CEN 5016: Software Engineering

Spring 2024

University of Central Florida

Dr. Kevin Moran

Week 2 - Class 1: Measurement & Metrics

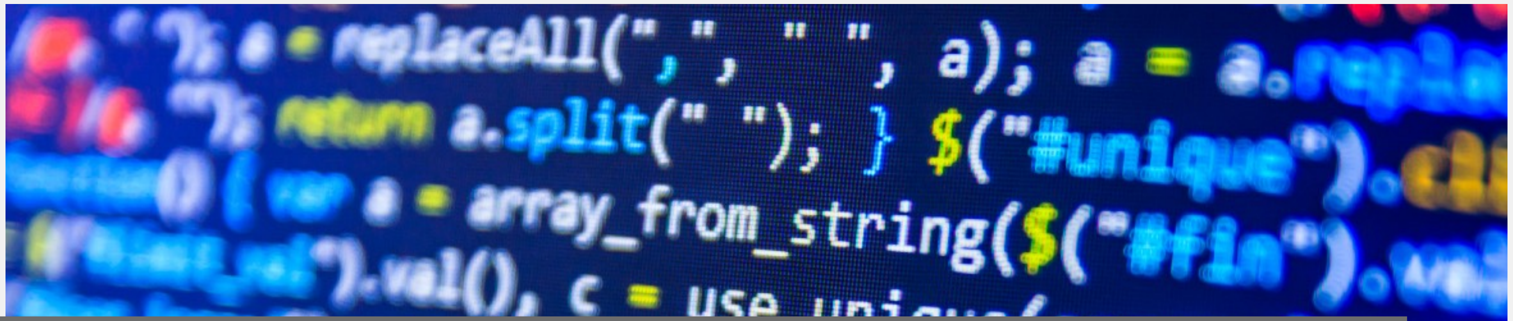
Administrivia

- Course Schedule Posted
- Office Hours Decided
  - Tuesday/Thursday 12:00pm-1:00pm (before class) Hybrid
  - Or by appointment
- Let me know if you are not on Ed Discussions
- Assignment 1, Getting started with Git, GitHub, and Typescript is due tonight at 11:59 pm
  - Use Megathread on Ed Discussions to ask questions
- Team-forming this week
  - Teams of 3 students
  - Look out for a post on Ed Discussions
- Assignment 2 out tomorrow

Software Archeology & Anthropology

Observation: Software is Full of Patterns

- File structure
- System architecture
- Code structure
- Names
- …



```
a = replaceAll(",", " ", a); a = a.repl
return a.split(" "); } $("#unique").cl
var a = array_from_string($("#fin").val(), c = use_unique(
```

On the Naturalness of Software

Abram Hindle, Earl T. Barr, Zhendong Su
Dept. of Computer Science
University of California at Davis
Davis, CA 95616 USA
{ajhindle,barr,su}@cs.ucdavis.edu

Mark Gabel
Dept. of Computer Science
The University of Texas at Dallas
Richardson, TX 75080 USA
mark.gabel@utdallas.edu

Premkumar Devanbu
Dept. of Computer Science
University of California at Davis
Davis, CA 95616 USA
devanbu@cs.ucdavis.edu

***Abstract*—Natural languages like English are rich, complex, and powerful. The highly creative and graceful use of languages like English and Tamil, by masters like Shakespeare and Avvaiyar, can certainly delight and inspire. But in practice, given cognitive constraints and the exigencies of daily life, most human utterances are far simpler and much more repetitive and predictable. In fact, these utterances can be very usefully modeled using modern statistical methods. This fact has led to the phenomenal success of statistical approaches to speech recognition, natural language translation, question-answering, and text mining and comprehension.**

We begin with the conjecture that most software is also natural, in the sense that it is created by humans at work, with all the attendant constraints and limitations—and thus, like natural language, it is also likely to be repetitive and predictable. We then proceed to ask whether a) code can be usefully modeled by statistical language models and b) such models can be leveraged to support software engineers. Using the widely adopted n-gram model, we provide empirical evidence supportive of a positive answer to both these questions. We show that code is also very repetitive, and in fact even more so than natural languages. As an example use of the model, we have developed a simple code completion engine for Java that, despite its simplicity, already improves Eclipse's built-in completion capability. We conclude the paper by laying out a vision for future research in this area.

***Keywords*-language models; n-gram; natural language processing; code completion; and code suggestion**

efforts in the 1960s. In the '70s and '80s, the field was reanimated with ideas from logic and formal semantics, which still proved too cumbersome to perform practical tasks at scale. Both these approaches essentially dealt with NLP from first principles—addressing *language*, in all its rich theoretical glory, rather than examining corpora of actual *utterances*, *i.e.*, what people actually write or say. In the 1980s, a fundamental shift to *corpus-based, statistically rigorous* methods occurred. The availability of large, on-line corpora of natural language text, including "aligned" text with translations in multiple languages,[1] along with the computational muscle (CPU speed, primary and secondary storage) to estimate robust statistical models over very large data sets has led to stunning progress and widely-available practical applications, such as statistical translation used by `translate.google.com`.[2] We argue that an essential fact underlying this modern, exciting phase of NLP is *natural language may be complex and admit a great wealth of expression, but what people write and say is largely regular and predictable.*

Our *central hypothesis* is that the same argument applies to software:

> *Programming languages, in theory, are complex,*

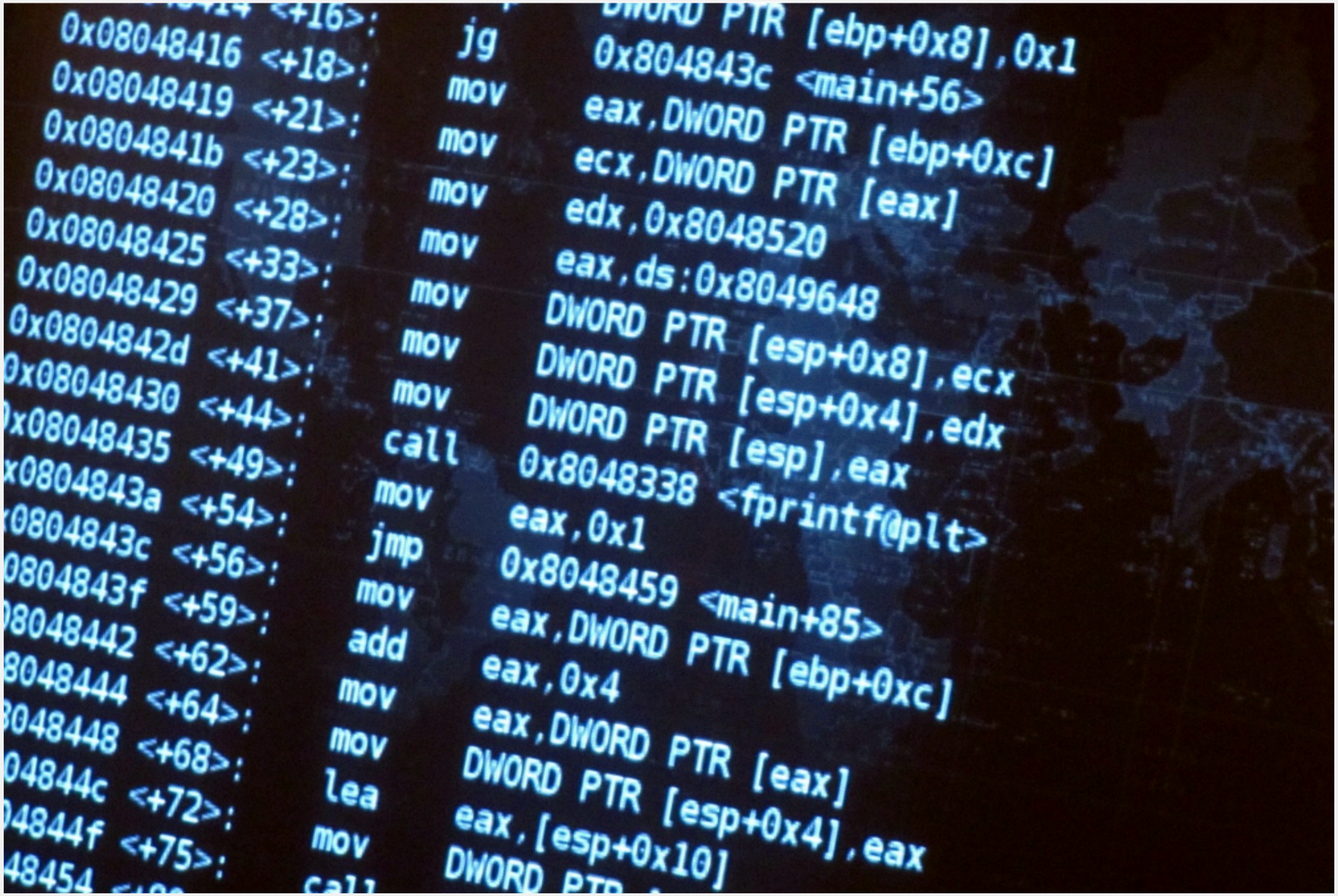
Observation: Software is Massively Redundant

- There is always something to copy/use as a starting point!

Observation: Code Must Run to Do Stuff!

Observation: If Code Runs, it Must have a Beginning…

Observation: If Code Runs, it Must Exist



```
 ...414 <+16>:          DWORD PTR [ebp+0x8],0x1
0x08048416 <+18>:  jg    0x804843c <main+56>
0x08048419 <+21>:  mov   eax,DWORD PTR [ebp+0xc]
0x0804841b <+23>:  mov   ecx,DWORD PTR [eax]
0x08048420 <+28>:  mov   edx,0x8048520
0x08048425 <+33>:  mov   eax,ds:0x8049648
0x08048429 <+37>:  mov   DWORD PTR [esp+0x8],ecx
0x0804842d <+41>:  mov   DWORD PTR [esp+0x4],edx
0x08048430 <+44>:  mov   DWORD PTR [esp],eax
x08048435 <+49>:  call  0x8048338 <fprintf@plt>
x0804843a <+54>:  mov   eax,0x1
0804843c <+56>:  jmp   0x8048459 <main+85>
0804843f <+59>:  mov   eax,DWORD PTR [ebp+0xc]
8048442 <+62>:  add   eax,0x4
8048444 <+64>:  mov   eax,DWORD PTR [eax]
048448 <+68>:  mov   DWORD PTR [esp+0x4],eax
04844c <+72>:  lea   eax,[esp+0x10]
4844f <+75>:  mov   DWORD PTR
48454 <+8     call
```

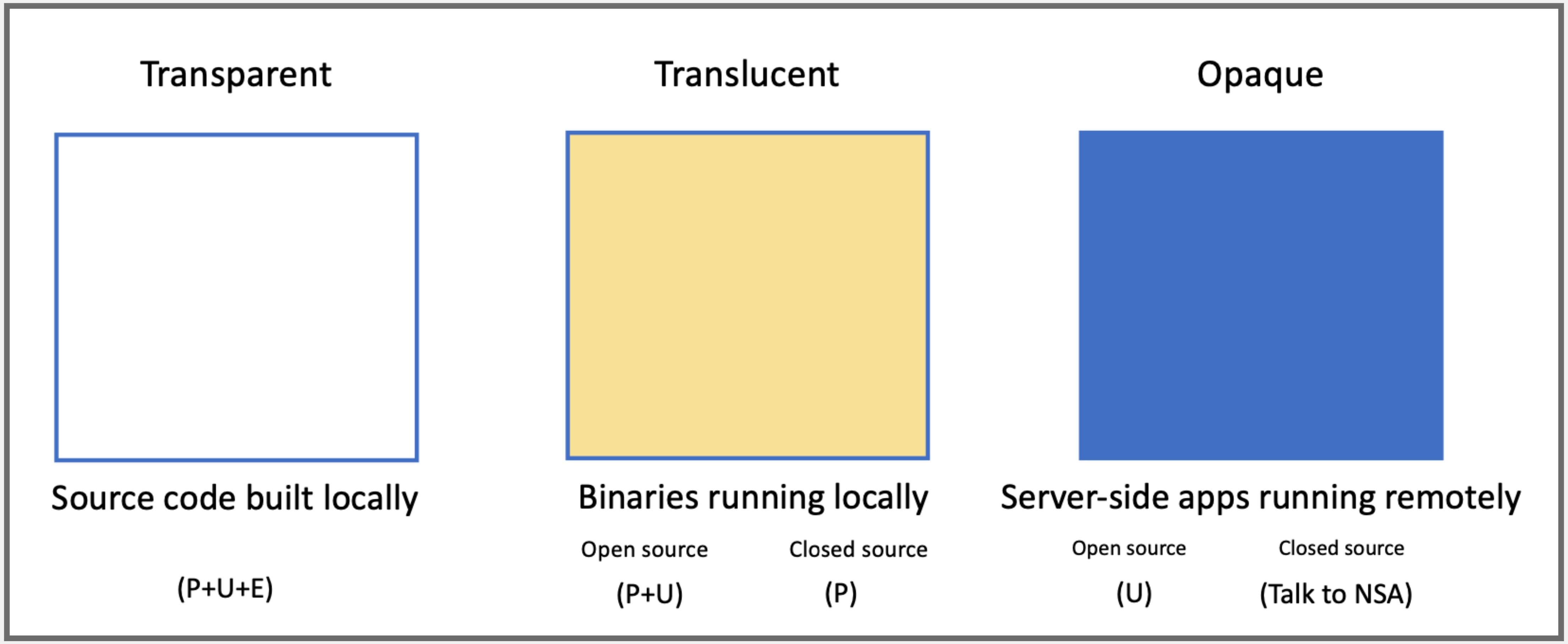
The Beginning: Entry Points

- Locally installed programs: run cmd, OS launch, I/O events, etc.
- Local applications in dev: build + run, test, deploy (e.g., docker)
- Web apps server-side: Browser sends HTTP request (GET/POST)
- Web apps client-side: Browser runs JavaScript, event handlers

Code Must Exist: But Where?

- Locally installed programs: run cmd, OS launch, I/O events, etc.
  - Binaries (machine code) on your computer
- Local applications in dev: build + run, test, deploy (e.g., docker)
  - Source code in repository (+ dependencies)
- Web apps server-side: Browser sends HTTP request (e.g., GET, POST)
  - Code runs remotely (you can only observe outputs)
- Web apps client-side: Browser runs JavaScript, event handlers
  - Source code is downloaded and run locally (see: browser dev tools!)

Can Running Code be Probed/Understood/Edited?



Transparent

Source code built locally

(P+U+E)

Translucent

Binaries running locally

Open source

(P+U)

Closed source

(P)

Opaque

Server-side apps running remotely

Open source

(U)

Closed source

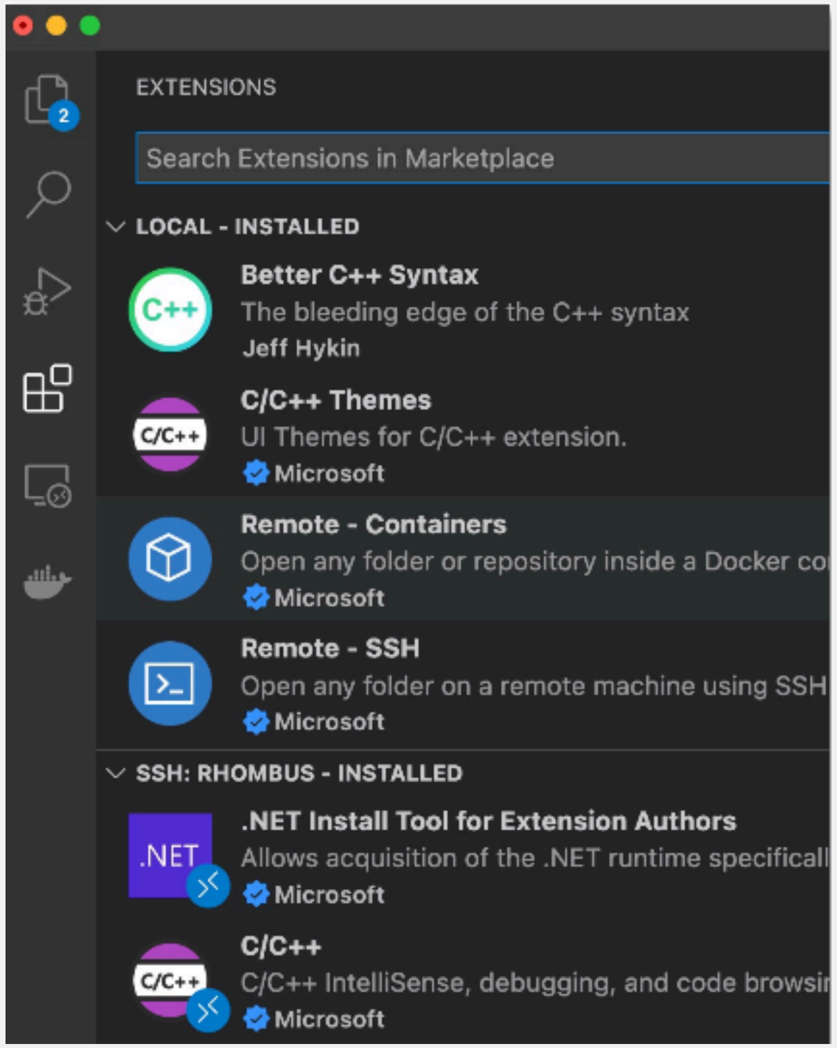
(Talk to NSA)

Creating a Model of Unfamiliar Code

Information Gathering

- Basic needs:
  - Code/file search and navigation
  - Code editing (probes)
  - Execution of code, tests
  - Observation of output (observation)
- At the command line: grep and find! (Google for tutorials)
- Many choices here on tools! Depends on circumstance.
  - grep/find/etc.
  - Knowing Unix tools is invaluable
  - A decent IDE
  - Debugger
  - Test frameworks + coverage reports
  - Google (or your favorite web search engine)
  - ChatGPT or LaMA

Static Information Gathering: Use an IDE!



Better C++ Syntax
The bleeding edge of the C++ syntax
Jeff Hykin

C/C++ Themes
UI Themes for C/C++ extension.
Microsoft

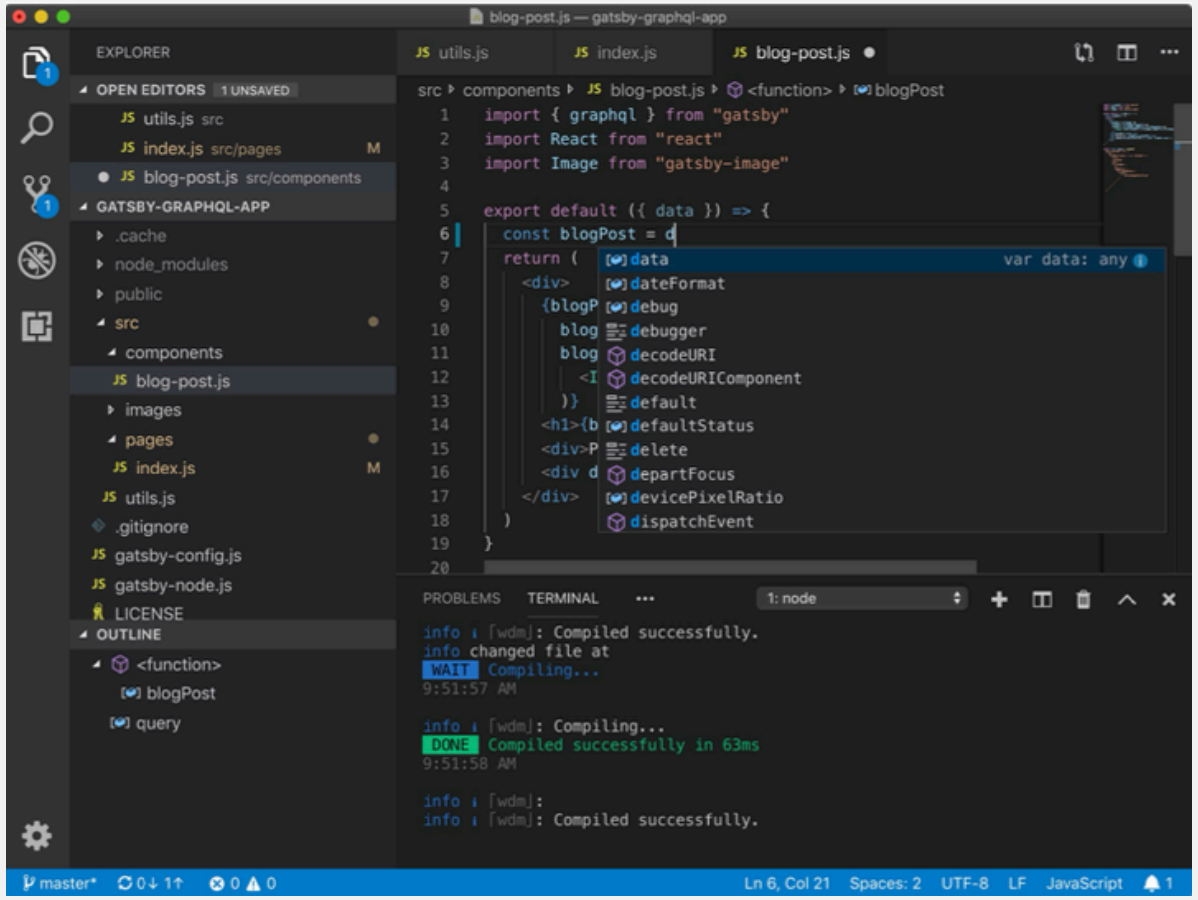
Remote - Containers
Open any folder or repository inside a Docker co
Microsoft

Remote - SSH
Open any folder on a remote machine using SSH
Microsoft

SSH: RHOMBUS - INSTALLED

.NET Install Tool for Extension Authors
Allows acquisition of the .NET runtime specificall
Microsoft

C/C++
C/C++ IntelliSense, debugging, and code browsi
Microsoft



```
import { graphql } from "gatsby"
import React from "react"
import Image from "gatsby-image"

export default ({ data }) => {
  const blogPost = d
  return (
    <div>
      {blogP
        blog
        blog
          <I
      )}
      <h1>{b
      <div>P
      <div d
    </div>
  )
}
```

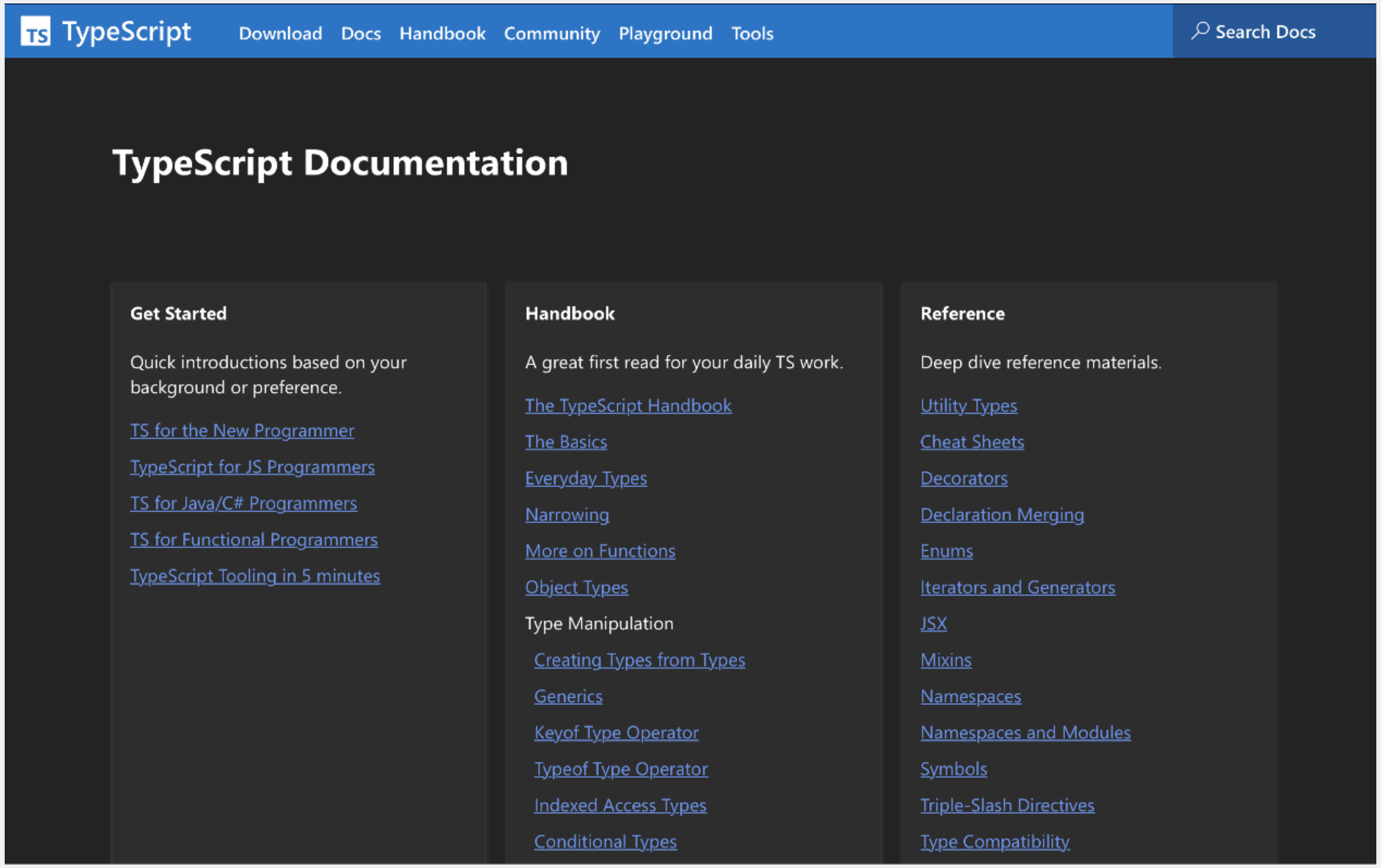
```
info ｉ ｢wdm｣: Compiled successfully.
info changed file at
 WAIT  Compiling...
9:51:57 AM

info ｉ ｢wdm｣: Compiling...
 DONE  Compiled successfully in 63ms
9:51:58 AM

info ｉ ｢wdm｣:
info ｉ ｢wdm｣: Compiled successfully.
```

Consider Documentation and Tutorials Judiciously

- Great for discovering entry points!
- Can teach you about general structure, architecture (more on this later in the semester)
- Often out of date.
- As you gain experience, you will recognize more of these, and you will immediately know something about how the program works
- Also: discussion boards; issue trackers



TypeScript Documentation

Get Started

Quick introductions based on your background or preference.

- TS for the New Programmer
- TypeScript for JS Programmers
- TS for Java/C# Programmers
- TS for Functional Programmers
- TypeScript Tooling in 5 minutes

Handbook

A great first read for your daily TS work.

- The TypeScript Handbook
- The Basics
- Everyday Types
- Narrowing
- More on Functions
- Object Types
- Type Manipulation
  - Creating Types from Types
  - Generics
  - Keyof Type Operator
  - Typeof Type Operator
  - Indexed Access Types
  - Conditional Types

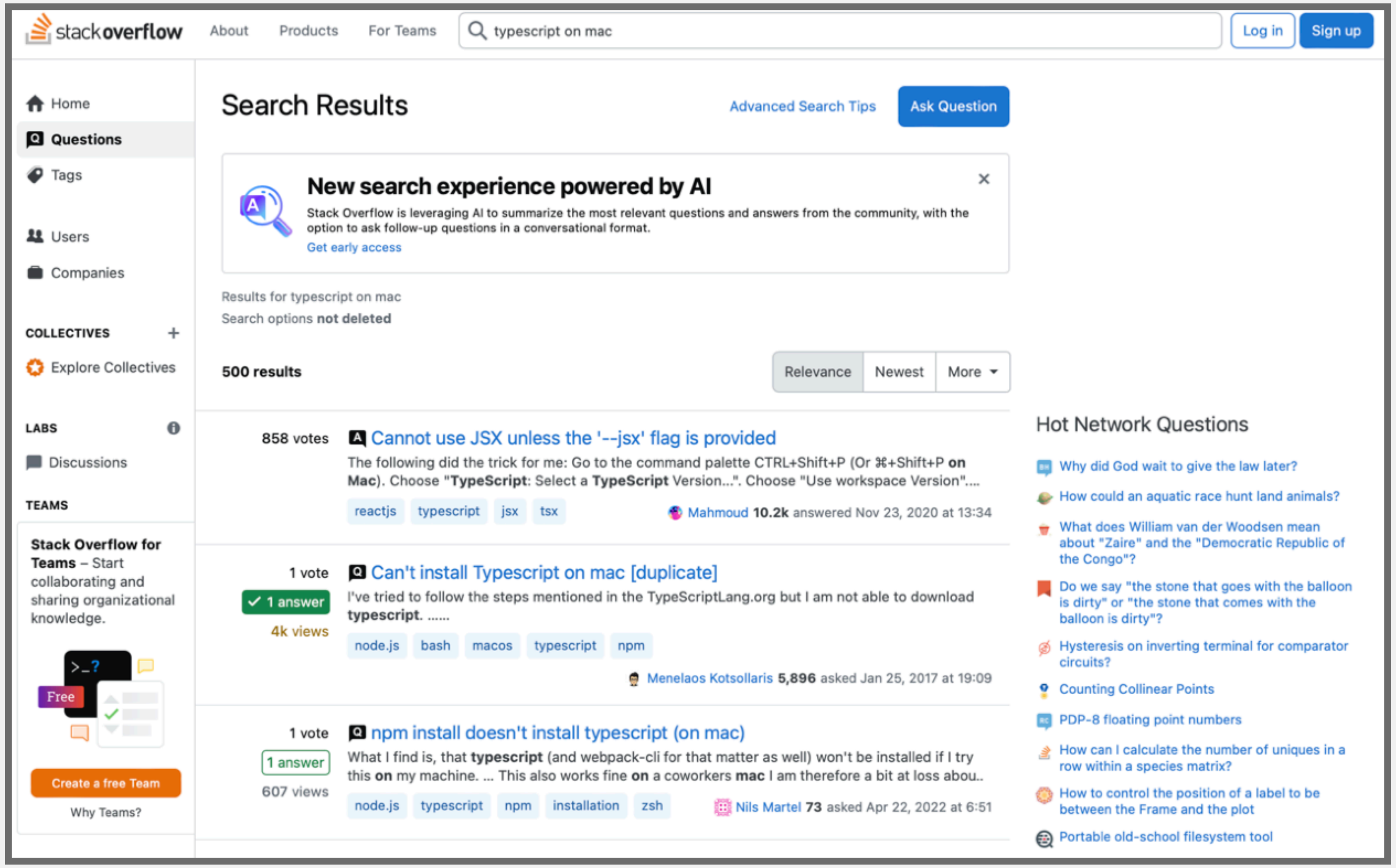
Reference

Deep dive reference materials.

- Utility Types
- Cheat Sheets
- Decorators
- Declaration Merging
- Enums
- Iterators and Generators
- JSX
- Mixins
- Namespaces
- Namespaces and Modules
- Symbols
- Triple-Slash Directives
- Type Compatibility

Discussion Boards and Issue Trackers

- Software is written by people.
- How can we talk to them?
- Fortunately, they probably aren’t dead.
- So, you can report problems on GitHub.
- Or, ask them questions on StackOverflow.



Search Results

New search experience powered by AI

Stack Overflow is leveraging AI to summarize the most relevant questions and answers from the community, with the option to ask follow-up questions in a conversational format.

Results for typescript on mac

Search options **not deleted**

500 results

858 votes — Cannot use JSX unless the '--jsx' flag is provided

The following did the trick for me: Go to the command palette CTRL+Shift+P (Or ⌘+Shift+P **on** **Mac**). Choose "**TypeScript**: Select a **TypeScript** Version...". Choose "Use workspace Version"....

reactjs typescript jsx tsx — Mahmoud **10.2k** answered Nov 23, 2020 at 13:34

1 vote, ✓ 1 answer, 4k views — Can't install Typescript on mac [duplicate]

I've tried to follow the steps mentioned in the TypeScriptLang.org but I am not able to download **typescript**. ......

node.js bash macos typescript npm — Menelaos Kotsollaris **5,896** asked Jan 25, 2017 at 19:09

1 vote, 1 answer, 607 views — npm install doesn't install typescript (on mac)

What I find is, that **typescript** (and webpack-cli for that matter as well) won't be installed if I try this **on** my machine. ... This also works fine **on** a coworkers **mac** I am therefore a bit at loss abou..

node.js typescript npm installation zsh — Nils Martel **73** asked Apr 22, 2022 at 6:51

Stack Overflow for Teams – Start collaborating and sharing organizational knowledge.

Hot Network Questions

- Why did God wait to give the law later?
- How could an aquatic race hunt land animals?
- What does William van der Woodsen mean about "Zaire" and the "Democratic Republic of the Congo"?
- Do we say "the stone that goes with the balloon is dirty" or "the stone that comes with the balloon is dirty"?
- Hysteresis on inverting terminal for comparator circuits?
- Counting Collinear Points
- PDP-8 floating point numbers
- How can I calculate the number of uniques in a row within a species matrix?
- How to control the position of a label to be between the Frame and the plot
- Portable old-school filesystem tool

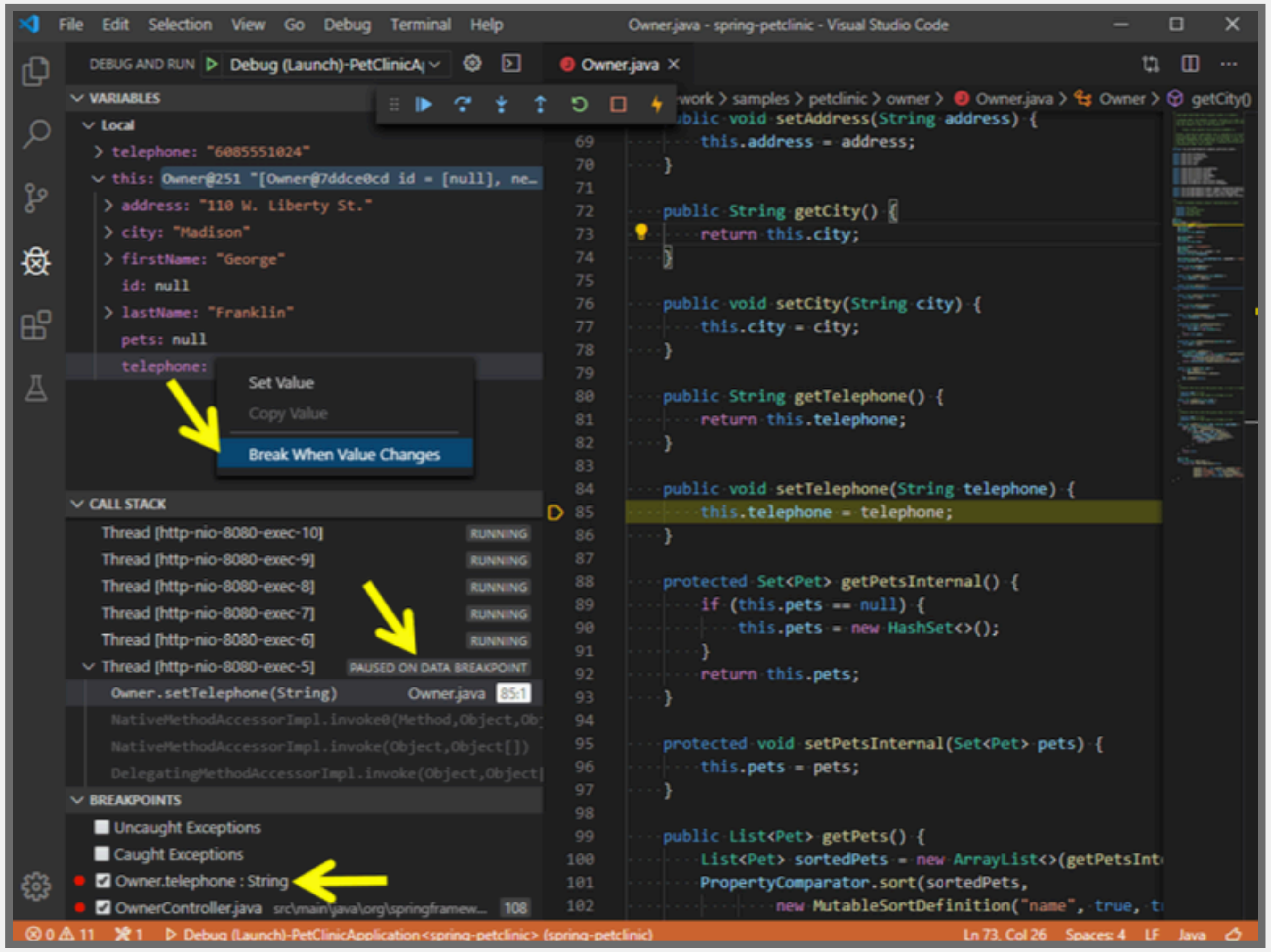
Dynamic Information Gathering

- Build it.
- Run it.
- Change it.
- Run it again.

- How did the behavior change?

Probes: Observe, Control, or "Lightly" Manipulate Execution

- print("this code is running!")
- Structured logging
- Debuggers
    - Breakpoint, eval, step through / step over
    - (Some tools even support remote debugging)
- Delete debugging
- Chrome Developer Tools


File
Edit
Selection
View
Go
Debug
Terminal
Help
Owner.java - spring-petclinic - Visual Studio Code
DEBUG AND RUN
Debug (Launch)-PetClinicA
Owner.java
work > samples > petclinic > owner > Owner.java > Owner > getCity()
Set Value
Copy Value
Break When Value Changes

VARIABLES

Local

```
> telephone: "6085551024"
v this: Owner@251 "[Owner@7ddce0cd id = [null], ne…
  > address: "110 W. Liberty St."
  > city: "Madison"
  > firstName: "George"
    id: null
  > lastName: "Franklin"
    pets: null
    telephone:
```

CALL STACK

```
  Thread [http-nio-8080-exec-10]        RUNNING
  Thread [http-nio-8080-exec-9]         RUNNING
  Thread [http-nio-8080-exec-8]         RUNNING
  Thread [http-nio-8080-exec-7]         RUNNING
  Thread [http-nio-8080-exec-6]         RUNNING
v Thread [http-nio-8080-exec-5]         PAUSED ON DATA BREAKPOINT
    Owner.setTelephone(String)          Owner.java  85:1
    NativeMethodAccessorImpl.invoke0(Method,Object,Ob
    NativeMethodAccessorImpl.invoke(Object,Object[])
    DelegatingMethodAccessorImpl.invoke(Object,Object
```

BREAKPOINTS

- □ Uncaught Exceptions
- □ Caught Exceptions
- ☑ Owner.telephone : String
- ☑ OwnerController.java src\main\java\org\springframew... 108

```java
    public void setAddress(String address) {
        this.address = address;
    }

    public String getCity() {
        return this.city;
    }

    public void setCity(String city) {
        this.city = city;
    }

    public String getTelephone() {
        return this.telephone;
    }

    public void setTelephone(String telephone) {
        this.telephone = telephone;
    }

    protected Set<Pet> getPetsInternal() {
        if (this.pets == null) {
            this.pets = new HashSet<>();
        }
        return this.pets;
    }

    protected void setPetsInternal(Set<Pet> pets) {
        this.pets = pets;
    }

    public List<Pet> getPets() {
        List<Pet> sortedPets = new ArrayList<>(getPetsInt
        PropertyComparator.sort(sortedPets,
                new MutableSortDefinition("name", true, t
```

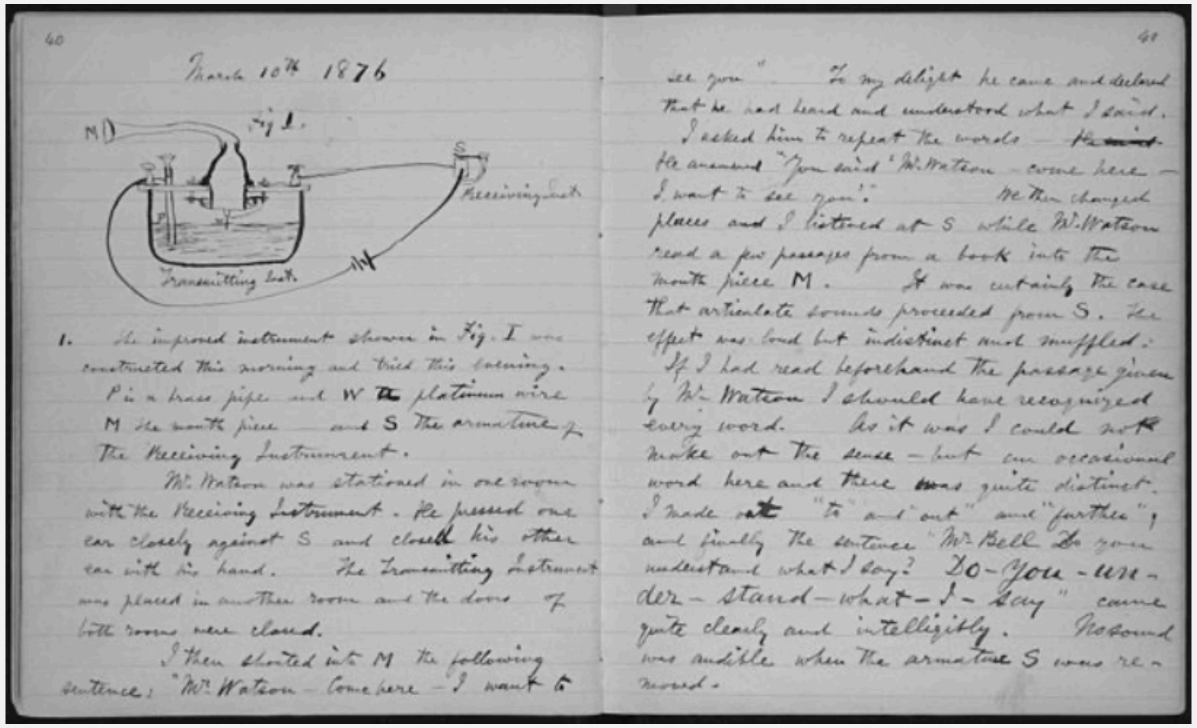
⊗ 0 ⚠ 11 1 ▷ Debug (Launch)-PetClinicApplication<spring-petclinic> (spring-petclinic) Ln 73, Col 26 Spaces: 4 LF Java

Step 0: Sanity Check Basic Model + Hypotheses

- *Confirm that you can build and run the code.*
  - Ideally both using the tests provided, and by hand.
- *Confirm that the code you are running is the code you built*
- *Confirm that you can make an externally visible change*
- *How? Where? Starting points:*
  - Run an existing test, change it
  - Write a new test
  - Change the code, write or rerun a test that should notice the change
- *Ask someone for help*

Document and Share Your Findings!

- Update README and docs
  - Or better: use a Developer Wiki
  - Use Mermaid for diagrams
- Screencast on Twitch
- Collaborate with others
- Include negative results, too!



40

March 10th 1876

Fig 1.

M
S
Receiving Inst.
P
W
Transmitting Inst.

1. The improved instrument shown in Fig. I was constructed this morning and tried this evening. P is a brass pipe and W ~~the~~ platinum wire M the mouth piece and S the armature of the Receiving Instrument.

Mr. Watson was stationed in one room with the Receiving Instrument. He pressed one ear closely against S and closed his other ear with his hand. The Transmitting Instrument was placed in another room and the doors of both rooms were closed.

I then shouted into M the following sentence: "Mr. Watson – Come here – I want to

41

see you." To my delight he came and declared that he had heard and understood what I said. I asked him to repeat the words – ~~He said~~ He answered "You said 'Mr. Watson – come here – I want to see you.'" We then changed places and I listened at S while Mr. Watson read a few passages from a book into the mouth piece M. It was certainly the case that articulate sounds proceeded from S. The effect was loud but indistinct and muffled. If I had read beforehand the passage given by Mr. Watson I should have recognized every word. As it was I could not make out the sense – but an occasional word here and there ~~was~~ was quite distinct. I made ~~out~~ "to" and "out" and "further"; and finally the sentence "Mr. Bell Do you understand what I say? Do-you-un-der-stand-what-I-say" came quite clearly and intelligibly. No sound was audible when the armature S was removed.

Metrics & Measurement

Goals for Today

- Use measurements as a decision tool to reduce uncertainty
- Understand difficulty of measurement; discuss validity of measurements
- Provide examples of metrics for software qualities and process
- Understand limitations and dangers of decisions and incentives based on measurements

Software Engineering

- Software Engineering: Principles, practices (technical and non-technical) for confidently building high-quality software.

What does this mean?
How do we know?
-> Measurement & Metrics
are key concerns

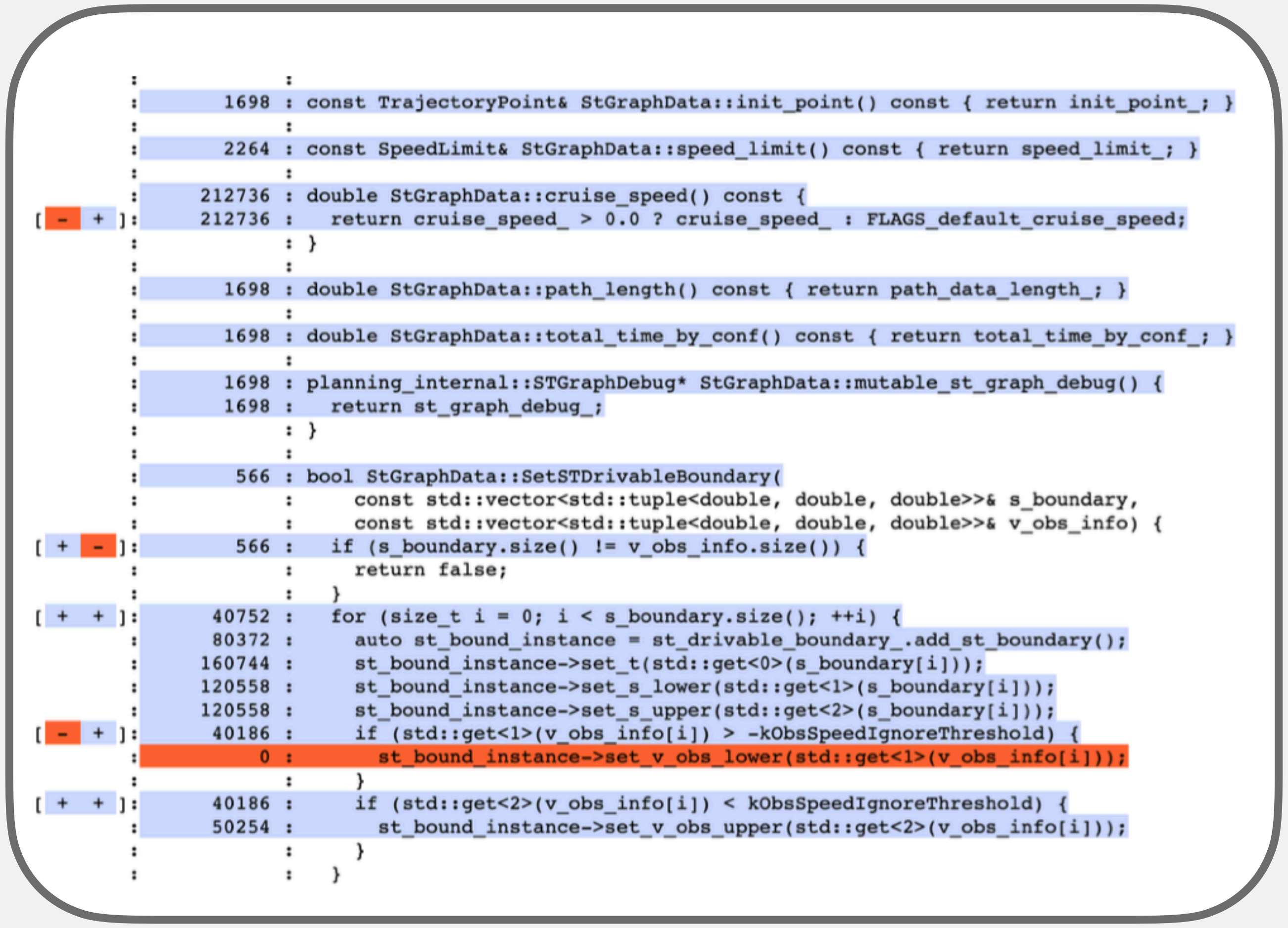
Case Study: Autonomous Vehicles

Case Study: Autonomous Vehicles

- By what methods can we judge AV software quality (e.g., safety)?

Test Coverage

- Amount of code executed during testing.
- Statement coverage, line coverage, branch coverage, etc.
- E.g., 75% branch coverage -> 3/4 if-else outcomes have been executed



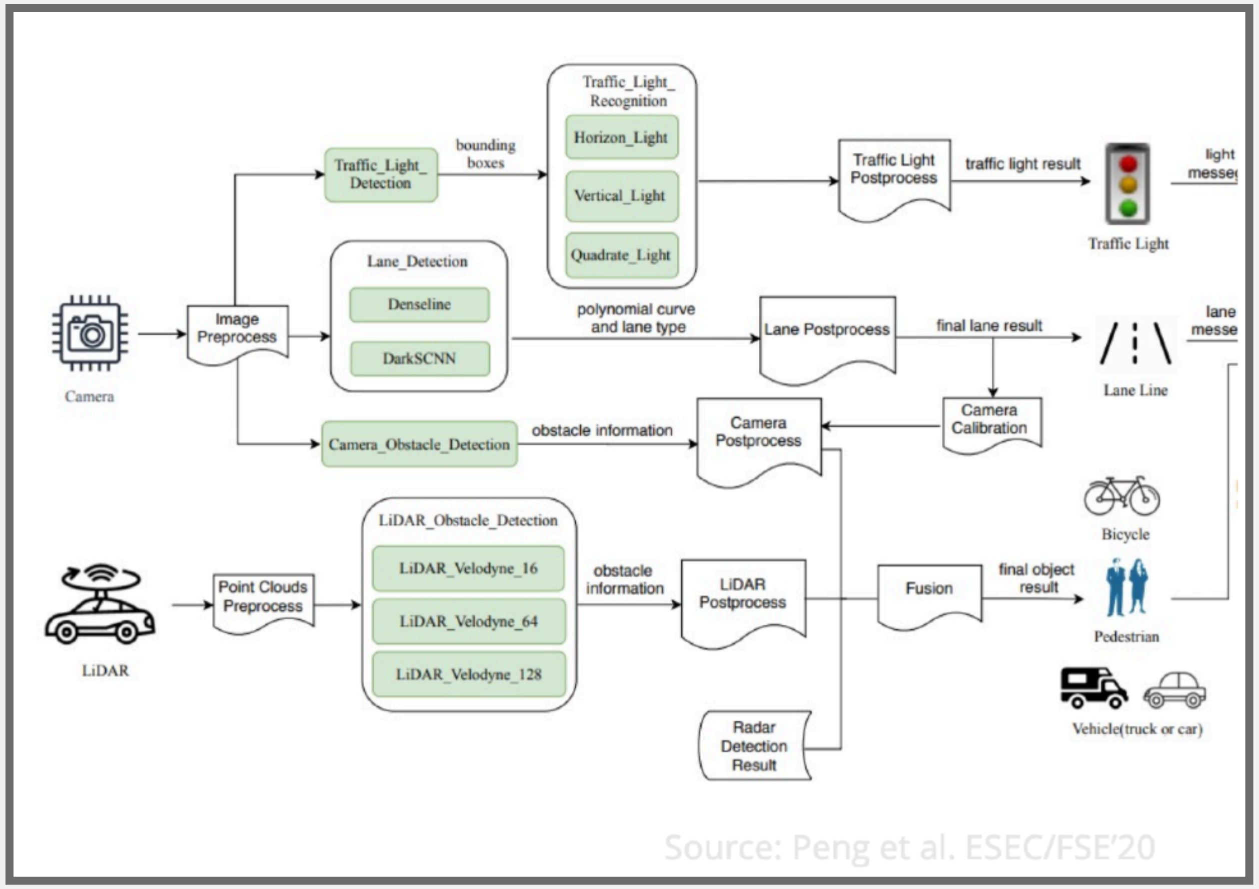
```
          :        :
          :   1698 : const TrajectoryPoint& StGraphData::init_point() const { return init_point_; }
          :        :
          :   2264 : const SpeedLimit& StGraphData::speed_limit() const { return speed_limit_; }
          :        :
          : 212736 : double StGraphData::cruise_speed() const {
[ -  + ]: 212736 :   return cruise_speed_ > 0.0 ? cruise_speed_ : FLAGS_default_cruise_speed;
          :        : }
          :        :
          :   1698 : double StGraphData::path_length() const { return path_data_length_; }
          :        :
          :   1698 : double StGraphData::total_time_by_conf() const { return total_time_by_conf_; }
          :        :
          :   1698 : planning_internal::STGraphDebug* StGraphData::mutable_st_graph_debug() {
          :   1698 :   return st_graph_debug_;
          :        : }
          :        :
          :    566 : bool StGraphData::SetSTDrivableBoundary(
          :        :     const std::vector<std::tuple<double, double, double>>& s_boundary,
          :        :     const std::vector<std::tuple<double, double, double>>& v_obs_info) {
[ +  - ]:    566 :   if (s_boundary.size() != v_obs_info.size()) {
          :        :     return false;
          :        :   }
[ +  + ]:  40752 :   for (size_t i = 0; i < s_boundary.size(); ++i) {
          :  80372 :     auto st_bound_instance = st_drivable_boundary_.add_st_boundary();
          : 160744 :     st_bound_instance->set_t(std::get<0>(s_boundary[i]));
          : 120558 :     st_bound_instance->set_s_lower(std::get<1>(s_boundary[i]));
          : 120558 :     st_bound_instance->set_s_upper(std::get<2>(s_boundary[i]));
[ -  + ]:  40186 :     if (std::get<1>(v_obs_info[i]) > -kObsSpeedIgnoreThreshold) {
          :      0 :       st_bound_instance->set_v_obs_lower(std::get<1>(v_obs_info[i]));
          :        :     }
[ +  + ]:  40186 :     if (std::get<2>(v_obs_info[i]) < kObsSpeedIgnoreThreshold) {
          :  50254 :       st_bound_instance->set_v_obs_upper(std::get<2>(v_obs_info[i]));
          :        :     }
          :        :   }
```

Model Accuracy

- Train machine-learning models on labelled data (sensor data + ground truth).

- Compute accuracy on a separate labelled test set.

- E.g., 90% accuracy implies that object recognition is right for 90% of the test inputs.



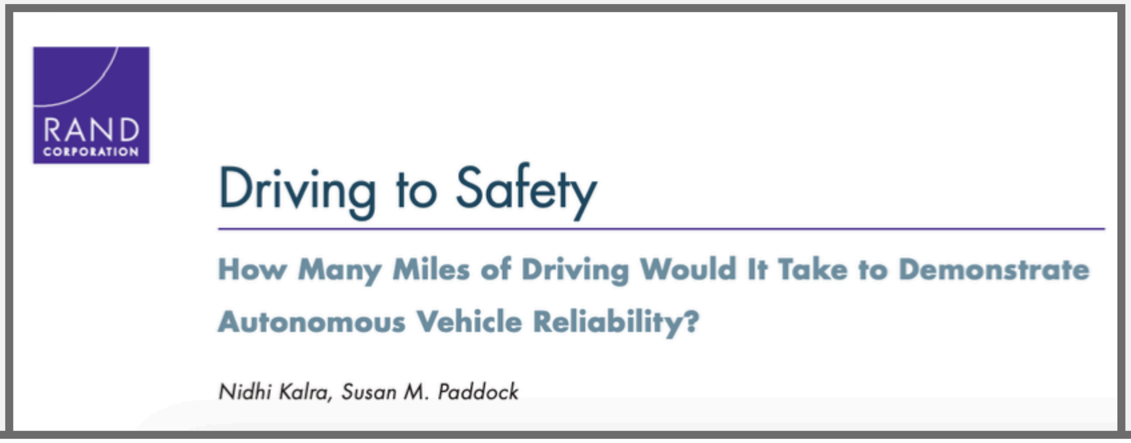
Source: Peng et al. ESEC/FSE'20

Failure Rate

- Frequency of crashes / fatalities

- Per 1,000 rides, per million miles, per month (in the news)

Mileage



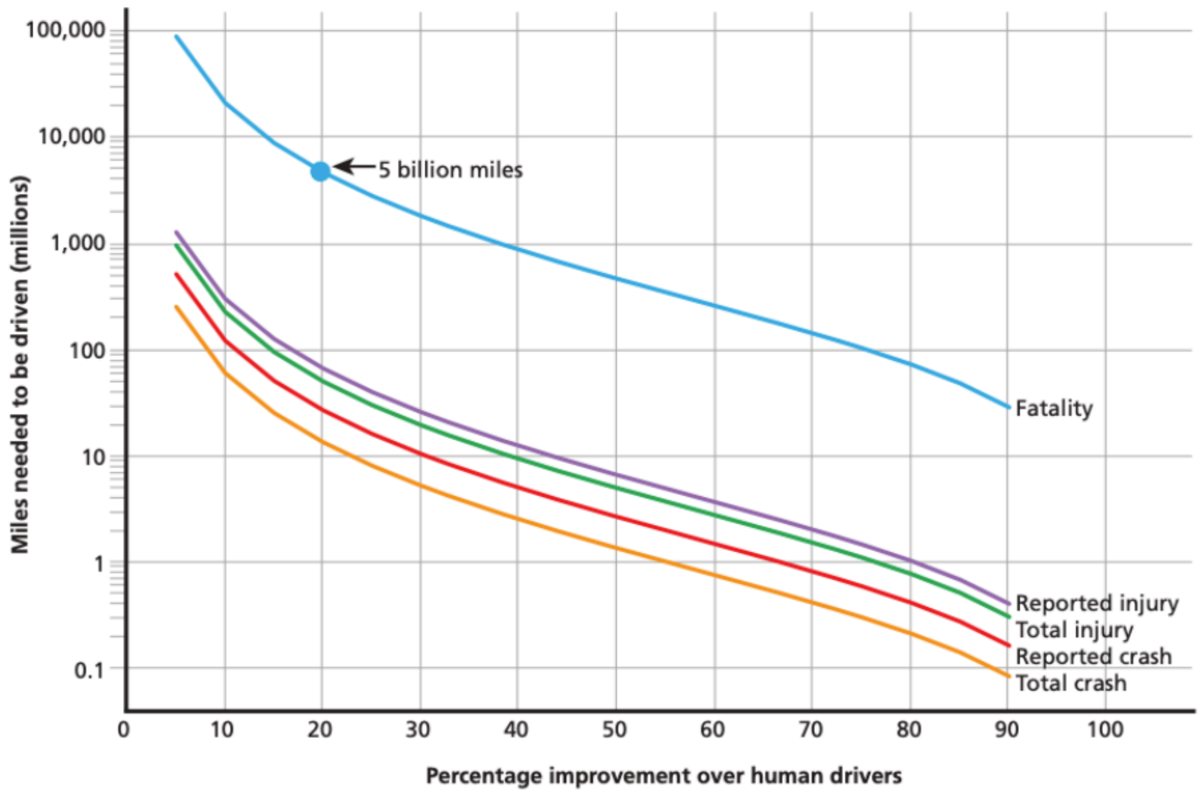
RAND CORPORATION

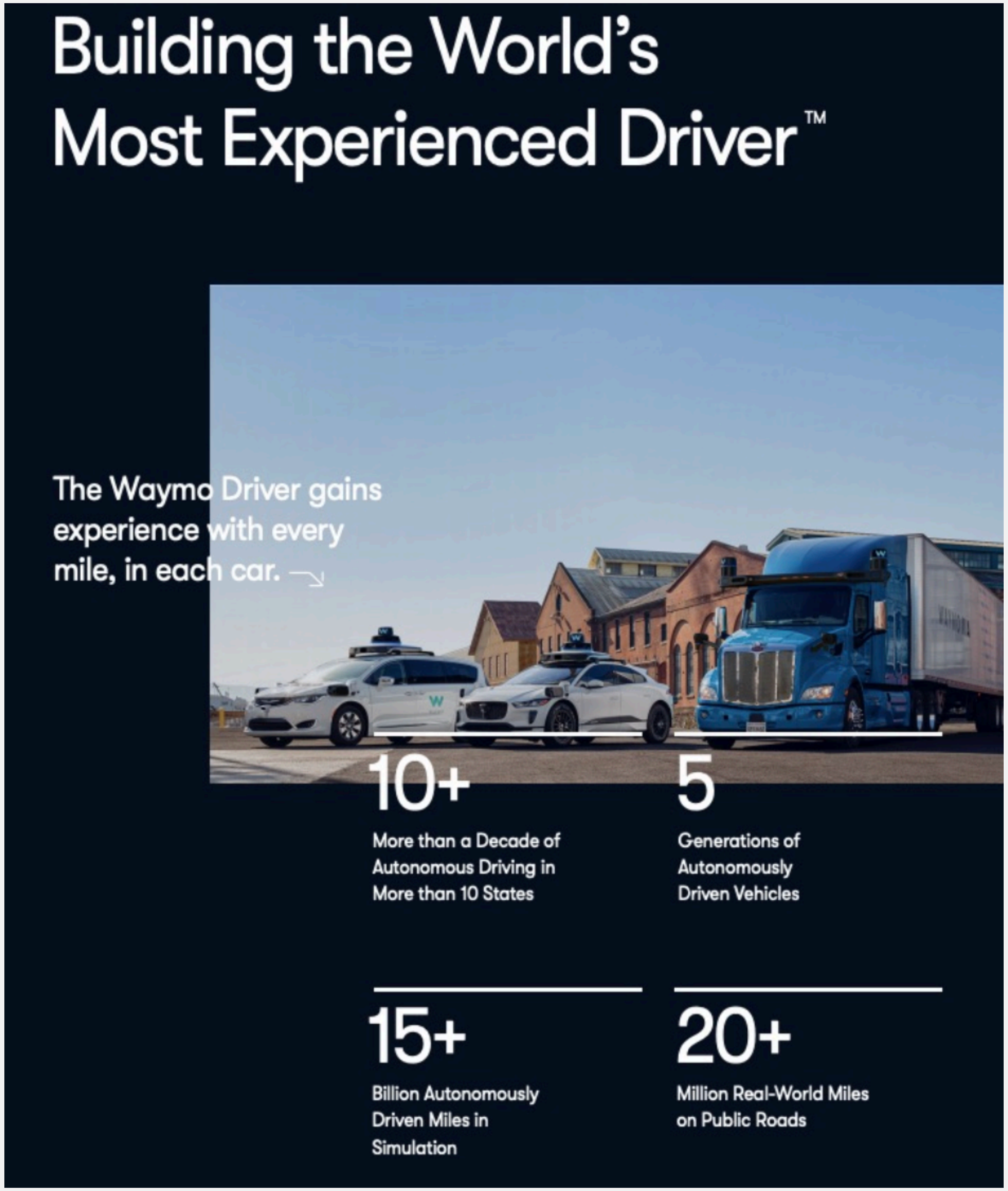
Driving to Safety

How Many Miles of Driving Would It Take to Demonstrate Autonomous Vehicle Reliability?

Nidhi Kalra, Susan M. Paddock

Figure 3. Miles Needed to Demonstrate with 95% Confidence that the Autonomous Vehicle Failure Rate Is Lower than the Human Driver Failure Rate





Building the World's Most Experienced Driver™

The Waymo Driver gains experience with every mile, in each car. →

10+

More than a Decade of Autonomous Driving in More than 10 States

5

Generations of Autonomously Driven Vehicles

15+

Billion Autonomously Driven Miles in Simulation

20+

Million Real-World Miles on Public Roads

Source: waymo.com/safety (September 2021)

What is Measurement?

- Measurement is the empirical, objective assignment of numbers, according to a rule derived from a model or theory, to attributes of objects or events with the intent of describing them. – Craner, Bond, “Software Engineering Metrics: What Do They Measure and How Do We Know?”

- A quantitatively expressed reduction of uncertainty based on one or more observations. – Hubbard, “How to Measure Anything ...”

Software Quality Metrics

- IEEE 1061 definition: “A software quality metric is a function whose inputs are software data and whose output is a single
numerical value that can be interpreted as the degree to which the software possesses a given attribute that affects its quality.”

- Metrics have been proposed for many quality attributes; may define own metrics

What Software Qualities Do We Care About?

- Functionality (e.g., data integrity)
- Scalability
- Security
- Extensibility
- Bugginess
- Documentation
- Performance
- Installability
- Availability
- Consistency
- Portability
- Regulatory compliance

What Process Qualities Do We Care About?

- On-time release
- Development speed
- Meeting efficiency
- Conformance to processes
- Time spent on rework
- Reliability of predictions
- Fairness in decision making
- Number of builds
- Code review acceptance rate
- Regulatory compliance

- Measure time, costs, actions, resources, and quality of work packages; compare with predictions
- Use information from issue trackers, communication networks, team structures, etc...

What People Qualities Do We Care About?

- Developers
  - Maintainability
  - Performance
  - Employee satisfaction and well-being • Communication and collaboration
  - Efficiency and flow
  - Satisfaction with engineering system • Regulatory compliance

- Customers
  - Satisfaction
  - Ease of use
  - Feature usage
  - Regulatory compliance

Everything is Measurable

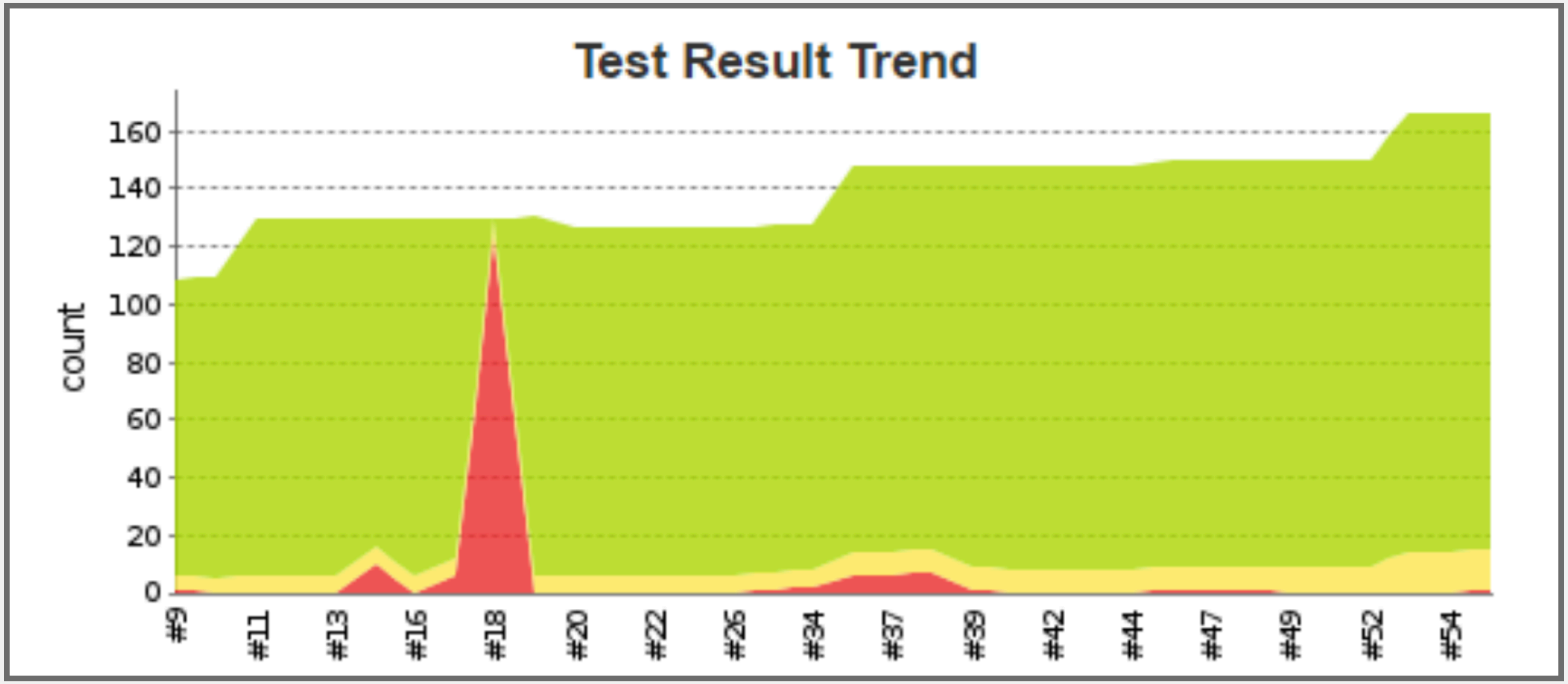
- If X is something we care about, then X, by definition, must be detectable.
  - How could we care about things like “quality,” “risk,” “security,” or “public image” if these things were totally undetectable, directly or indirectly?
  - If we have reason to care about some unknown quantity, it is because we think it corresponds to desirable or undesirable results in some way.
- If X is detectable, then it must be detectable in some amount.
  - If you can observe a thing at all, you can observe more of it or less of it 21
- If we can observe it in some amount, then it must be measurable.

Why Measure?

Measurement for Decision Making

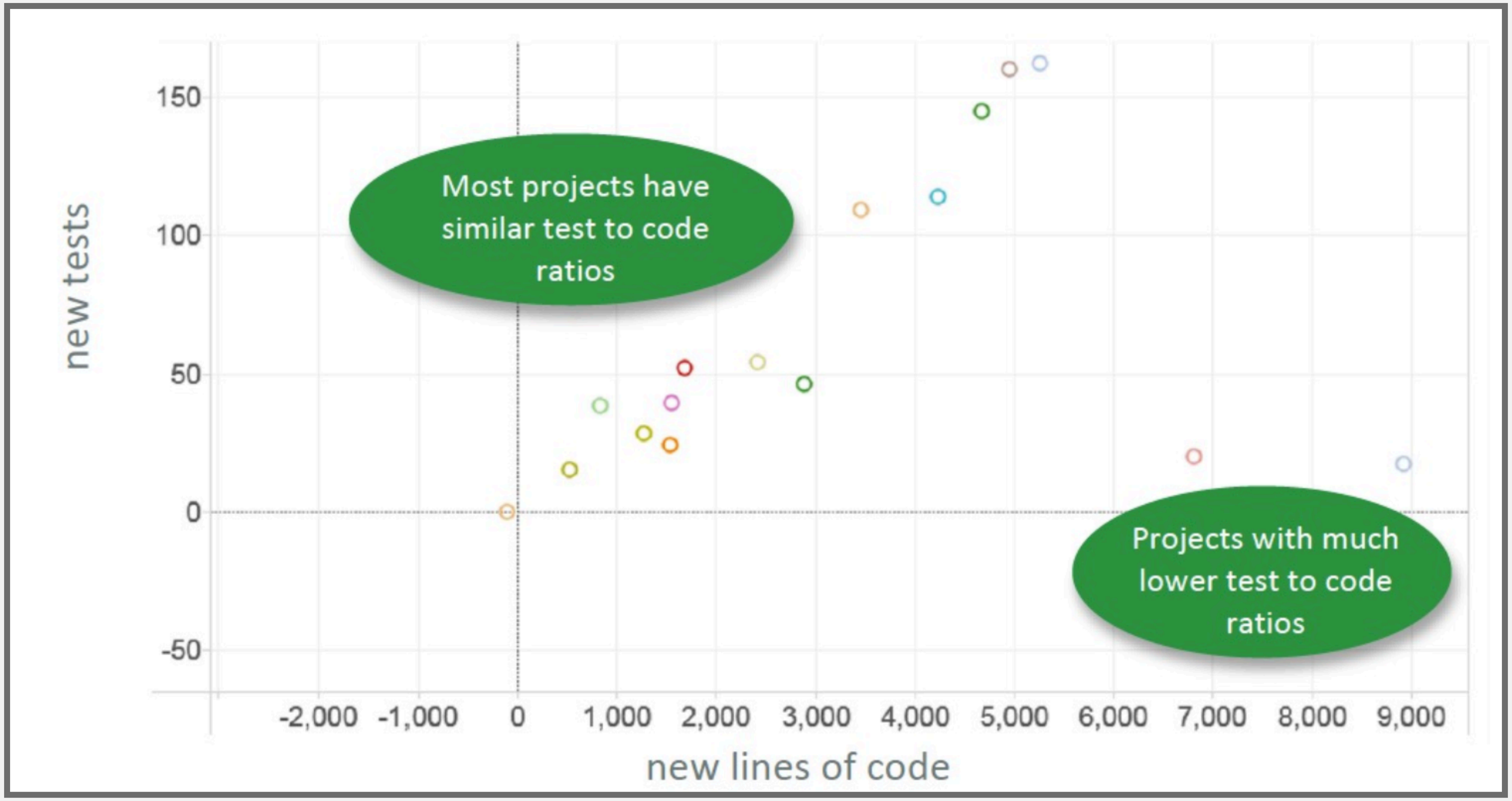
- Fund project?
- More testing?
- Fast enough? Secure enough?
- Code quality sufficient?
- Which feature to focus on?
- Developer bonus?
- Time and cost estimation? Predictions reliable?

Trend Analyses



Benchmarking Against Standards

- Monitor many projects or many modules, get typical values for metrics
- Report deviations

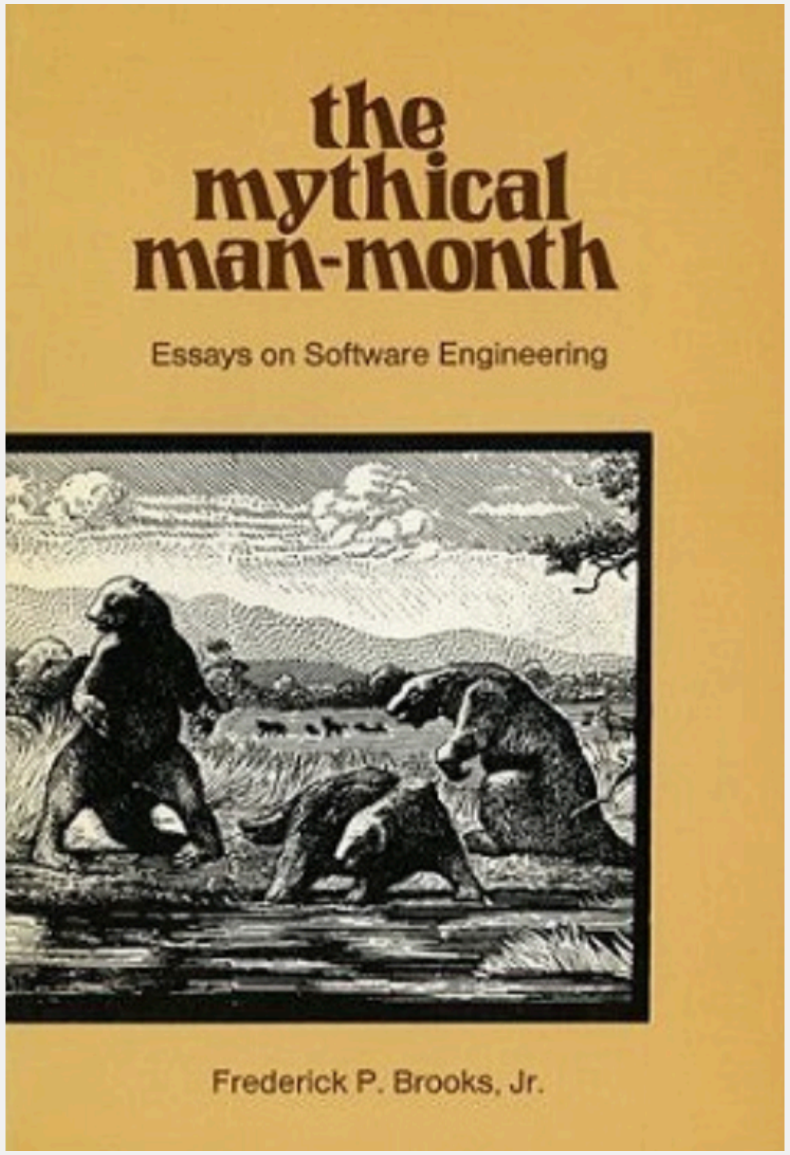


Most projects have similar test to code ratios

Projects with much lower test to code ratios

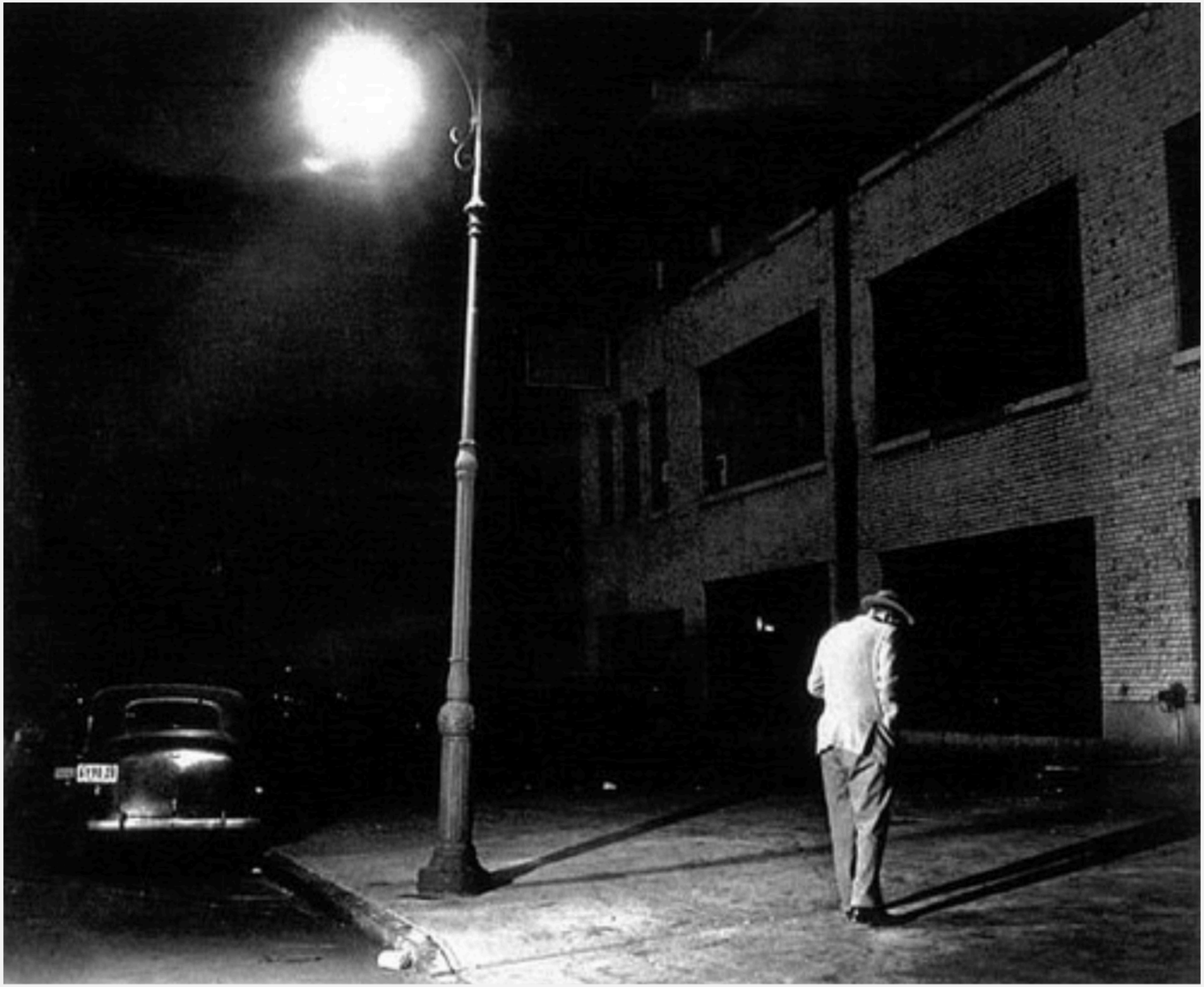
Antipatterns in Effort Estimation

- IBM in the 60s: Would account in “person-months”
  e.g. Team of 2 working 3 months = 6 person-months
- LoC ~ Person-months ~ $$$
- Brooks: “Adding manpower to a late software project [just] makes it later.”



Measurement is Difficult

The Streetlight Effect



The Streetlight Effect

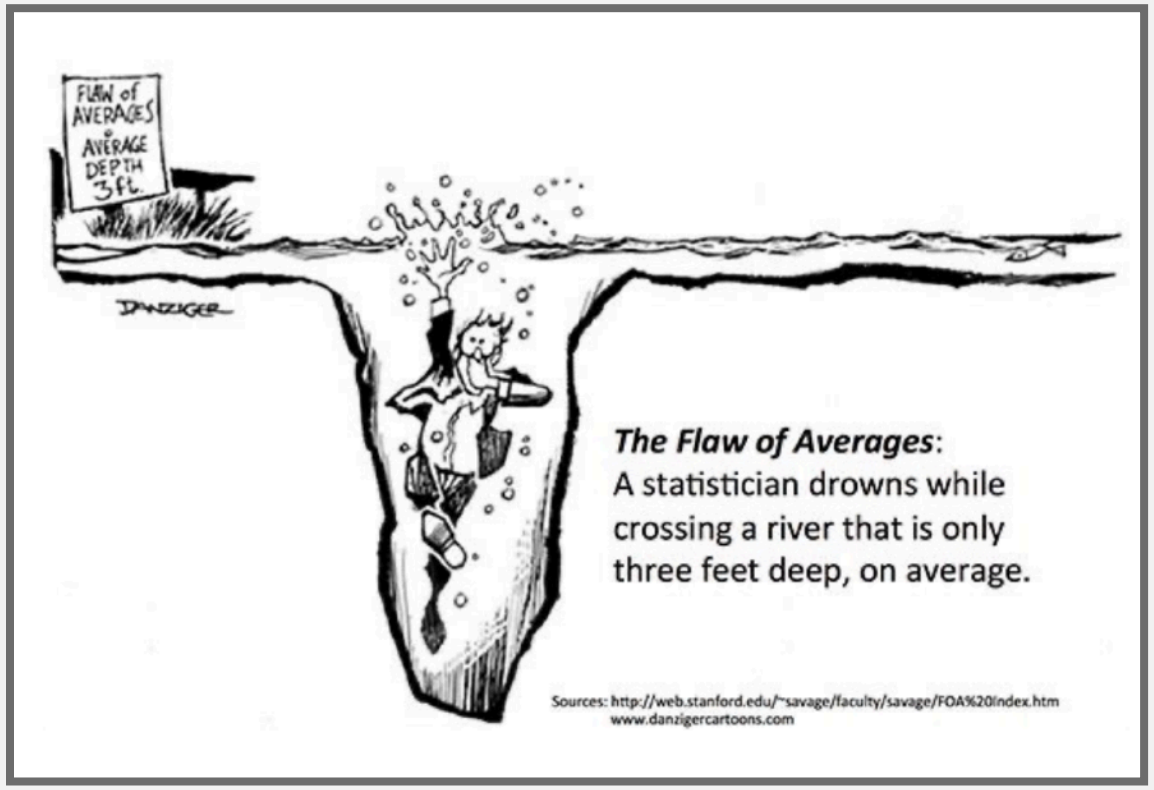
- A known observational bias.
- People tend to look for something only where it’s easiest to do so.
- If you drop your keys at night, you’ll tend to look for it under streetlights.

What could Possibly go Wrong?

- Bad statistics: A basic misunderstanding of measurement theory and what is being measured.

- Bad decisions: The incorrect use of measurement data, leading to unintended side effects.

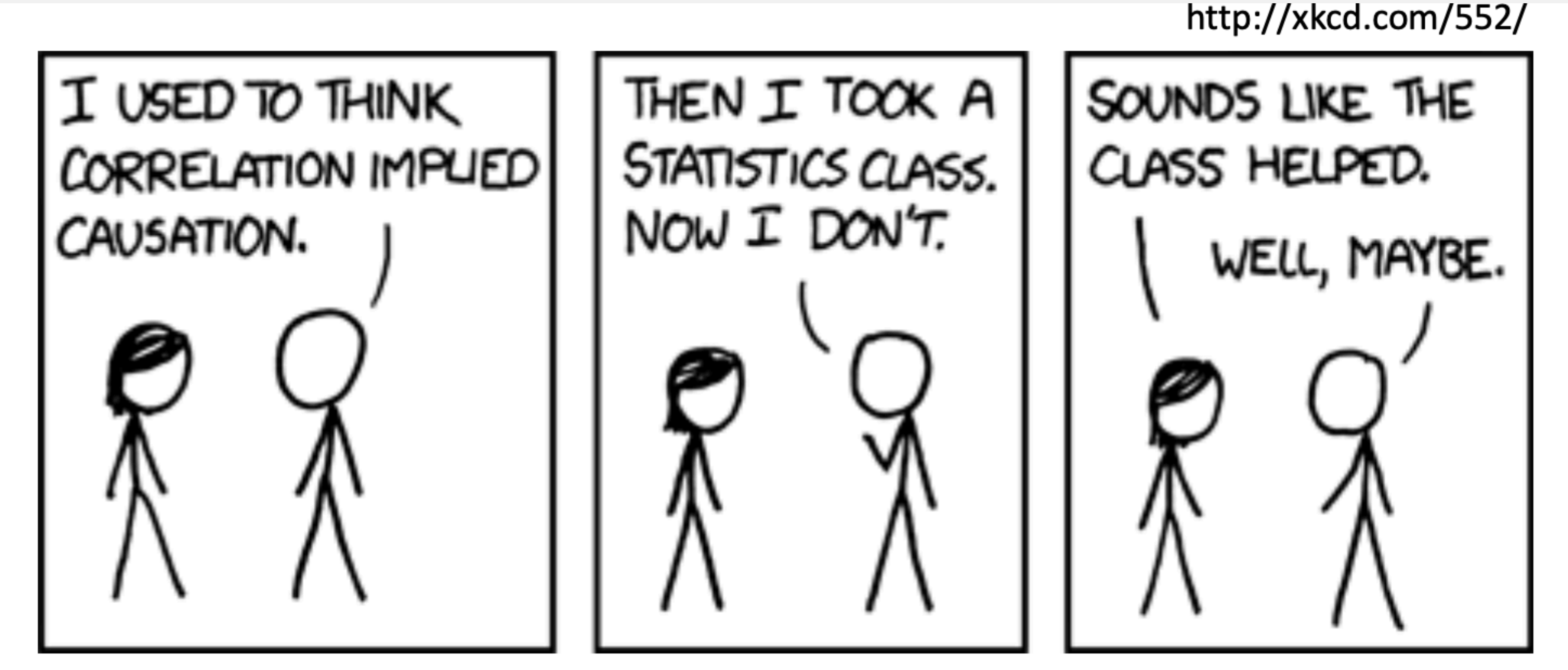
- Bad incentives: Disregard for the human factors, or how the cultural change of taking measurements will affect people.



The Flaw of Averages: A statistician drowns while crossing a river that is only three feet deep, on average.

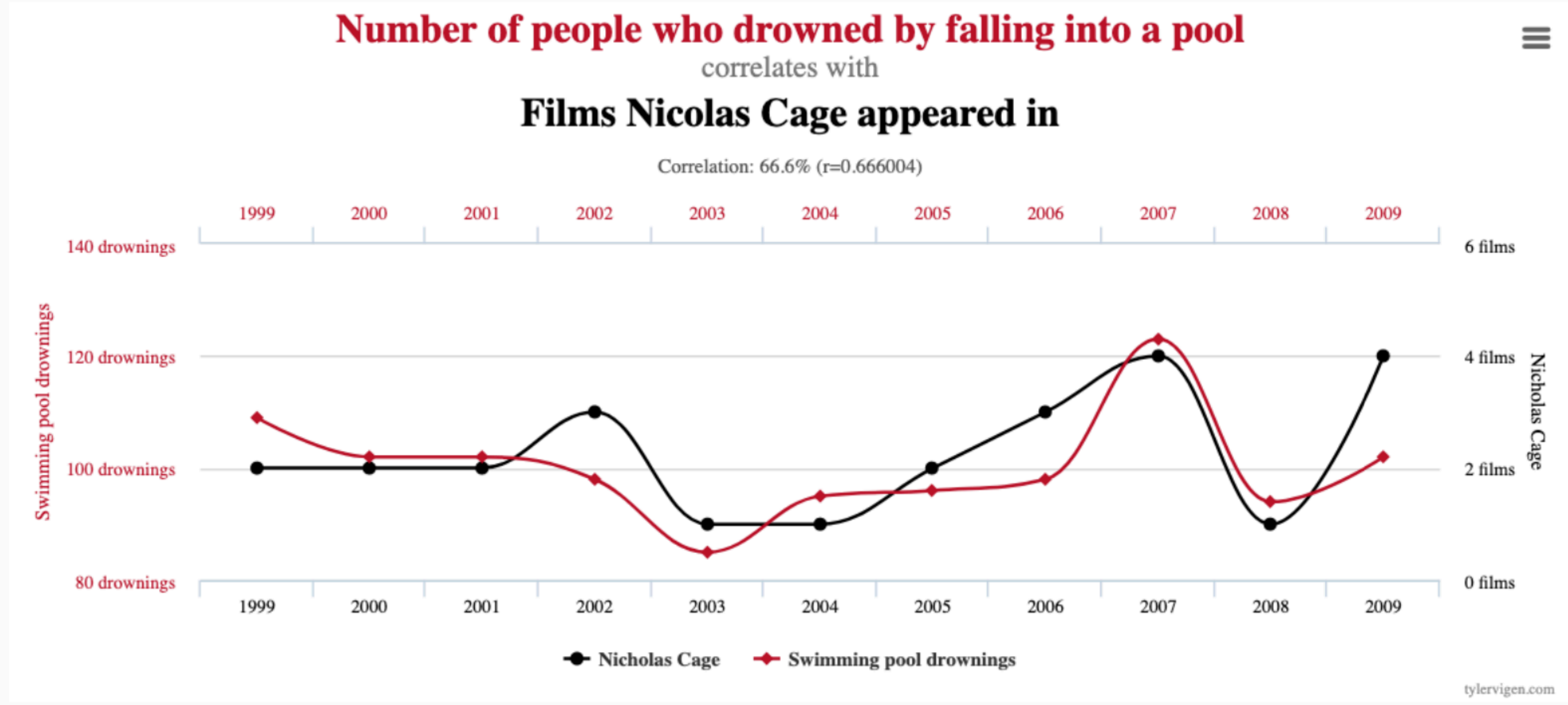
Sources: http://web.stanford.edu/~savage/faculty/savage/FOA%20Index.htm
www.danzigercartoons.com

Making Inferences


http://xkcd.com/552/
I USED TO THINK CORRELATION IMPLIED CAUSATION.
THEN I TOOK A STATISTICS CLASS. NOW I DON'T.
SOUNDS LIKE THE CLASS HELPED.
WELL, MAYBE.

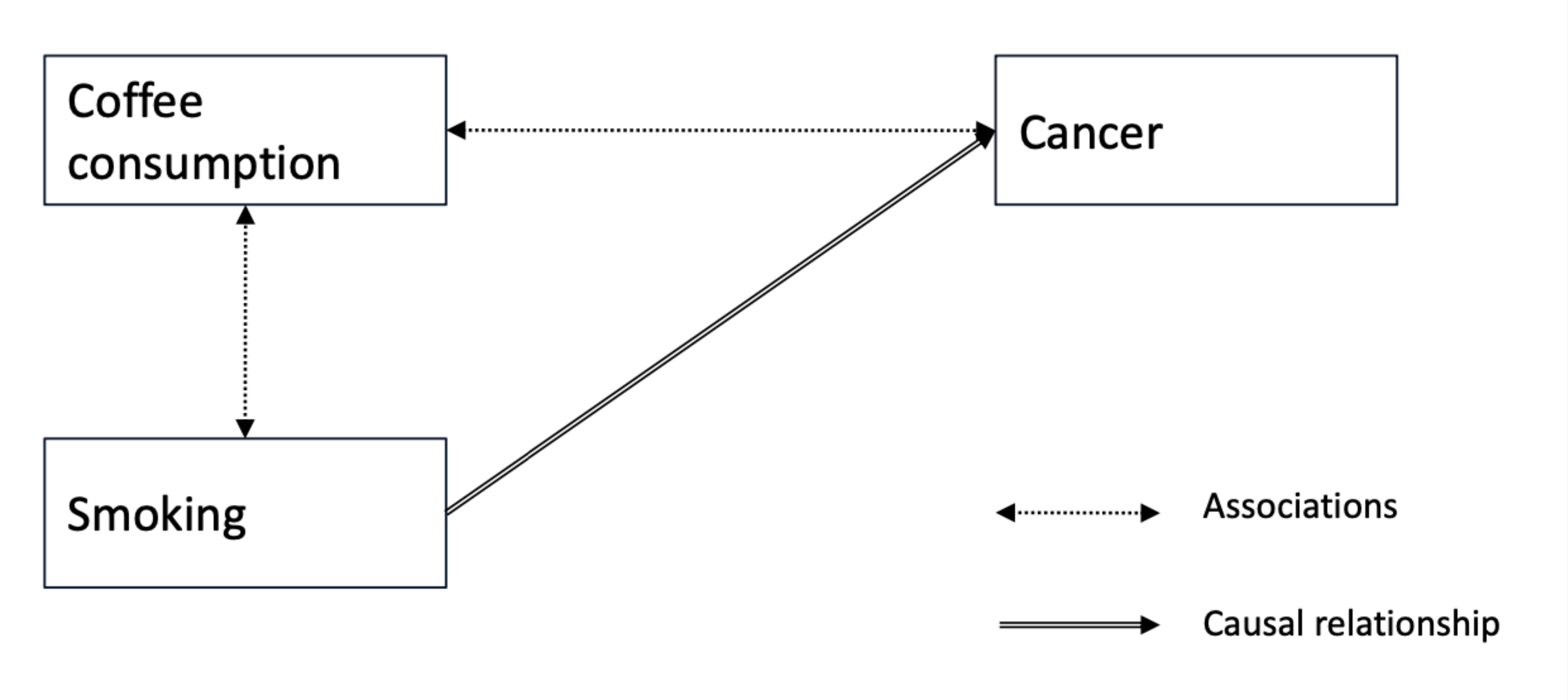
- To infer causation:
    - Provide a theory (from domain knowledge, independent of data)
    - Show correlation
    - Demonstrate ability to predict new cases (replicate/validate)

Spurious Correlations



Data sources: Centers for Disease Control & Prevention and Internet Movie Database

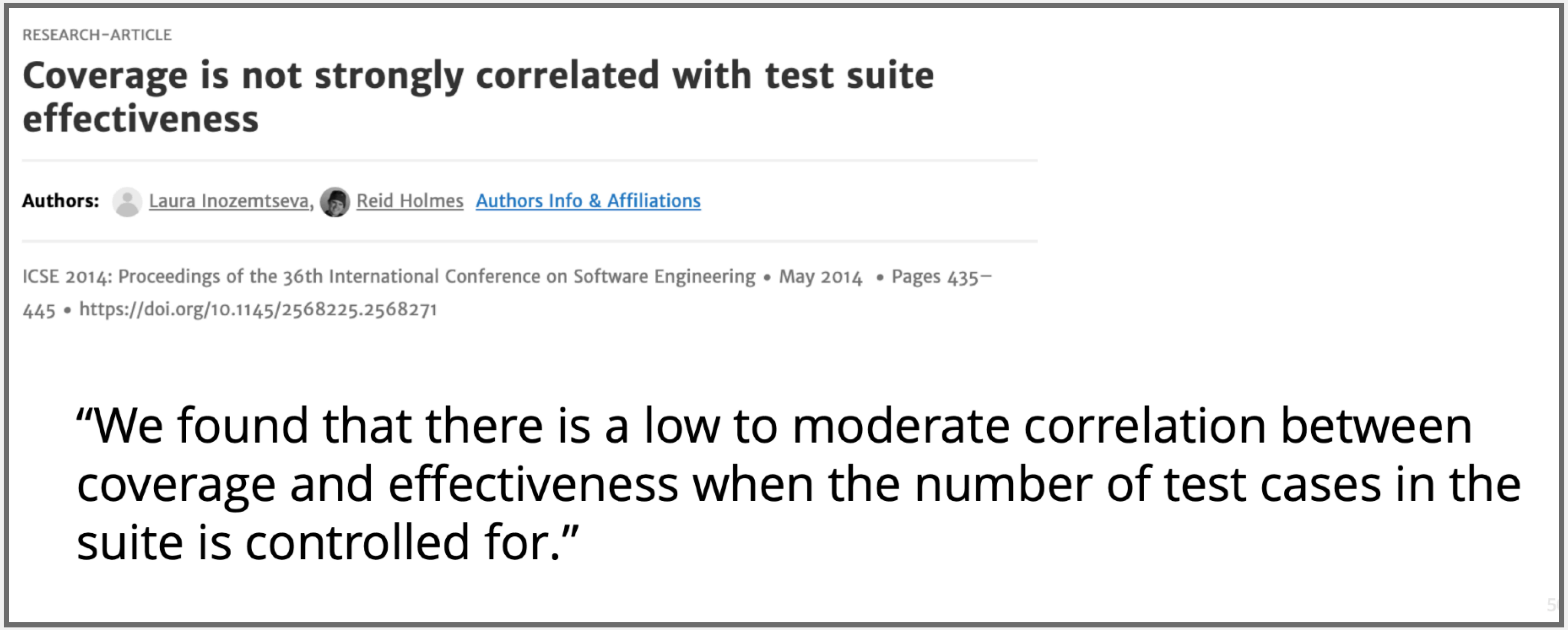
Confounding Variables



- If you look only at the coffee consumption → cancer relationship, you can get very misleading results

- Smoking is a confounder

SWE Research



RESEARCH-ARTICLE

Coverage is not strongly correlated with test suite effectiveness

Authors: Laura Inozemtseva, Reid Holmes Authors Info & Affiliations

ICSE 2014: Proceedings of the 36th International Conference on Software Engineering • May 2014 • Pages 435–445 • https://doi.org/10.1145/2568225.2568271

"We found that there is a low to moderate correlation between coverage and effectiveness when the number of test cases in the suite is controlled for."

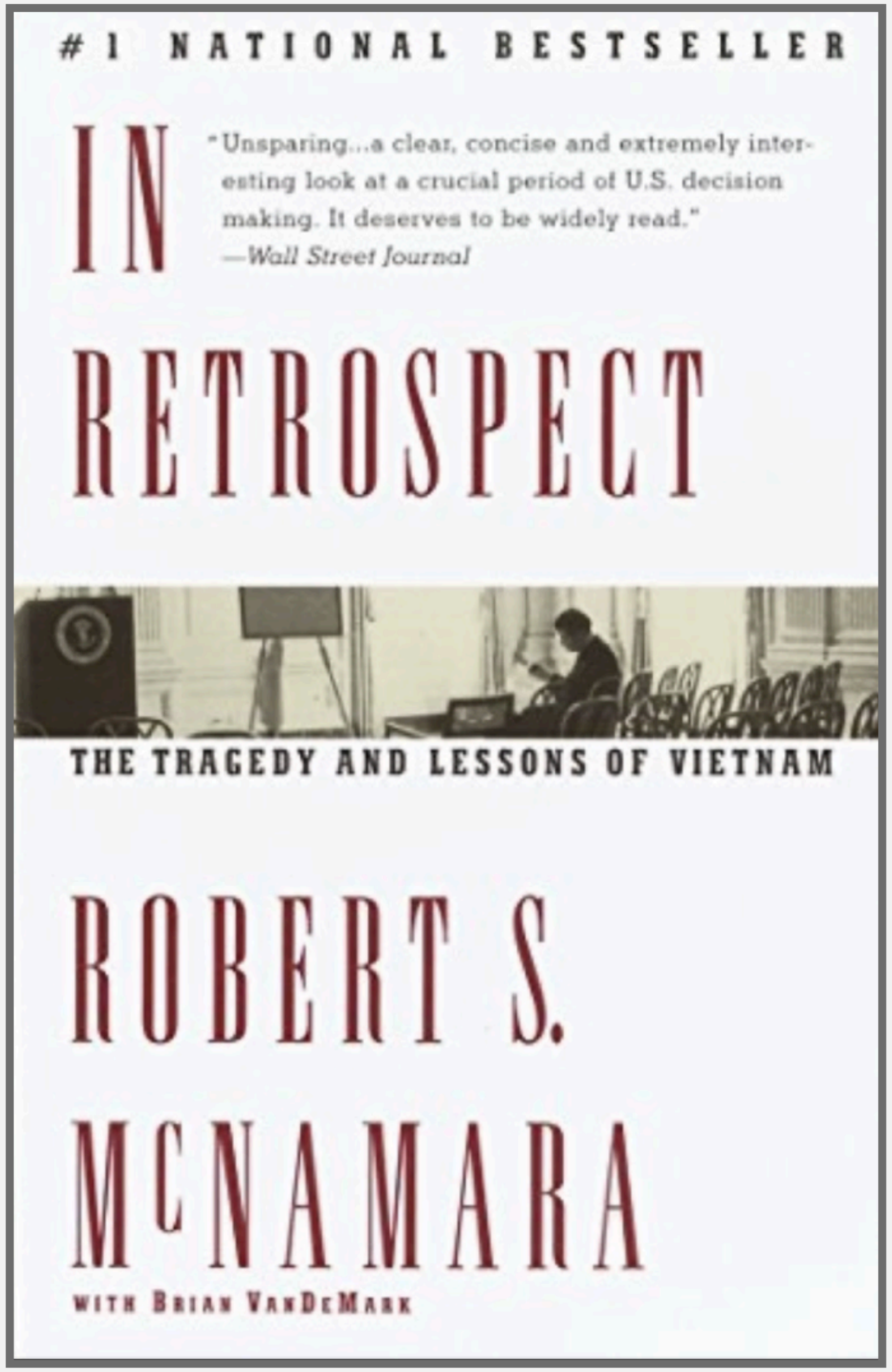
Measurements Validity

- **Construct validity** – Are we measuring what we intended to measure?
- **Internal validity** – The extent to which the measurement can be used to explain some other characteristic of the entity being measured
- **External validity** – Concerns the generalization of the findings to contexts and environments, other than the one studied

Measurements Reliability

- Extent to which a measurement yields similar results when applied multiple times
- Goal is to reduce uncertainty, increase consistency
- Example: Performance
  - Time, memory usage
  - Cache misses, I/O operations, instruction execution count, etc.
- Law of large numbers
  - Taking multiple measurements to reduce error
- Trade-off with cost

McNamara Fallacy



#1 NATIONAL BESTSELLER

IN RETROSPECT

"Unsparing...a clear, concise and extremely interesting look at a crucial period of U.S. decision making. It deserves to be widely read."
—*Wall Street Journal*

THE TRAGEDY AND LESSONS OF VIETNAM

ROBERT S. McNAMARA

WITH BRIAN VANDEMARK

McNamara Fallacy

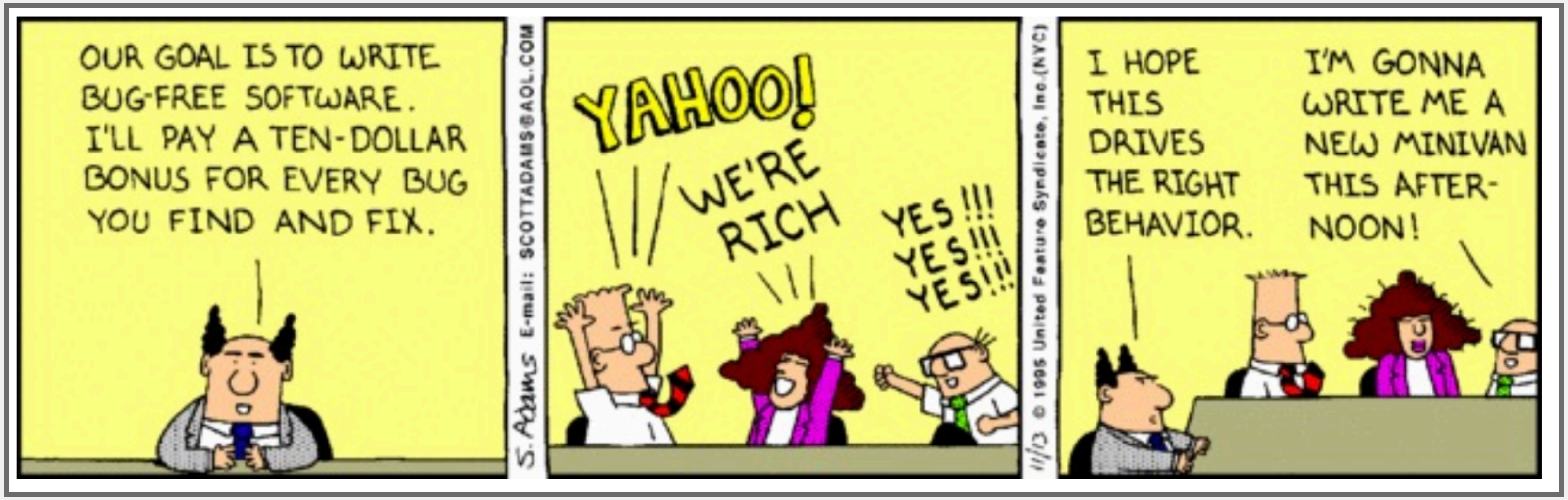
- Measure whatever can be easily measured.
- Disregard that which cannot be measured easily.
- Presume that which cannot be measured easily is not important.
- Presume that which cannot be measured easily does not exist.

McNamara Fallacy

- There seems to be a general misunderstanding to the effect that a mathematical model cannot be undertaken until every constant and functional relationship is known to high accuracy. This often leads to the omission of admittedly highly significant factors (most of the “intangibles” influences on decisions) because these are unmeasured or unmeasurable. To omit such variables is equivalent to saying that they have zero effect... Probably the only value known to be wrong...
    - J. W. Forrester, Industrial Dynamics, The MIT Press, 1961

Metrics & Incentives

- Goodhart's law: "When a measure becomes a target, it ceases to be a good measure."


OUR GOAL IS TO WRITE BUG-FREE SOFTWARE. I'LL PAY A TEN-DOLLAR BONUS FOR EVERY BUG YOU FIND AND FIX.
S. Adams E-mail: SCOTTADAMS@AOL.COM
YAHOO!
WE'RE RICH
YES!!! YES!!! YES!!!
11/13 © 1995 United Feature Syndicate, Inc.(NYC)
I HOPE THIS DRIVES THE RIGHT BEHAVIOR.
I'M GONNA WRITE ME A NEW MINIVAN THIS AFTER-NOON!

Simplistic Productivity Measures

- Lines of code per day?
  - Industry average 10-50 lines/day
  - Debugging + rework ca. 50% of time

- • Function/object/application points per month • Bugs fixed?
  - Milestones reached?

Incentivizing Productivity

- What happens when developer bonuses are based on
  - Lines of code per day?
  - Amount of documentation written?
  - Low number of reported bugs in their code?
  - Low number of open bugs in their code?
  - High number of fixed bugs?
  - Accuracy of time estimates?

Developer Productivity Myths

- Productivity is all about developer activity
- Productivity is only about individual performance
- One productivity metric can tell us everything
- Productivity measures are useful only for managers
- Productivity is only about engineering systems and developer tools

WARNING!!

- Most software metrics are controversial
  - Usually only plausibility arguments, rarely rigorously validated
  - Cyclomatic complexity was repeatedly refuted, yet is still used
  - “Similar to the attempt of measuring the intelligence of a person in terms of the weight or circumference of the brain”
- Use carefully!
- Code size dominates many metrics
- Avoid claims about human factors (e.g., readability) and quality, unless validated
- Calibrate metrics in project history and other projects
- Metrics can be gamed; you get what you measure

Summary

- Measurement is difficult but important for decision making
- Software metrics are easy to measure but hard to interpret,
  validity often not established
- Many metrics exist, often composed; pick or design suitable metrics if needed
- Careful in use: monitoring vs incentives
- Strategies beyond metrics

Questions to Consider for Your Projects

- What properties do we care about and how do we measure them?
- What is being measured? Does it (to what degree) capture the thing you care about? What are its limitations?
- How should it be incorporated into process?
- What are potentially negative side effects or incentives?