What do We Know and How Well do We Know It?

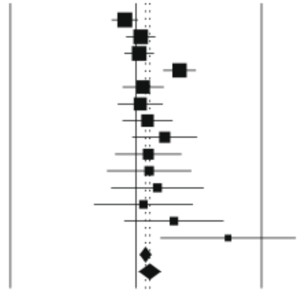
Current knowledge about Software Engineering practices

David Budgen

ICSOFT 2015

Durham University

My Agenda



Q1. How have our ideas about software engineering ‘knowledge’ been evolving?

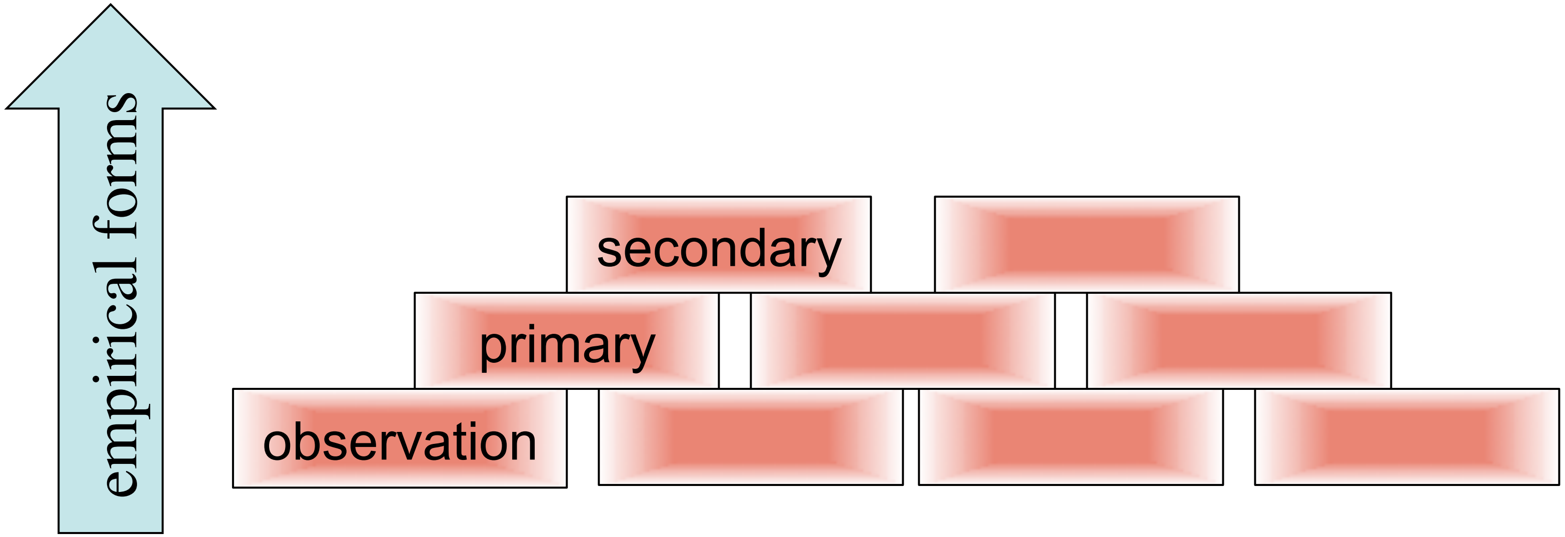
Q2. How has the adoption of evidence-based studies changed the nature and quality of that knowledge?

Q3. How well can EBSE (evidence-based software engineering) inform practice, teaching, and research?

Q4. What might we do to improve the quality of our knowledge?

Durham University

Q1. How have our ideas about SE knowledge been evolving?



The *scope* of SE knowledge

Software Engineering knowledge spans a wide range of forms.

- ❑ In one corner we are concerned with ‘process’ issues (management of projects, cost modelling, planning, lifecycles,…)
- ❑ In another we have very ‘techie’ aspects (testing, design, implementation, methods & tools,…)
- ❑ Also, many of our activities require human skill/knowledge (testing, design, programming,…)

As a result, teaching about this and deciding what practices to use for software development is apt to be ‘catalogue-based’ rather than using some ‘formal’ framework.

…and the knowledge itself

Our ‘catalogued knowledge’ still largely draws upon (informally codified) experiences of experts and so it:

- ❑ has limited underpinnings from theory
- ❑ embodies some design experiences
- ❑ is apt to be entangled with advocacy
- ❑ has ill-defined criteria for determining its limitations
- ❑ rarely uses empirical findings in any systematic way

Last point is evident from reference lists in textbooks and authoritative (catalogue-like) sources such as the SWEBOK (Software Engineering Body of Knowledge)

The position in 2000

In a series of studies, Glass, Vessey and Ramesh surveyed a range of publications in CS, IS and SE to identify the forms of research method used in these, and the reference disciplines (in terms of the use of these forms in other disciplines). Their work covered papers published in 1995-1999.

Next few slides draw heavily upon their findings (but focus mainly on those relating to SE).

Glass *et al.* – *Study Method*

Defined a classification system addressing:

- ❑ **topic** (subject matter of research)
- ❑ **research approach** (mainly formulative/descriptive)
- ❑ **research method**
- ❑ **reference discipline** (where did any theories come from?)
- ❑ **level of analysis**

Analysed 1485 journal papers from the 5-year period

Analysis was performed by two coders/paper, levels of agreement were in range 70-90%, and any differences were resolved after discussion

Research Method (part-table)

Research Method	CS	SE	IS
Conceptual Analysis	15.1%	**43.5%**	14.7%
Conceptual Analysis (Mathematical)	**73.4%**	10.6%	12.1%
Concept Implementation (proof of concept)	2.9%	**17.1%**	1.6%
Case Study	0.2%	2.2%	12.5%
Field Study	0.2%	<1%	**24.5%**
Simulation	1.8%	1.1%	1.4%

Durham University

From this, can conclude that in 2000 SE knowledge…

…was still based upon a culture of building things and using analysis rather than empirical evaluation…

…was codified in a range of forms, but few of these could be considered as ‘formal’ in any sense…

Which leads to the question: *what has changed*?

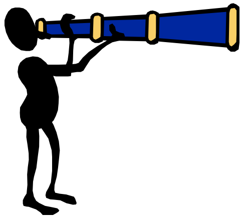
What has changed?

The last twenty years has seen a significant expansion in the use of *empirical techniques* in software engineering as well as the adoption of a wider range of forms for these.

This period has also seen the emergence of:

- ❑ Two specialist conferences (EASE and ESEM)
- ❑ A specialist journal (Empirical SE)
- ❑ A special section of another journal (IST) for systematic reviews

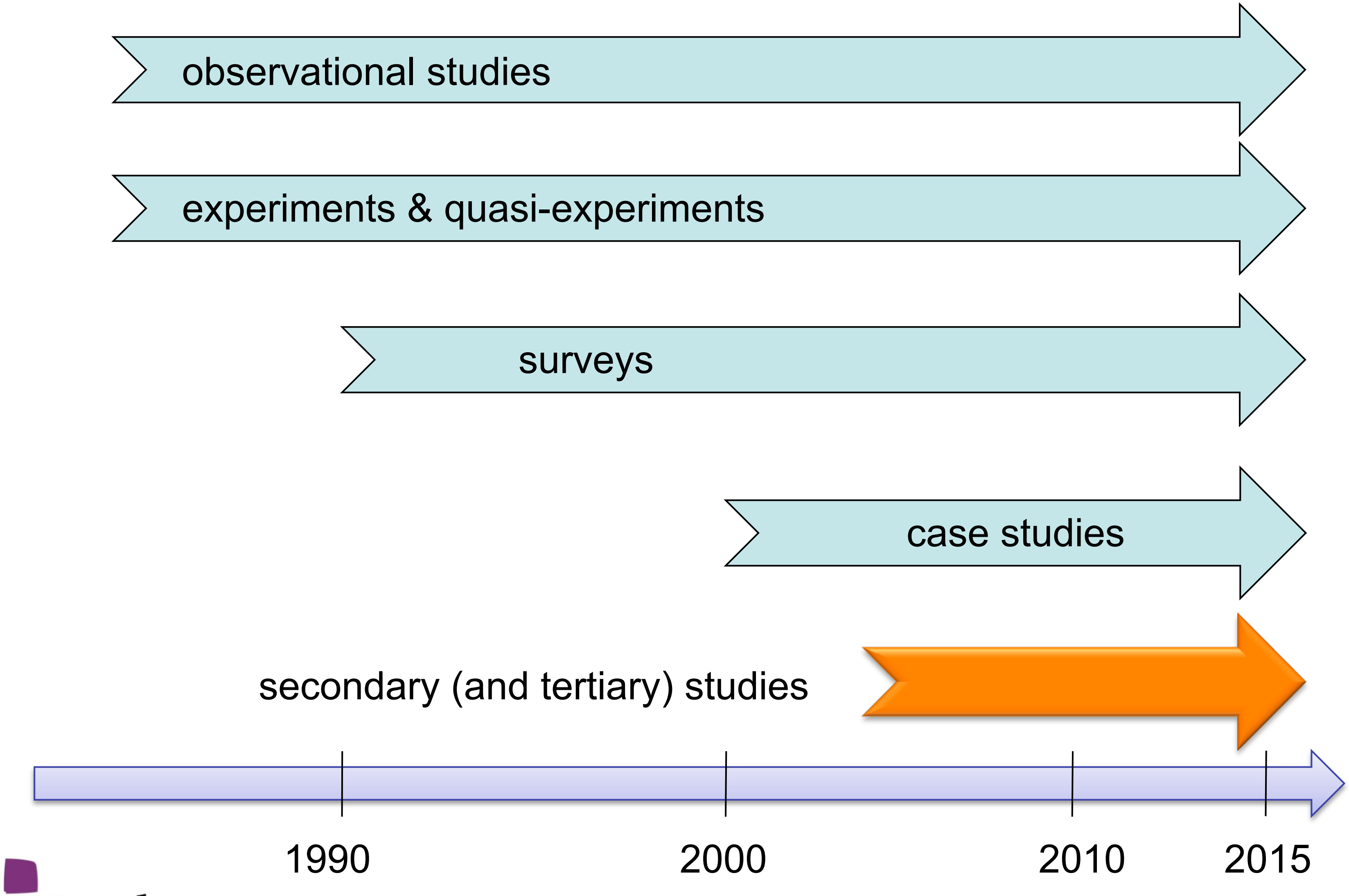
In particular (and the focus of this presentation), since 2004 the adoption of the *evidence-based* paradigm has provided a mechanism for aggregating software engineering knowledge.



Empirical: Relying upon observation and experimental investigation rather than upon theory.

Experimental: “A study in which an *intervention* (i.e. a *treatment*) is deliberately controlled to observe its effects” (Shadish et al., 2002)

An empirical time-line for SE


observational studies
experiments & quasi-experiments
surveys
case studies
secondary (and tertiary) studies
1990
2000
2010
2015

A1: In Summary

Q1. How have our ideas about software engineering ‘knowledge’ been evolving?

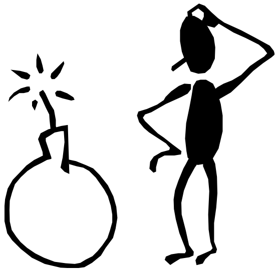
A1. New sources of knowledge have become available, based on more systematic ways of investigating how well our tools and techniques work, and in what circumstances.

From an empirical perspective, we also have begun to employ a wider range of empirical study forms, particularly the use of case studies and secondary studies.

Q2. How has the adoption of evidence-based studies changed that knowledge?

The evidence-based paradigm originated in clinical medicine, where it was originally it was seen as a way of improving teaching. It now forms a major influence upon clinical practice, as well as upon many other branches of health/social care, and has also been adopted in disciplines such as education and management.

Since 2004 researchers have been investigating how we can employ these idea for software engineering.



Durham University

Evidence-based practice…

…involves performing a secondary study that involves systematically finding, judging and synthesing the outcomes of *all* relevant (primary) studies of a treatment.

It offers the means of reducing the effects of the variability that naturally occurs in individual human-based studies, and of doing so systematically, in order to reduce bias that might be introduced by the researchers or by their choice of sources.

Variation

Natural Sciences	Any variation in the results of experiments tends to come from errors in measurement and so are usually small.
Humans as Recipients (Clinical RCTs)	We expect some ‘spread’ in the outcomes because humans vary physically, and in how they respond to a treatment.
Humans as Participants (Software Engineering)	We expect a large ‘spread’ in the outcomes because each person involved will have different abilities, skills, and experience, rather as we expect a class of students to have a wide range of marks on many modules.

Copyright Cochrane Logo goes here

- ❑ The logo of the *Cochrane Collaboration* illustrates the concept of pooling data, taken from a landmark study in New Zealand.
- ❑ The horizontal lines in the 'Forest Plot' represent the results from a series of 7 trials of an intervention used with pregnant mothers who were likely to give birth prematurely.
- ❑ Individually, only two of the studies showed statistically significant benefits from the treatment.
- ❑ The diamond at the bottom shows the result of a meta-analysis conducted 8 years later, with the aggregated data strongly indicating clear benefit from the intervention.

The 5 steps of EB practice

1. Convert information need into an answerable question.
2. Track down the best evidence relating to this in a systematic and unbiased way.
3. Critically appraise the evidence for validity, impact and applicability (usefulness).

systematic review

4. Integrate the critical appraisal with domain expertise and stakeholder requirements.
5. Evaluate outcomes and improve above steps.

knowledge translation

Secondary studies

For a *systematic review*, the protocol seeks to minimise bias by:

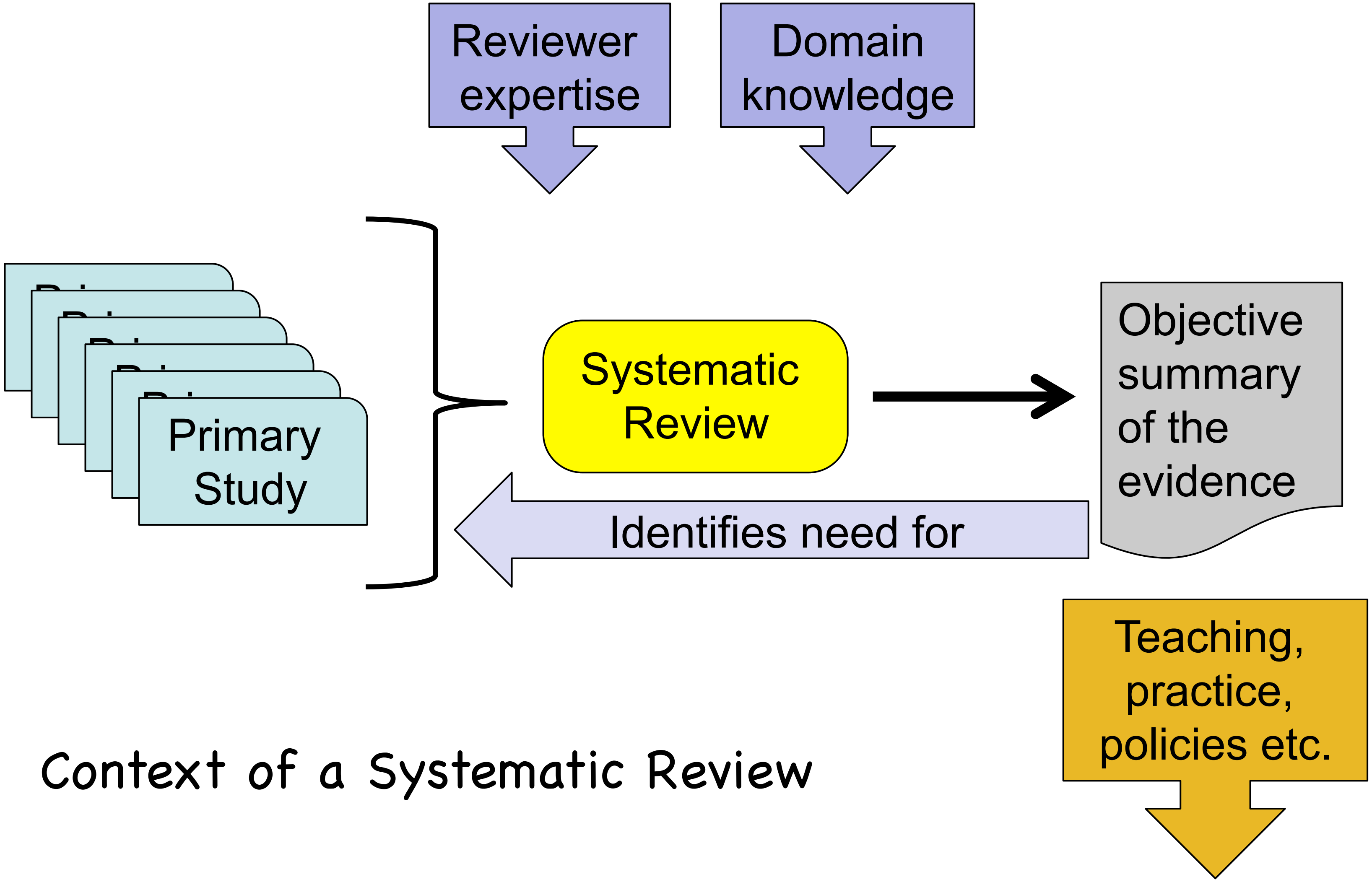
- ❑ specifying a well-focused *research question*
- ❑ Identifying a set of *keywords* for searching
- ❑ specifying how and where to *search* for source material (may be manual and/or electronic)
- ❑ providing clear *inclusion/exclusion* rules for selecting primary studies

A broader form of systematic review, termed a *mapping study*, can help identify where primary studies on a topic are 'clustered' and where there are 'gaps' in the issues covered.

Influence on primary studies

Secondary studies:

- ❑ encourage codification of the *reporting standards* for primary studies, since for effective aggregation they need to be able to extract data using some common basis
- ❑ help identify *where* further primary studies are needed, since they may ‘map the terrain’ around a given topic (indeed, this is one of the roles of a mapping study)
- ❑ provide the basis for *further studies* by raising new questions through their findings



Context of a Systematic Review

Further research: two examples

Our paper “Empirical evidence about the UML: a systematic literature review”, which found no support for the idea that the UML provided modelling forms that were better than others, or even that were useful, led to Marian Petre’s award-winning paper at ICSE 2013, reporting on interviews wih 50 experienced developers to find out how far they actually *used* the UML.

A study of what was known about effectiveness of design patterns (Cheng and Budgen, 2012) found little or no positive outcomes. We followed this up with two surveys to find out which GoF patterns were considered useful (hardly any), and why some were seen as particularly problematical.

A2: In summary

Q2. How has the adoption of evidence-based studies changed the nature and quality of that knowledge?

A2. EBSE practices make it possible to systematically find and ‘pool’ knowledge from individual studies, reducing any bias caused by individual variation.

Where the outcome from the primary studies reinforce each other they can also provide a sounder basis for guidelines regarding practice.

They can help identify where new research is needed.

Q3. How well can EBSE inform practice, teaching, and research?

This part of my talk draws heavily upon a ‘tertiary’ mapping study performed by four of us in 2011 (presented at ICSE 2012), and that we are currently updating.

Goal was to identify secondary studies that contained material (and ‘guidelines’) that would be meaningful for students taking an introductory software engineering module.

Durham University

The study

A ‘tertiary’ analysis of published systematic reviews. Original paper covered studies published to mid-2011 and we are now updating this to include those published to end 2014.

Team:

- ❑ David Budgen
- ❑ Pearl Brereton
- ❑ Sarah Drummond
- ❑ Nikki Williams

The sources

1. All of the studies identified in three broad tertiary studies (wide searches) covering publications up the end of 2009.
2. Systematic reviews (including mapping studies) published in five major software engineering journals 2010-2014 and found by manual searching:
   ① IEEE Transactions on Software Engineering
   ② Empirical Software Engineering
   ③ Information & Systems Technology
   ④ Journal of Systems & Software
   ⑤ Software Practice & Experience

Studies found

Overall, our manual search found 216 publications (after removing duplicate reports of studies)

Working in pairs for the initial sifting on title & abstract followed by checking the papers produced 59 usable studies – although we have also noted that many other studies contain material that could be useful for specialist/advanced teaching

We have also performed an electronic search for 2010-2014, which found a further 260+ papers. These are still to be analysed, although few look to be on mainstream topics.

Selection

Each paper was read by two people (assigned randomly) who were looking for:

- ❑ Any explicit recommendations about practice that would be useful for teaching about that topic
- ❑ Scope for a teacher to identify useful recommendations even when these were not extracted by the original authors
- ❑ Relevance to mainstream SE teaching

The outcomes were categorised using the SEEK (Software Engineering Education Knowledge) from SE2004.

Some observations

- Few authors provide explicit recommendations (strictly this is the role of ‘knowledge translation’ which is still immature)
- All major SEEK *knowledge areas* (KAs) are covered, although not evenly
- Relatively few studies address the more technical issues such as requirements and design, but there is quite good coverage of issues of software process and software management, which are not easily taught through models
- Even when results limited or have only weak significance, can help students appreciate lack of right/wrong answers for SE

Summary of SEEK coverage

SEEK KA	Title	#SRs	#KU without data/#KU
PRF	Professional Practice	2	1/3
MAA	Modelling & Analysis	9	3/7
DES	Software Design	3	4/6
VAV	Verification & Validation	9	2/5
EVO	Evolution	3	1/2
PRO	Software Process	11	0/2
QUA	Software Quality	6	2/5
MGT	Software Management	16	2/5

Some examples

These have been chosen to be illustrative of the range of topics and knowledge quality – and also to show how the outcomes might conflict with `expert opinion'. Look at three quite different topics:

① Requirements elicitation techniques

② Pair programming

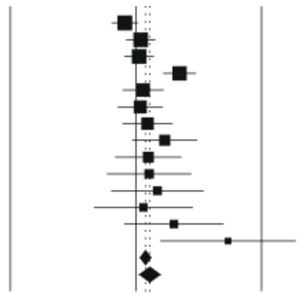
③ Needs of start-up companies

1. Requirements elicitation

SEEK KA	Title	Authors	Guidelines
MAA	Eliciting Requirements	Dieste & Juristo (2011)	**Topic:** Elicitation methods and their effectiveness. **Results:** Structured interviews are better than unstructured interviews; other more sophisticated techniques are no better (and some worse) than unstructured interviews; implying that structured interviews are best.

564 papers found → 26 used

2. Pair Programming



SEEK KA	Title	Authors	Guidelines
PRO	The effectiveness of pair programming: A meta-analysis	Hannay et al. (2009)	**Topic:** Effectiveness of pair programming versus 'solo' practice. **Results:** Some support for the use of PP where you need a solution that is of high quality, or where you need a quick one. (But can be less productive, and considerable heterogeneity in the primary studies implies other factors may be involved.)

236 papers found → 18 used

3. Development needs

SEEK KA	Title	Authors	Guidelines
MGT	Software Development in Startup Companies	Paternoster et al. (2014)	**Topic:** Software development under uncertain conditions (no history) **Results:** Some evidence to support use of light-weight methodologies as well as for using fast releases for prototyping (as a means of obtaining user feedback to help address issues of uncertainty).

1057 papers found → 43 used

A3. In Summary

Q3. How well can EBSE inform practice, teaching and research?

A2. There are now many published Systematic Reviews.

For *practice* (and policy-making), there are several studies that offer guidelines about when it may be useful to adopt particular techniques (such as pair programming), but more are needed.

For *teaching*, there are many studies that provide an overview of different topics, and that provide some ideas about what techniques are likely to be effective, and when.

For *research*, the outcomes from the reviews provide many opportunities to dig deeper.

Q4. What might we do to improve the quality of our knowledge?

These reflections mainly relate to primary studies, and come from a number of sources, including:

- ❑ A study of empirical quality performed by a team of experienced researchers, looking at reporting of experimental studies
- ❑ The experiences acquired from writing a book with Barbara Kitchenham and Pearl Brereton (“Evidence-Based Software Engineering & Systematic Reviews”)
- ❑ The tertiary study of teaching material described for Q3

My observations are partly anecdotal (!)

1. Is empirical quality improving?

A brief report of a study conducted by a team of seven experienced empirical researchers.

Published as:

“Trends in the quality of human-intensive software engineering experiments—A quasi-experiment” (2013). Barbara Kitchenham, Dag IK Sjøberg, Tore Dybå, Pearl Brereton, David Budgen, Martin Höst and Per Runeson, IEEE Transactions on Software Engineering, 39(7), pp1002-1017

The study

Examined 70 experimental and quasi-experimental papers published in four SE journals

- ❑ 1992-2002
- ❑ 2006-2010

Each paper assessed by three of us using:

- ❑ a questionnaire with 9 quality questions scored 0-3
- ❑ an overall subjective quality score in the range 0-3

Outcomes

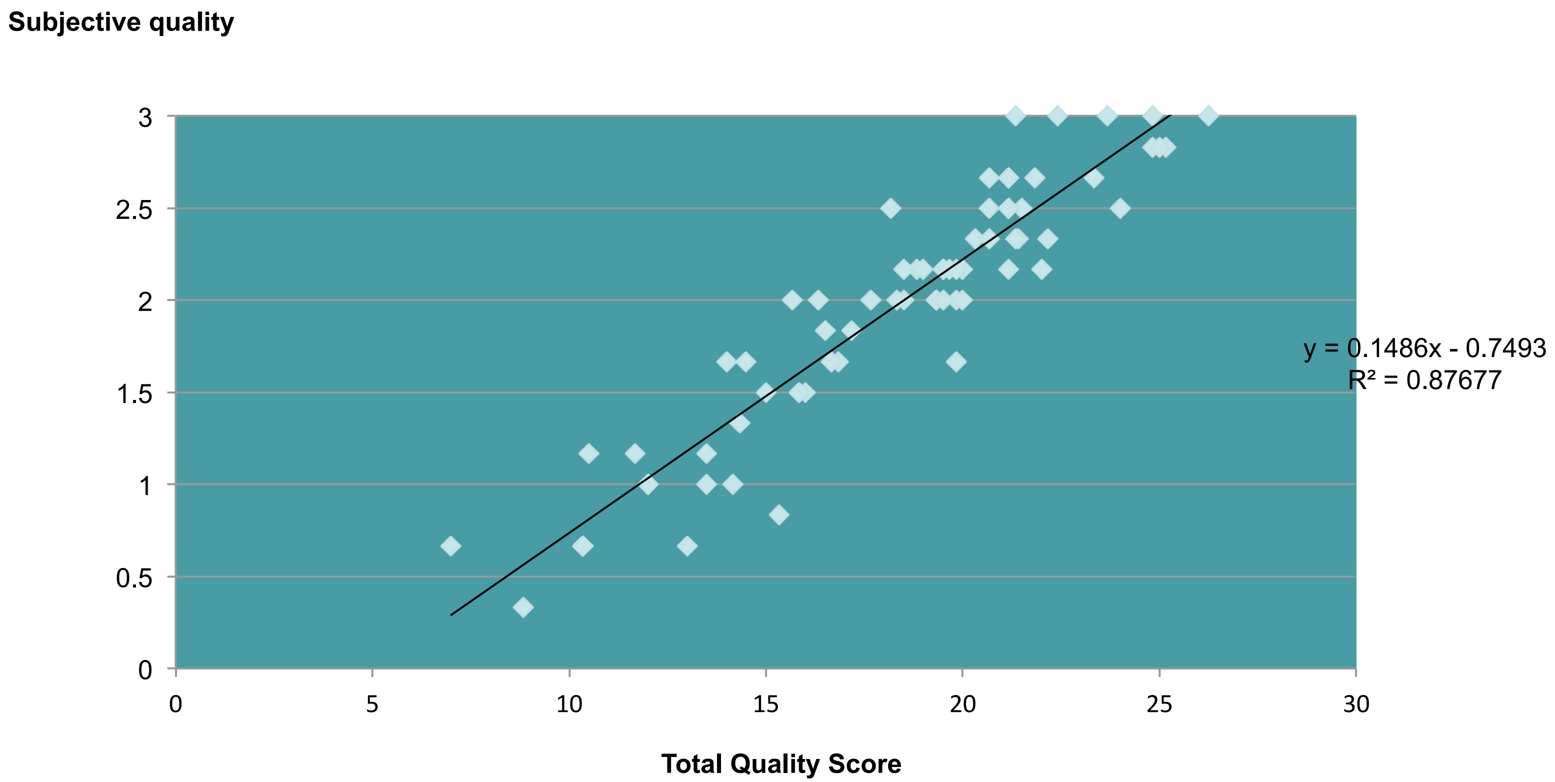
Observed a steady increase in quality over the period

Significant linear relationship between total quality score (9 questions) and subjective assessment

No indication that this increase in quality was directly caused by referencing the articles and texts on methodological issues written by SE researchers

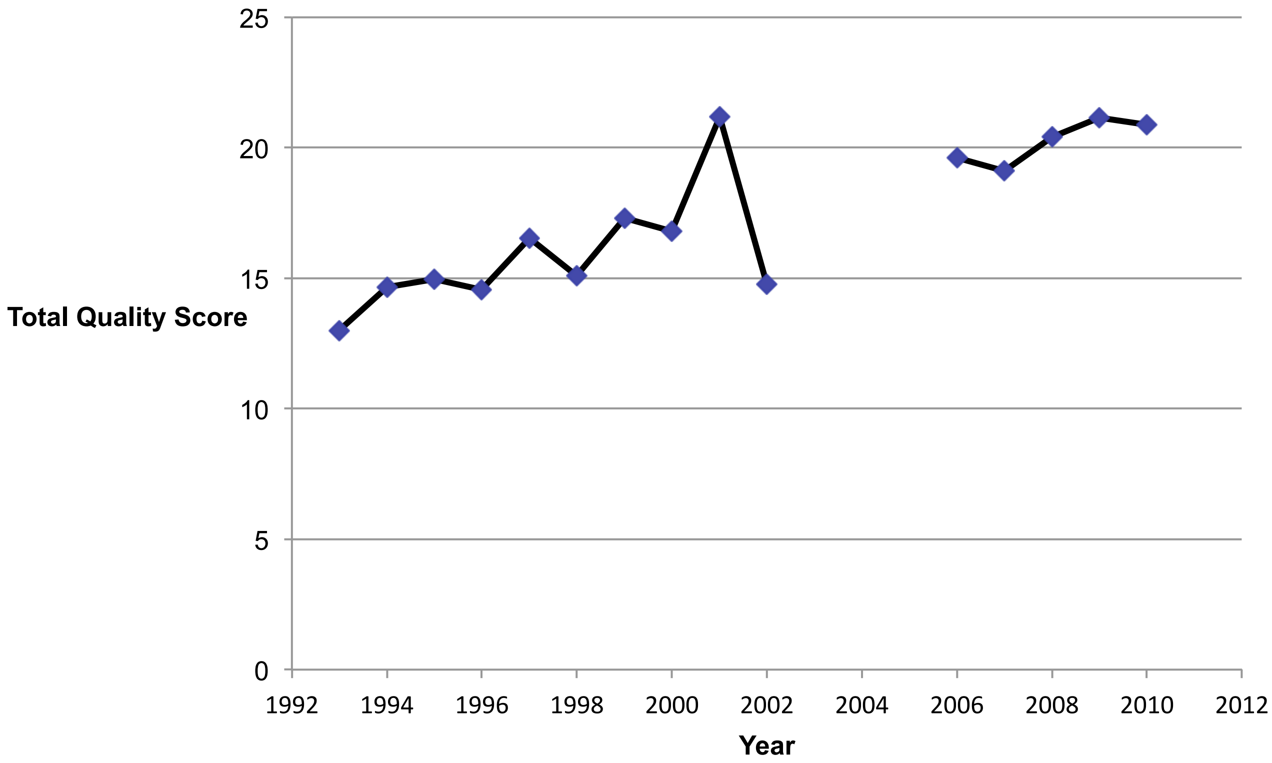
Durham University

The two quality measures



Durham University

Variation of quality with time



Possible Limitations

- May reflect quality of reporting, not of the study: which is why we used two different measures
- Experimenter bias from expectation: but 2001 scored higher than any other year
- Reliability of judges: previous studies a bit mixed, but this is why we used three judges/paper
- We took only one paper from each first author: might have biased first period
- Sources: we used only four (high quality) journals

2. Replication

For any study where humans are participants in some way, there is a question of how reproducible any results are.

Replication studies are widely used in most disciplines. The concepts introduced by Lindsay & Ehrenberg (1993) divide these into two roles:

- ❑ *Close replication:* seeking to keep most conditions as near the same as possible, and hence to see if a different group of participants confirms the results of the original study.
- ❑ *Differentiated replication:* Varying different aspects of the study to explore the boundaries of any effects observed.

Challenges for replication

- Replication in SE is difficult, and results apt to be inconsistent. A systematic review by da Silva et al. (2014) observed that:
    - ❑ researchers are unlikely to publish non-confirmatory results for their own work;
    - ❑ negative results are probably less likely to be accepted for publication;
    - ❑ negative results might be easier to publish when related to the work of others.
- This also confirms earlier observations that internal replications were more likely to be confirmatory than external ones [health warning: this could be a further reflection of publication bias]

3. Size of studies

A further challenge for human-based studies is for a primary study to have enough participants to achieve a reasonable level of statistical power (the ability of a statistical test to reveal a true pattern in the data).

For SE this is compounded by the need for participants to have appropriate levels of experience or expertise.

Distributed experiments

One way to involve more participants is to spread the study over different sites (in clinical studies, these are known as *multi-site trials*).

However for SE, this is again compounded by the skill issue.

We have conducted a trial of this idea, which used a fairly simple topic, and learned a lot about organising such studies. The idea does seem to have potential, but need to be developed further.

This also ties in with the question of having students as participants (yet another can of worms…).

A4. In Summary

Q4. What might we do to improve the quality of our knowledge?

A4. Many things! In particular, it is worth investigating ways to encourage:

- Better reporting practices
- Effective replications
- Larger experimental studies (distributed)
- Effective use of participant types (students vs practitioners)

So, what do we know (and…)?

- Empirical studies are not trivial to perform (especially when involving humans) but provide deeper insight and understanding than individual ‘expert opinion’.
- The evidence-based paradigm provides a means to synthesis empirical knowledge as well as to minimise the effects of local variation in studies.
- So far, our studies are mainly researcher-driven (unlike say, education, where policy-makers sponsor systematic reviews).
- HENCE, our knowledge is patchy, of uneven quality, and not always focused on the most ‘useful’ areas.
- BUT it is growing and we can expect it to improve and evolve.

Durham University

Acknowledgements

I'd like to acknowledge the contribution of the many researchers who have been working in the field of EBSE over the past ten years (and beyond), especially those quoted here.

In particular, I'd like to acknowledge the joint work with Barbara Kitchenham and Pearl Brereton on our forthcoming book, as well as the tertiary study on teaching material with Pearl, Nikki Williams and Sarah Drummond.

[I have learned a lot from all these people, but probably not as much as I should have done!]