (JSS) yearly ranking
  - Glass et al since 1996, 2011 one covers 2004-2008
  - 1.0 points if 1 au, 0.7/au for 2 au, 0.5/au for 3, 0.4/au for >3
  - Originally based on only 7 journals (TSE, TOSEM, JSS, SPE, EMSE, IST, IEEE SW)
    - But has evolved over the years, also including conferences nowadays
- Feldt had/has summary lists:
  - http://www.cse.chalmers.se/~feldt/advice/isi_listed_se_journals.html
  - Thomson Reuters threatened legal action
  - Alternative one not updated since 2016

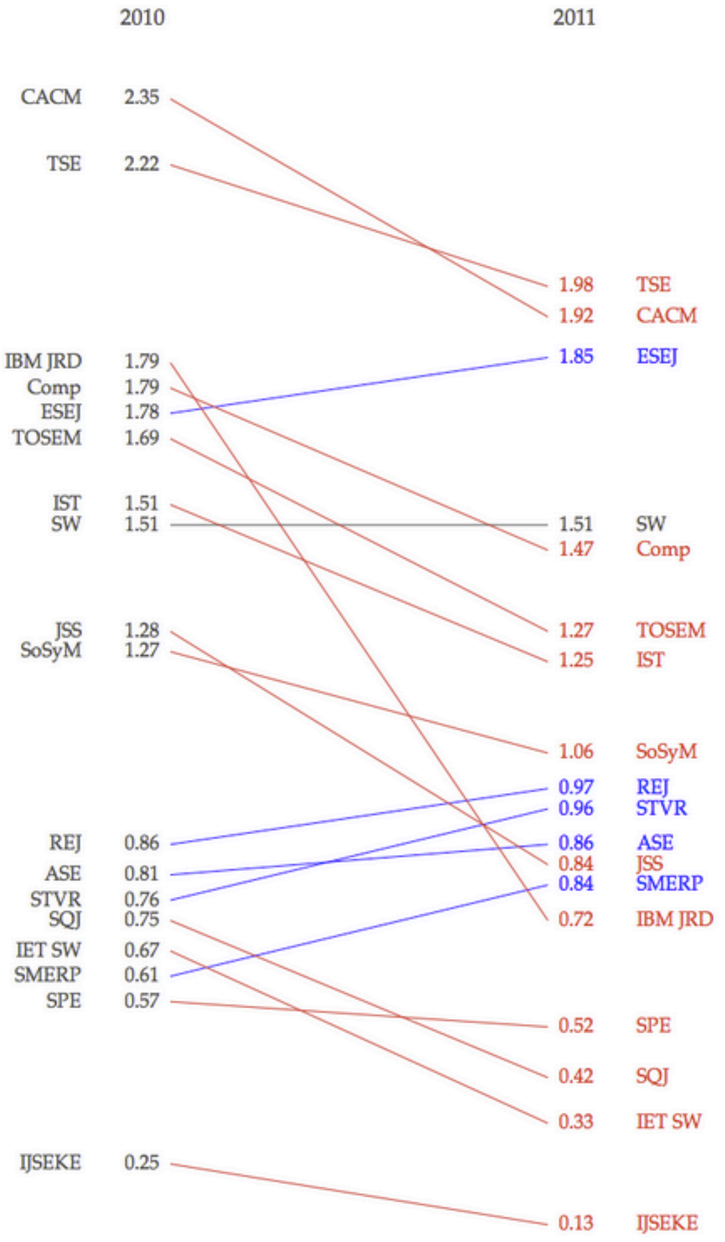
ISI JOURNALS - SOFTWARE ENGINEERING

Journal	11	10	09	08	JSS
1. IEEE Transactions on Software Engineering (TSE)	**1.98**	2.22	3.75	3.57	Yes
2. Communications of the ACM (CACM)	**1.92**	2.35	2.35	2.65	
3. Empirical Software Engineering (ESEJ)	**1.85**	1.78	1.61	1.09	Yes
4. IEEE Software (SW)	**1.51**	1.51	2.04	2.10	Yes
5. IEEE Computer (Comp)	**1.47**	1.79	2.21	2.09	
6. ACM Transactions on Software Engineering and Methodology (TOSEM)	**1.27**	1.69	2.03	3.96	Yes
7. Information and Software Technology (IST)	**1.25**	1.51	1.82	1.20	Yes
8. Software and Systems Modeling (SoSyM)	**1.06**	1.27	1.53	N/A	
9. Requirements Engineering Journal (REJ)	**0.97**	0.86	0.93	1.63	
10. Software Testing Verification & Reliability (STVR)	**0.96**	0.76	1.63	1.05	
11. Automated Software Engineering (ASE)	**0.86**	0.81	1.27	N/A	
12. Journal of Systems and Software (JSS)	**0.84**	1.28	1.34	1.24	Yes
13. Software Maintenance and Evolution - Research & Practice (SMERP)	**0.84**	0.61	1.14	0.97	
14. IBM Journal of Research and Development (IBM JRD, IBM Systems Journal was merged into this one in 2009)	**0.72**	1.79	1.29	1.88	
15. Software Practice & Experience (SPE)	**0.52**	0.57	0.67	0.71	Yes
16. Software Quality Journal (SQJ)	**0.42**	0.75	0.98	0.95	
17. IET Software (IET SW, was called 'IEE Proceedings - Software' before 2007)	**0.33**	0.67	0.65	0.54	

ISI JOURNALS - CS, TECHNOLOGY, MACHINE LEARNING ETC.

Journal	IF11	IF10	IF09	IF08
1. IEEE Transactions on Evolutionary Computation (TEC)	**3.34**	4.37	4.59	3.74
2. IEEE Transactions on Systems, Man & Cybernetics: B	**3.08**	2.67	3.01	2.36
3. Applied Soft Computing (ASOC)	**2.61**	2.08	N/A	N/A
4. Expert Systems with Applications	**2.20**	1.92	2.91	2.60
5. IEEE Transactions on Systems, Man & Cybernetics: A	**2.12**	2.08	2.03	2.08
6. IEEE Transactions on Systems, Man & Cybernetics: C	**2.01**	2.09	2.02	1.38
7. Machine Learning (JML)	**1.59**	1.96	1.66	2.33
8. IEEE Transactions on Reliability	**1.29**	1.29	1.33	1.32
9. IEEE Transaction on Dependable and Secure Computing (TDSC)	**1.14**	1.41	2.09	2.09
10. Advances in Engineering Software	**1.09**	1.00	1.05	1.19
11. Evolutionary Computation (EC)	**1.06**	2.63	3.10	3.00
12. Genetic Programming and Evolvable Machines (GPEM)	**1.00**	1.17	1.09	N/A
13. ACM Transactions on Programming Languages and Systems (TOPLAS)	**0.95**	1.17	0.87	1.44
14. Journal of Functional Programming (JFP)	**0.88**	1.37	1.47	0.98
15. Science of Computer Programming	**0.62**	1.28	1.46	1.27
16. Systems Engineering	**0.42**	0.79	0.58	N/A
17. IEICE Transactions on Fundamentals of Electronics, Communications...	**0.23**	0.29	0.37	0.44

Impact Factors for Software Engineering Journals



by Robert Feldt based on Thomson Reuters ISI data 2012-07-04

Questions to discuss

a, What are the most common bibliometric measures out there and how do they work?

b, What are the disadvantages/advantages of using such measures?

c, Does the usefulness of the measures differ between research areas? For example, is the medical area (as per the intro text from KI) different from Software Engineering?

d, Should I be aware of these measures in my own career/project? How?