The Science of Software Engineering

Marvin Zelkowitz
Department of Computer Science
University of Maryland
College Park, Maryland
marv@zelkowitz.com

ESEM - 2009
Organization of talk

- Some personal comments on how I arrived at the theme of this talk
- What are the issues in developing a science of software engineering?
- What's next?

So what have I been doing for the past 40 years?

- Most of my professional life has been at the University of Maryland, teaching and doing research in the general area of software engineering.
- But those who know me, know that I have three other areas of great interest.
One is attending science fiction conventions

A second is my interest in model railroading

Layout obviously unfinished.
A third is that I consider myself a professional skeptic.

I belong to an organization of skeptics.

Ha! Ha!
A third is that I consider myself a professional skeptic

That's not true!

No!

You can't do that!

It's all bogus!

What does this really mean?
And how does this relate to software engineering?
This is the general theme of the rest of this talk.
Common view of skeptics -
There's no way to help them

Are skeptics cynics?

- **Cynic** - One who shows a disposition to disbelieve in the sincerity or goodness of human motives and actions, and is wont to express this by sneers and sarcasms.
- **Skeptic** - one with doubt or incredulity as to the truth of some assertion or supposed fact.

Oxford English Dictionary

>A cynic disbelieves everything, a skeptic wants to be convinced.
Why is skepticism important?

Alternative medicine? UFOs? Dowsing? Spirits and ghosts?
Moon Landing was a hoax? Alien abduction? Natural remedies?
Therapeutic touch?

What do you believe? What should your opinion be about these?
NCAS’ mission: NCAS is an independent nonprofit educational and scientific organization that promotes critical thinking and scientific understanding, with a focus on paranormal and fringe-science claims. NCAS ... serves as an advocate for science and reason, actively promoting the scientific method, rational inquiry, and education.

Emphasis on proper use of science and the scientific method in everyday life

"Science is a way of thinking, much more than it is a body of facts." - Carl Sagan

E=MC²

People have 23 pairs of chromosomes

Water freezes at 32°F

Science is not a bag of facts

*Unless you are one of the 6 billion non-Americans, where water freezes at 0°C
Does science tell us reality?

- Science is not a bag of facts - it is a process to understand the world

Bad Science

- “Voodoo science” (Bob Park)
  - Bad science - Applying scientific method incorrectly
  - Mistakes - Performing an experiment incorrectly
  - Fraud - Intentional deception

- Pseudoscience - Violating accepted principles of the world (e.g., perpetual motion vs. thermodynamics)
### Methods of pseudoscience


<table>
<thead>
<tr>
<th>Ad Hominem</th>
<th>Argument By Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>Affirming The Consequent</td>
<td>Argument By Repetition (Ad Nauseam)</td>
</tr>
<tr>
<td>Amazing Familiarity</td>
<td>Argument by Rhetorical Question</td>
</tr>
<tr>
<td>Ambiguous Assertion</td>
<td>Argument by Scenario</td>
</tr>
<tr>
<td>Appeal To Anonymous Authority</td>
<td>Argument By Selective Observation</td>
</tr>
<tr>
<td>Appeal To Authority</td>
<td>Argument By Selective Reading</td>
</tr>
<tr>
<td>Appeal To Coincidence</td>
<td>Argument By Slogan</td>
</tr>
<tr>
<td>Appeal To Complexity</td>
<td>Argument From Adverse Consequences</td>
</tr>
<tr>
<td>Appeal To False Authority</td>
<td>Argument From Age</td>
</tr>
<tr>
<td>Appeal To Force</td>
<td>Argument From Authority</td>
</tr>
<tr>
<td>Appeal To Pity</td>
<td>Argument From False Authority</td>
</tr>
<tr>
<td>Appeal To Widespread Belief</td>
<td>Argument From Small Numbers</td>
</tr>
<tr>
<td>Argument By Emotive Language</td>
<td>Argument From Spurious Similarity</td>
</tr>
<tr>
<td>Argument By Fast Talking</td>
<td>Argument Of The Beard</td>
</tr>
<tr>
<td>Argument By Generalization</td>
<td>Argument To The Future</td>
</tr>
<tr>
<td>Argument By Gibberish</td>
<td>Bad Analogy</td>
</tr>
<tr>
<td>Argument By Half Truth</td>
<td>Begging The Question</td>
</tr>
<tr>
<td>Argument By Lack Of Evidence</td>
<td>Burden Of Proof</td>
</tr>
<tr>
<td>Argument By Personal Charm</td>
<td>Causal Reductionism</td>
</tr>
<tr>
<td>Argument By Pigheadedness</td>
<td>Changing The Subject</td>
</tr>
<tr>
<td>Argument By Poetic Language</td>
<td>Cliche Thinking</td>
</tr>
<tr>
<td>Argument By Prestigious Jargon</td>
<td>Common Sense</td>
</tr>
</tbody>
</table>

---

### Pseudoscience-2

<table>
<thead>
<tr>
<th>Complex Question (Tying)</th>
<th>Meaningless Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Confusing Correlation And Causation</td>
<td>Misunderstanding Statistics</td>
</tr>
<tr>
<td>Disproof By Fallacy</td>
<td>Moving The Goalposts</td>
</tr>
<tr>
<td>Euphemism</td>
<td>Needling</td>
</tr>
<tr>
<td>Error Of Fact</td>
<td>Non Sequitur</td>
</tr>
<tr>
<td>Exception That Proves The Rule</td>
<td>Not Invented Here</td>
</tr>
<tr>
<td>Excluded Middle</td>
<td>Outdated Information</td>
</tr>
<tr>
<td>Extended Analogy</td>
<td>Pious Fraud</td>
</tr>
<tr>
<td>Failure To State</td>
<td>Poisoning The Wells</td>
</tr>
<tr>
<td>Fallacy Of Composition</td>
<td>Psychogenetic Fallacy</td>
</tr>
<tr>
<td>Fallacy Of Division</td>
<td>Reductio Ad Absurdum</td>
</tr>
<tr>
<td>Fallacy Of The General Rule</td>
<td>Reductive Fallacy (Oversimplification)</td>
</tr>
<tr>
<td>Fallacy Of The Crucial Experiment</td>
<td>Reifying</td>
</tr>
<tr>
<td>False Cause</td>
<td>Short Term Versus Long Term</td>
</tr>
<tr>
<td>False Compromise</td>
<td>Slippery Slope Fallacy</td>
</tr>
<tr>
<td>Genetic Fallacy</td>
<td>Special Pleading (Stacking The Deck)</td>
</tr>
<tr>
<td>Having Your Cake</td>
<td>Statement Of Conversion</td>
</tr>
<tr>
<td>Hypothesis Contrary To Fact</td>
<td>Stolen Concept</td>
</tr>
<tr>
<td>Inconsistency</td>
<td>Straw Man</td>
</tr>
<tr>
<td>Inflation Of Conflict</td>
<td>Two Wrongs Make A Right</td>
</tr>
<tr>
<td>Internal Contradiction</td>
<td>Weasel Wording</td>
</tr>
<tr>
<td>Least Plausible Hypothesis</td>
<td>Lies</td>
</tr>
</tbody>
</table>
**Does science tell us reality?**

- A scientific theory is characterized by making *predictions* that can be disproved or falsified by observations; Nothing is ever said about "truth" – Truth and falsity are philosophical concepts.
- Example: We don’t know why or how gravity works
  - But we are quite sure if we step off the roof of a building, we will fall
  - And we are quite sure we know how long it will take and how fast we will hit the ground

Einstein's relativity revised Newton's theory of gravity with better predictions, but it still has to include explaining why stepping off a roof will make you go "splat" when you hit the ground.

---

**How does this relate to software engineering? Let's first discuss - Homeopathy**

- Conceived by Dr. Samuel Hahnemann in 1810
- Based on Law of similars – “Like cures like”
  - Any material that causes a reaction can be used, *if sufficiently dilute*, to eliminate that reaction
  - If pepper makes you sneeze, then a sufficiently dilute solution of pepper can cure the sneezing from allergies
By “dilute” we mean really really really dilute

- 1X dilution - 1 in 10, 2X dilution - 1 in 100 = 1:10^2 ...

  - 20X dilution - 1:10^{20}
  - But only 6.023\times10^{23} molecules per mole (e.g., for water it would be 18 grams)
  - So at dilutions of 30X, only 1 chance out of about 1,000,000 that even 1 molecule of substance is present in one cup of solution
  - Most homeopathic solutions are at least 1 C = 100X = 1:10^{100}

- Homeopathy is big business today. Billions of dollars annually in the USA; used worldwide

---

Identifying Pseudoscience -

<table>
<thead>
<tr>
<th>Use of vague, exaggerated or untestable claims</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over reliance on confirmation rather than refutation</td>
</tr>
<tr>
<td>Lack of openness to testing by other experts</td>
</tr>
<tr>
<td>Absence of progress</td>
</tr>
<tr>
<td>Personalization of issues; Proof by authority</td>
</tr>
<tr>
<td>Lack of scientific method</td>
</tr>
</tbody>
</table>
## Identifying Pseudoscience - Homeopathy

<table>
<thead>
<tr>
<th>Use of vague, exaggerated or untestable claims</th>
<th>Dilutions of 1C non-measurable: The higher the dilution, the better the effect; No rational underlying theory. Only proponents can “see” effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over reliance on confirmation rather than refutation</td>
<td>Testimonials on effectiveness. No blind studies of effects</td>
</tr>
<tr>
<td>Lack of openness to testing by other experts</td>
<td>Allowed to be sold in USA. FDA prevented by law from studying its effectiveness</td>
</tr>
<tr>
<td>Absence of progress</td>
<td>No change in “theory” since 1810</td>
</tr>
<tr>
<td>Personalization of issues; Proof by authority</td>
<td>Homeopathy = Hahnemann</td>
</tr>
<tr>
<td>Lack of scientific method</td>
<td>No multiple controlled studies ESEM – October 2009</td>
</tr>
</tbody>
</table>

I realized about 15 years ago, that my interest in skepticism and working on experimental software engineering were really the same thing.

How has software engineering changed over the past 20 years?

What needs to be done to further improve the field?
### Identifying Pseudoscience - Software Engineering?

<table>
<thead>
<tr>
<th>Use of vague, exaggerated or untestable claims</th>
<th>Dilutions of 1C non-measurable: The higher the dilution, the better the effect; No rational underlying theory; Only proponents can “see” effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Over reliance on confirmation rather than refutation</td>
<td>Testimonials on effectiveness; No blind studies of effects</td>
</tr>
<tr>
<td>Lack of openness to testing by other experts</td>
<td>Allowed to be sold in USA. FDA prevented by law from studying its effectiveness</td>
</tr>
<tr>
<td>Absence of progress</td>
<td>No change in “theory” since 1810</td>
</tr>
<tr>
<td>Personalization of issues; Proof by authority</td>
<td>Homeopathy = Hahnemann</td>
</tr>
<tr>
<td>Lack of scientific method</td>
<td>No multiple controlled studies</td>
</tr>
</tbody>
</table>

- How often is falsification of results attempted?

- “My technique makes programming easier” - How much easier? What does “easier” mean? How much would be “important”?

- “I tried it and it works” - (ESEM audience is rather good at multiple studies)

- “Paper is unpublishable since someone already ran that study”

- How often is a technique studied by a group other than the developer?

- Techniques often associated with developer. “I wrote it and I know what I’m doing.”

- How often is falsification of results attempted?
Falsifiability

- The scientific method works by hypothesis generation followed by experimentation.
- A major goal of experimentation is falsifiability.
  - Philosopher Karl Popper asserted that a hypothesis, proposition, or theory is scientific only if it is falsifiable.
  - A major goal of experimentation is to show that the theory is false (e.g., the negative is false).
  - Only when this occurs can you begin to assert that maybe theory is correct.
- Have you read “the results do not confirm this theory, so we will modify the approach ...”

So how has software engineering been doing in the science domain?

- Where were we 25 years ago in applying the scientific method?
- Where are we today?
- What still has to be done?
The language of science - Mathematics

"I think you should be more explicit here in step two."

Need for measurement

- A quote you see quite often in experimental software engineering venues:
  "I often say that when you can measure what you are speaking about, and express it in numbers, you can know something about it. But when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind."
  -- Lord Kelvin

- Corollary to above: We need relevant measurements

"The government is very keen on amassing statistics --- they collect them, add them, raise them to the nth power, take the cube root and prepare wonderful diagrams. But what you must never forget is that every one of those figures comes in the first instance from the village watchman, who just puts down what he damn pleases."
  -- British economist Josiah Stamp, 1929
Software engineering

- Let's start to get specific about SE.
- Does software engineering follow this model of science?
- In software engineering, the tools, methods, and techniques (e.g., the technologies) are our "theorems".
- Experimentation is how we partially validate (e.g., prove) whether these technologies are effective.
  - Note: As an experimental approach toward proof, we can only give approximations as to how well each technology works.
  - What can we say about experimentation in software engineering?

How do we measure the "science of software engineering"?

- Often there is a lack of validation before using a new technology.
  - Anecdotal evidence that we don't validate our claims.
  - Study by Tichy (1994) and Zelkowitz-Wallace (1998) confirm this.
  - Only 15% of papers in other scientific fields.
- Can we understand why this is so and how can we change this?
Role of experimentation studies

- Dolores Wallace and I reviewed over 600 published papers.
- Basic conclusion:
  - Approximately 50% of all software engineering papers had little validation of the claims in those papers.
  - Similar results were found by Walter Tichy in an earlier 1994 study.
- How does the software engineering community justify the many new technologies?
  - What methods are used to validate technologies?

Tichy study

- Data:
  - Reviewed 403 papers
  - Sources: ACM journals and conferences, IEEE TSE
- Classification of papers
  - Formal theory - proofs, ...
  - Design and modeling - designs which are not formal
  - Empirical study - evaluation of existing technology
  - Hypothesis testing - experiments to test a hypothesis
  - Other - anything else, e.g. surveys
- Conclusions:
  - 40% of computer science papers without validation
  - 50% of software engineering papers without validation
  - Comparable numbers are neuroscience (12%) and optical engineering (15%)
  - But only considered design and modeling papers. Perhaps too narrow a view
1998 Validation methods

- Experimental models often taken from domains like psychology and medicine:
  - View experimentation as the replication of a hypothesis under varying controlled conditions
  - Can we take a larger view of experimentation that applies in the software domain?
- This “classical method” of the controlled replicated experiment:
  - Not always feasible
  - Expensive (especially with large developments)
  - And there are other ways to evaluate technologies

Other experimental models

- Replicated experiments
  - Chemistry - Rows of test tubes
  - Psychology - Rows of freshmen students working on a task
- Observations of what happens
  - Medicine - Clinical trials, but “do no harm”
  - Astronomy - Observe events if and when they occur
- Data Mining of completed activities
  - Archaeology - Dig up the past
  - Forensic science - Recreate what happened
Developed a 15-step taxonomy of experimental methods

- **Classes of methods**
  - **Controlled method** - Multiple instances of an observation in order to provide for statistical validity of the results.
  - **Observational method** - Collect relevant data as it develops. In general, there is relatively little control over the development process.
  - **Historical method** - Collect data from completed projects.

Basic 12-step program

1. **Project monitoring.** Collect accounting data from a project and then study it.
2. **Case study.** Collect detailed project data.
3. **Field study.** Monitor several projects (e.g., survey).
4. **Literature search.** Evaluate previously published studies.
5. **Legacy data.** Evaluate data from a previously-completed project.
6. **Lessons learned.** Perform a qualitative analysis on a completed project.
7. **Static analysis.** Use a control flow analysis tool on the completed project.
8. **Replicated experiment.** Develop multiple instances of a project.
9. **Synthetic.** Replicate a simpler version of the technology in a laboratory.
10. **Dynamic analysis.** Execute a program using actual data.
11. **Simulation.** Generate data randomly according to a theoretical distribution.
12. **Theoretical.** Formal description of an abstract model.
But the list is incomplete

- What software engineers often do?
  - For a new technology validation often consists of: “I tried it, and I like it”
  - Validation often consists of a few trivial examples of using the technology to show that it works.
  - We added this validation as a weak form of case study as an assertion.

- **Assertion** - A simple form of case study that does not meet scientific standards for experimental validation. More like a feasibility study than a validation.

Evaluation of taxonomy

- Do the 13 methods described previously make any sense?
- Do research groups really use them?
- This led to the 1998 Zelkowitz-Wallace study and the 2006 update to it
  
  *(IEEE Computer, May 1998)*
1998 study

- National Institute of Standards and Technology
- Validate Tichy conclusions on a wider sample
  - Can we classify methods used to validate claims in a larger context than the Tichy survey?
  - Sources - All papers published in those years in:
    - IEEE Software
    - Transactions on Software Engineering
    - ICSE proceedings
  - 612 papers reviewed
- Every paper classified by 2 people
- But 13 categories were not enough

Non-validation methods

Validation methods
- 12 validation methods given previously

Non-validation methods

- **Assertion.** Informal feasibility demonstration.
- **Not applicable.** Paper was not appropriate for an experimental validation (e.g., tutorial, survey, news item)
- **No experimentation.** Default if none of the previous 14 methods applied. (e.g., paper should have had a validation, but didn’t)
Basic Data
(1998 study + 2006 update)

<table>
<thead>
<tr>
<th>Year</th>
<th>Project Monitoring</th>
<th>Case Study</th>
<th>Field Study</th>
<th>Literature Search</th>
<th>Legacy Lessons Learned</th>
<th>Static Analysis</th>
<th>Replicated Synthetic</th>
<th>Dynamic Analysis</th>
<th>Simulation</th>
<th>Assertion</th>
<th>Theoretical</th>
<th>No Experiment</th>
<th>Not Applicable</th>
<th>TTL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1985</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1990</td>
<td>0.04</td>
<td>0.10</td>
<td>0.03</td>
<td>0.01</td>
<td>0.03</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.03</td>
<td>0.01</td>
<td>0.03</td>
<td>0.04</td>
<td>0.01</td>
<td>0.25</td>
</tr>
<tr>
<td>1995</td>
<td>0.09</td>
<td>0.15</td>
<td>0.08</td>
<td>0.02</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
<td>0.02</td>
<td>0.01</td>
<td>0.02</td>
<td>0.01</td>
<td>0.09</td>
<td>0.07</td>
<td>0.54</td>
</tr>
<tr>
<td>2000</td>
<td>0.10</td>
<td>0.15</td>
<td>0.08</td>
<td>0.02</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
<td>0.02</td>
<td>0.01</td>
<td>0.02</td>
<td>0.01</td>
<td>0.10</td>
<td>0.07</td>
<td>0.58</td>
</tr>
<tr>
<td>2005</td>
<td>0.11</td>
<td>0.16</td>
<td>0.09</td>
<td>0.02</td>
<td>0.01</td>
<td>0</td>
<td>0</td>
<td>0.02</td>
<td>0.01</td>
<td>0.02</td>
<td>0.01</td>
<td>0.11</td>
<td>0.07</td>
<td>0.63</td>
</tr>
</tbody>
</table>

**1998 data**

- Method:
  - Project Monitoring: About 19%
  - Case Study: About 32%
Conclusions from 1998 study

- Most prevalent validation mechanisms were lessons learned and case studies, each about 10%.
- Simulation was used in about 5% of the papers, while the remaining techniques were each used in under 3% of the papers.
- BUT
  - Almost 20% of the papers had no experimental validation.
  - Assertions (a weak form of validation) were about one-third of the papers.
  - Resulting in over 50% of the papers having no real validation! (Different methodology, but same basic result as Tichy survey.)

Unexpected conclusions from 1998 study

- Every paper could fit into one of our categories, but:
  - Some papers can apply to 2 categories. We chose what we believed to be the major evaluation category.
  - We ignored what author said they were doing and tried to figure it out from context.
- Sometimes category extremely hard to uncover.
  - Authors often never stated why they were writing this paper.
  - e.g., “In this paper we describe our database Zork” never once saying why we need another database product or what it will do for us that existing database products don’t do.
  - Authors fail to state how they propose to validate their hypotheses.
  - Words like experiment, prototype, validate, pilot study, case study, ... used to mean many different things. We ignored those words and tried to objectively classify paper.
But there was one interesting result

- But percentages of “no experimentation” dropped from 27% in 1985 to 19% in 1990 to only 12% in 1995.
  - Perhaps indicative of a favorable trend

  - Perhaps a revised survey would show something interesting
  - So survey extended in 2006: 361 additional papers classified


---

1998 data
Anecdotal information

- "Case studies" steadily increasing
- "Assertions" dropping
- "No experimentation" dropping – generally, but not much
- More use of data repositories
  - Open source development histories (Mozilla, Apache) source of many papers
    - Increase in "Legacy" and "Dynamic analysis" methods
- Little change in controlled experiments (small increase to 7% of total)
Trends

![Trend Chart]

**Threats to validity**

- Consistency of taxonomy process
- Each data point very dependent on specific editors and program committees (e.g., 2000 ICSE)
- Change of scope in IEEE Software
- Quality of validation not indicated
  - Only tried to classify method used to validate paper, not whether the validation was correct
Study results

- We have proposed a 15-way approach toward developing a quantitative model of software experimentation.
- In general, the trend observed in 1998 that an increasing number of papers have an empirical validation component seems to be continuing through 2005.
- Informal feasibility studies (assertions) seem to be greatly declining
- Still need to look at the quality of those evaluations

Where do we go from here????

- Experimentation is only the beginning
- We need theories to explain software engineering phenomena
Aspects of a scientific theory

- **Parsimony**: Explains a variety of phenomena with a short, coherent explanation (Occam’s razor).
- **Generality**: Permits a reduction in the number of independent phenomena, with the single theory explaining a wide range of phenomena.
- **Prediction**: Anticipates future events, often to a high degree of accuracy.
- **Explanatory**: Provides a compelling underlying mechanism.

Development of theories

- **General**
- **System-specific**
  - Descriptive
  - Predictive
  - Explanatory

Development of theories


ESEM – October 2009

There are activities looking at the problem

- R. Snodgrass - U. of Arizona - Ergalics
- Discussion of role of conferences and journals in Comm. of the ACM (May 2009)
- Data sharing, ISERN meeting, 2005
- Empirical Software Engineering Journal
- But few community-wide discussion of the problems
Summary

- We need to become better skeptics, both outside and within the software engineering community.
- We need to be more aware of when we deviate into voodoo science and must avoid the pitfalls of pseudoscience.
- We have few predictive models and almost no explanatory models of software processes - We cannot tell researchers “invent new better theories”, but we can be more aware of that goal.
- Although computer science has vastly improved its adherence to the scientific method over the past 25 years, we still can do better.
- Take-away message: Until we overcome those limitations, our impact on the general scientific establishment will be limited.

That’s all folks!
The Science of Software Engineering

Marvin Zelkowitz
Department of Computer Science
University of Maryland
College Park, Maryland
marv@zelkowitz.com

ESEM - 2009