

Code review is an architectural necessity

Colin Dean
@colin dean

@ColinDean

Software Engineer

Organizer, Abstractions.io

Wearer of many hats



My words are my own and not my employer's, past or present.

Please save questions until the end of the presentation.

Agenda

- Quick anecdote
- What is code review?
- What problems does code review solve?
- Quality attributes code review ensures
- Tips for code reviews
- Limitations



The Holcad

WESTMINSTER COLLEGE'S STUDENT NEWSPAPER SINCE 1884

A three-year voyage concludes



Colin Dean
Editor-in-chief

It's been a few months short of four years since the first time I picked up a copy of *The Holcad*. After reading a few editions, as I recall, I decided that I wanted to get involved.

I had been the "computer productions manager" for Wilmington's Blue and Gold, the position charged with maintaining the 20-computer journalism lab. I was expected to know how to do essentially everything needed to assemble "The Extra," a letter-size section for the majority of the articles, as only a few were selected to go in the main newsprint portion of the high school paper.

I visited co-editors-at-the-time Megan Sigler and Josh Storey at the new *Holcad* office in the recently-built McKelvey Campus Center.

With my previous experience at the Blue and Gold, I hoped to be able to score at least a photography position with *The Holcad*, but something involving layout would have been really cool.

Unfortunately, Sigler and Storey told me that freshmen cannot be on staff of *The Holcad* for a number of reasons I won't waste space listing. Instead, they suggested that I try Argo and apply for *Holcad* staff in the spring of 2004. I did both, and found myself eventually working for *The Holcad* as a layout and copy editor for the 2004-2005 academic year.

Sigler was more-or-less responsible for my interest in journalism

editor last semester. She studied abroad in Spain this semester, but will be returning to staff next year. She's another layout warrior and source of many layout staff inside jokes.

Leanne Stitt is our news editor, the one responsible for all news stories. Each week, she scours the campus for story ideas. She decides what's newsworthy, what's rumor and what's superfluous. She's the heart and soul of what the readers read. Without her, *The Holcad* couldn't cover the breadth or depth of information that it does.

Jenn Corkadel is my main managing editor and editor of the Features section. She's responsible for the day-to-day operations of *The Holcad*. In addition to layout and writing a column, finding content worthy for Features, she also acts as my foil; she helps keep me in reality. She probably spends as much time in the office as I do—if not more—receiving stories from writers and forwarding them to copy editors, designing page layouts with Leanne and me, managing *The Holcad*'s budget. One of her many specialties is making sure that layout days are never without an appropriate amount of sarcasm and humor.

I got a lot of credit for the success of *The Holcad*, and I've also caught the heat when we've messed up. However, there are a lot of people behind the scenes who rarely get their names in the paper except for a column here and there or a few characters in the staff box at the bottom of this page.

David Timm is our distribution manager. He's the guy who wakes up before most students do each Friday so that he can get *Holcad*s and take them to the pick-up spots throughout campus.

He

know I needed them. And so, trusty readers, this is my disembarkation point. I've been voyaging on this vessel of burden for three long years, devoting every Thursday, and lots more time during other days, to bring you the news of Westminster. For the past year, I've been sailing the seas of journalism with spyglass and compass in hand and in the company of my capable first mate, helmsman and the finest crew I could as-

semble.

Land ho, mates.

Don't miss Colin's senior send-off bio on Page B-4.

Shondie bids grads adieu



Bill Shondelmyer
Piece of mind

This semester is finally over and with it, my time as opinion columnist. It has been a blast, but I cannot help but wonder if I fell into the same rut as Peck.

It seemed that by the end of his time on the job, he came to the realization that angering people was the most he could accomplish. I think I vaguely remember a conversation we had on the topic.

I understand now what he meant. By this point, I have written at least half a dozen columns on what it is to be a "good" person. Conscience, caring, compassion, rhetoric, opinion... I really was just preaching to the choir.

But at least I was entertaining, right?

I think I may have gotten a rise out of one or two people, though. That makes my day. Rattling the cage and making people a little anxious might just have made them more careful or more

tempted to submit my articles before Thursday afternoon...

Next year should be interesting, with a new staff and a new A-2. Well, no more me, at least. I will still be around, as I am still waiting to get tenure, but I will not be writing as much.

To all the seniors: congratulations.

To all the juniors: welcome to the end of the journey.

To all the sophomores: you are halfway done; it is all downhill



Agenda

- Quick anecdote
- **What is code review?**
- What problems do code review solve?
- Quality attributes code review ensures
- Tips for code reviews
- Limitations

What is code review?

Code review is the process by which those who maintain a software codebase evaluate a proposed change to that codebase, regardless of the source of the proposed change.

Code review is **systematic examination** of computer source code.

Peer Review

Code Review

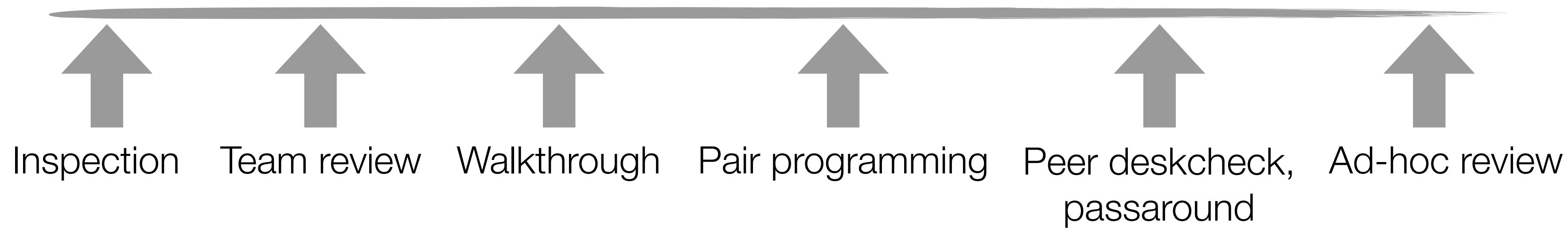
Code Review Vocabulary

- **Change** - an individual unit of work altering what exists
- **Submission** - a collection of changes
- **Submitter** - the person proposing the submission
- **Reviewer** - the people evaluating the submission
- **Annotation** - remarks or ratings bestowed upon the submission

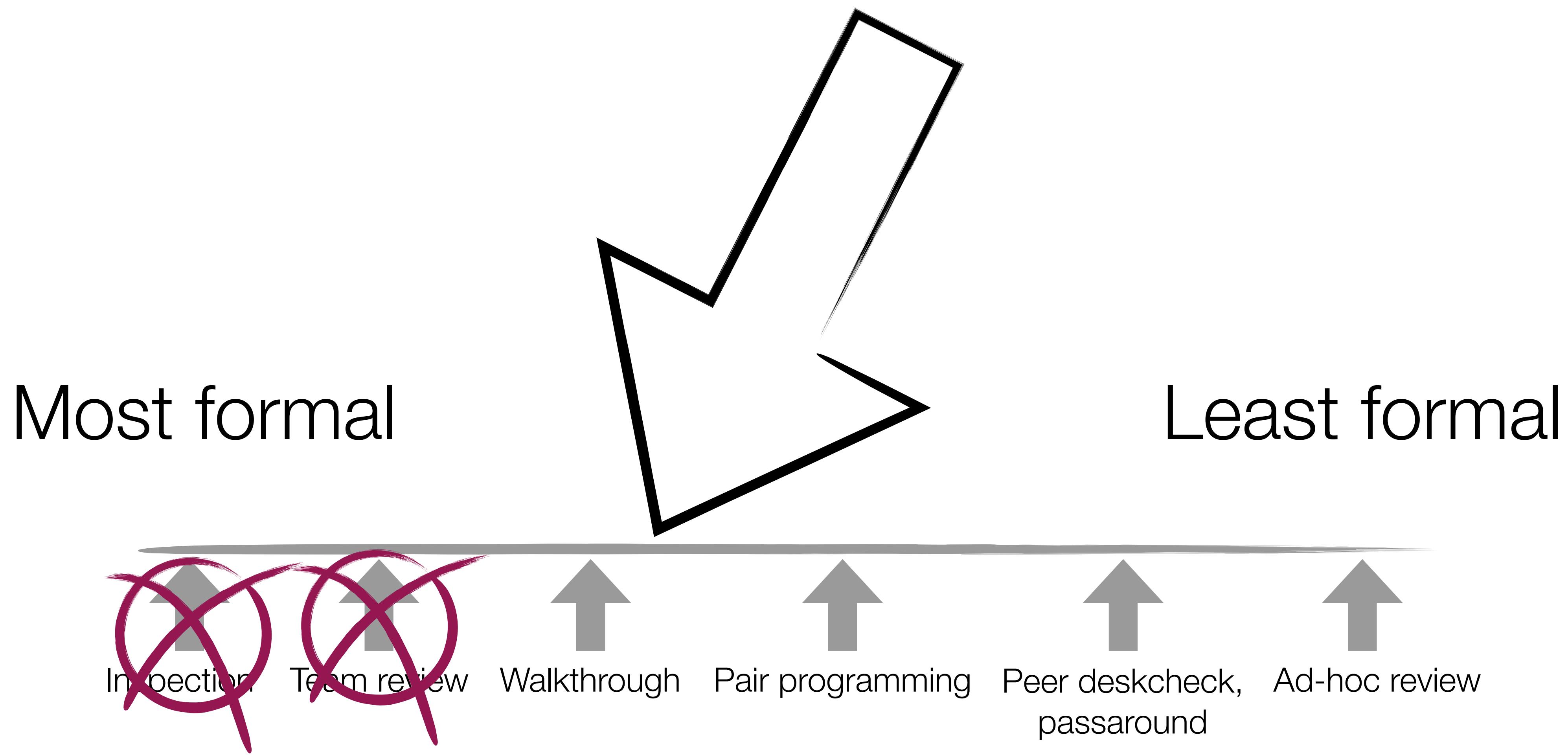
The **submitter** proposes **changes** in a **submission**, which is evaluated by a **reviewer**, who **annotates** or accepts it.

Most formal

Least formal



Wiegers' peer review formality spectrum



Wiegers' peer review formality spectrum

GitHub

```
108 -     var timestamp = [date.getHours(), date.getMinutes(),
date.getSeconds()].join(":");
```

```
108 +     var timestamp = [('0'+date.getHours()).slice(-2),
('0'+date.getMinutes()).slice(-2), ('0'+date.getSeconds()).slice(-2)].join(":");
```

 colin dean added a note 21 days ago PittMesh, a MetaMesh project member +😊 🖊 ✖

Extract a function here:

```
function pad(what) { return ('0'+what).slice(-2); }
```

 colin dean added a note 21 days ago PittMesh, a MetaMesh project member +😊 🖊 ✖

Then you can do `[date.getHours(), date.getMinutes(),
date.getSeconds()].map(pad).join(":")`.

Add a line note

```
109     var message = "[" + timestamp + "] " + text;
110
111     transcriptEl.appendFirst(createLogLineElement(message, style));
112 -     console.log("%c"+message, style);
```

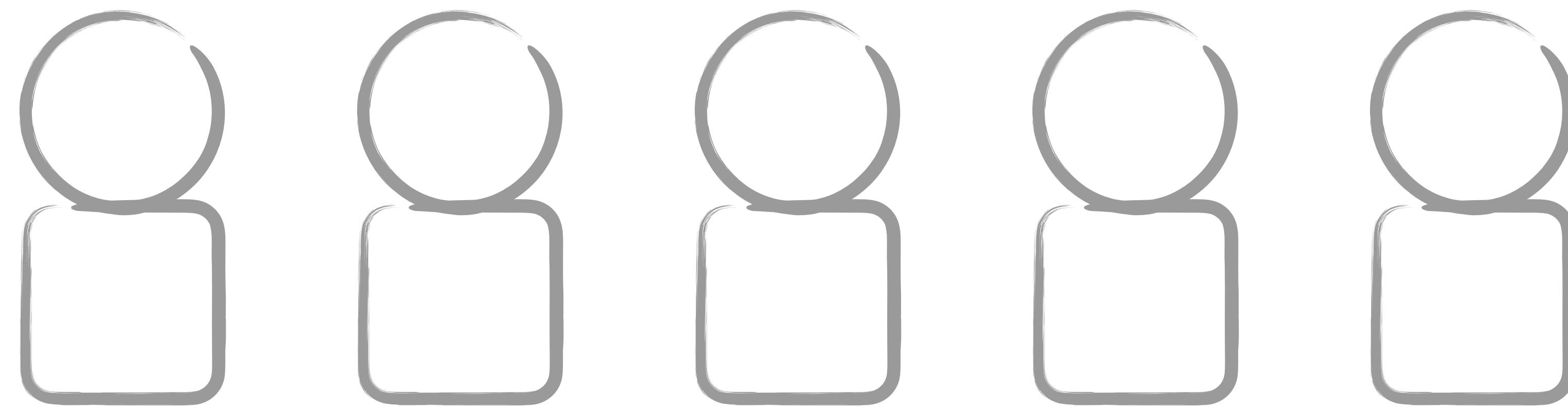
```
109     var message = "[" + timestamp + "] " + text;
110
111     transcriptEl.appendFirst(createLogLineElement(message, style));
112 +     console.log("[%d:%d:%d] %c"+text, date.getHours(), date.getMinutes(),
date.getSeconds(), style);
```

Agenda

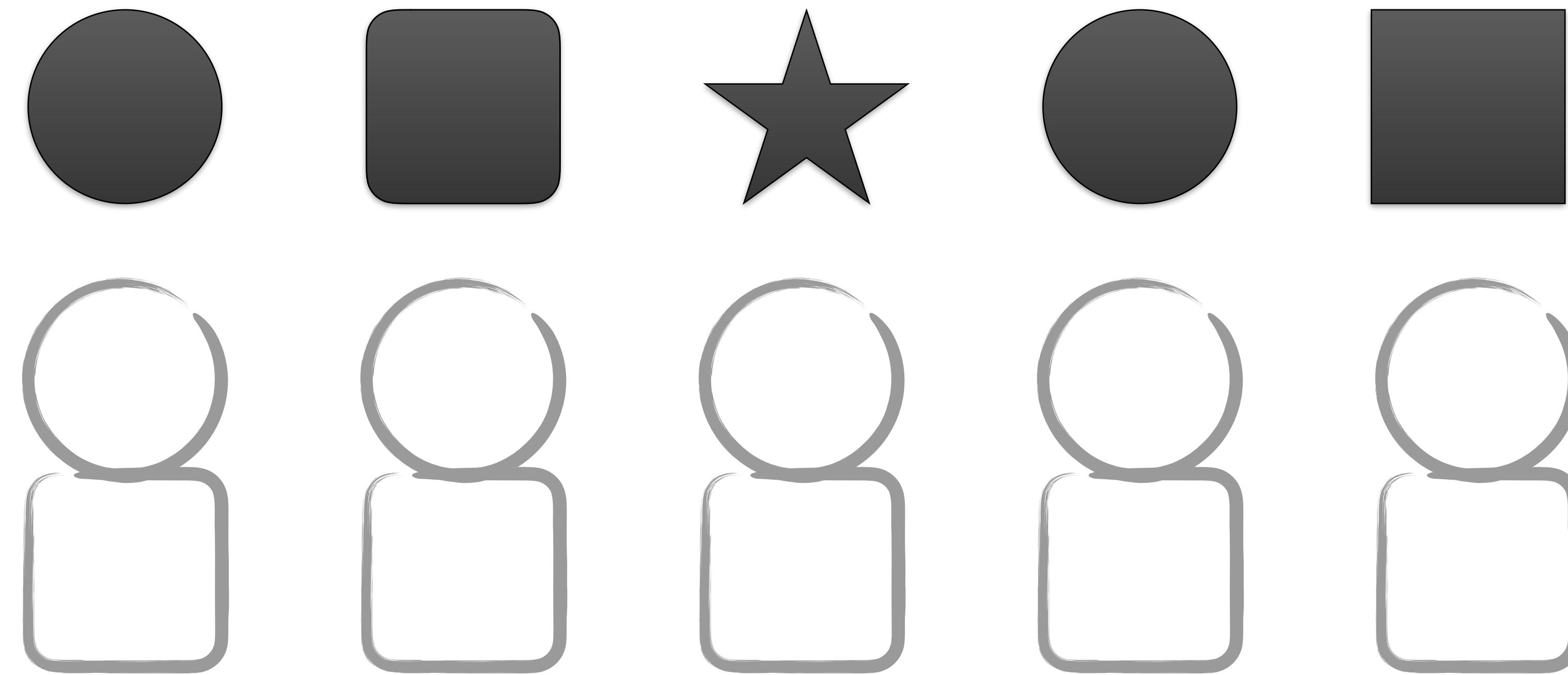
- Quick anecdote
- What is code review?
- **What problems does code review solve?**
- Quality attributes code review ensures
- Tips for code reviews
- Limitations

Aside from the primary goal of reducing defects,
Code review solves two major problems.

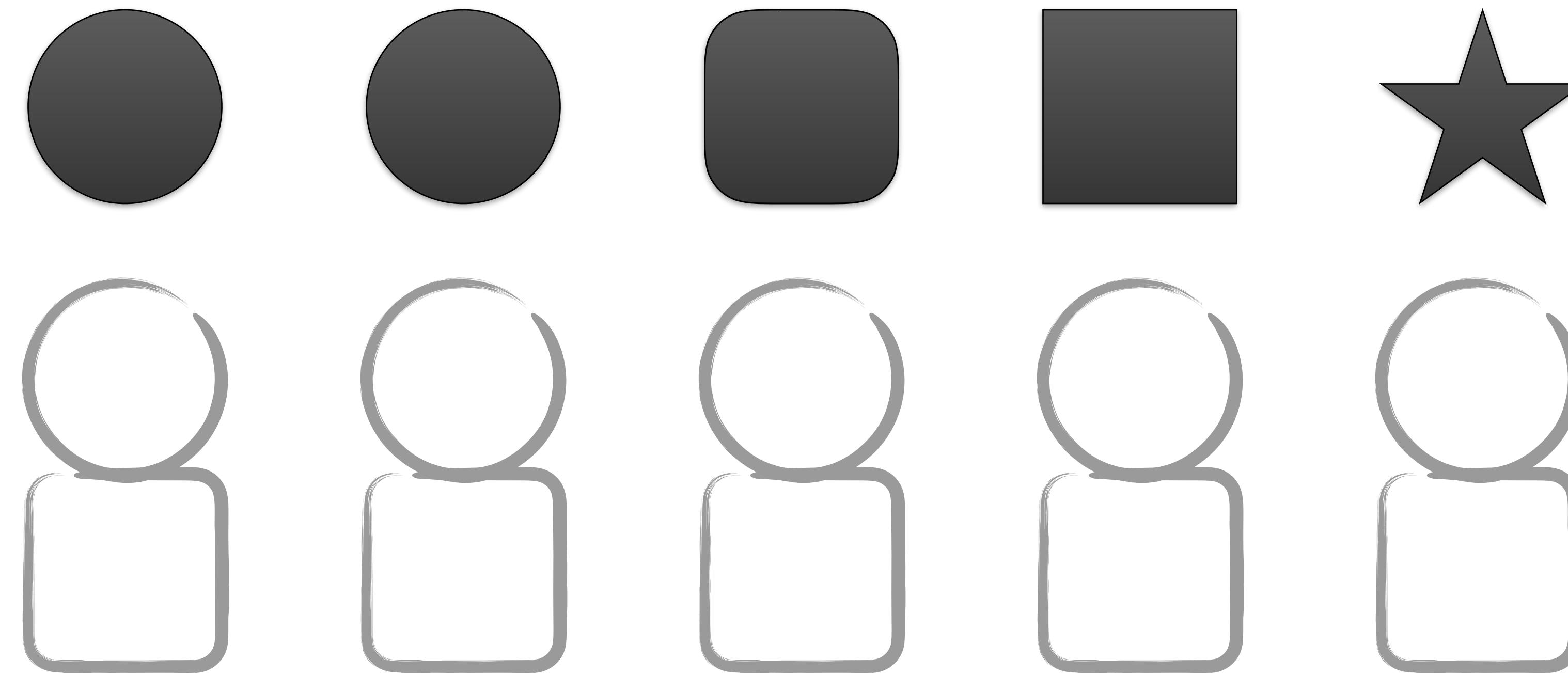
Code review solves
Mental model synchronization



#githubuniverse #codereview



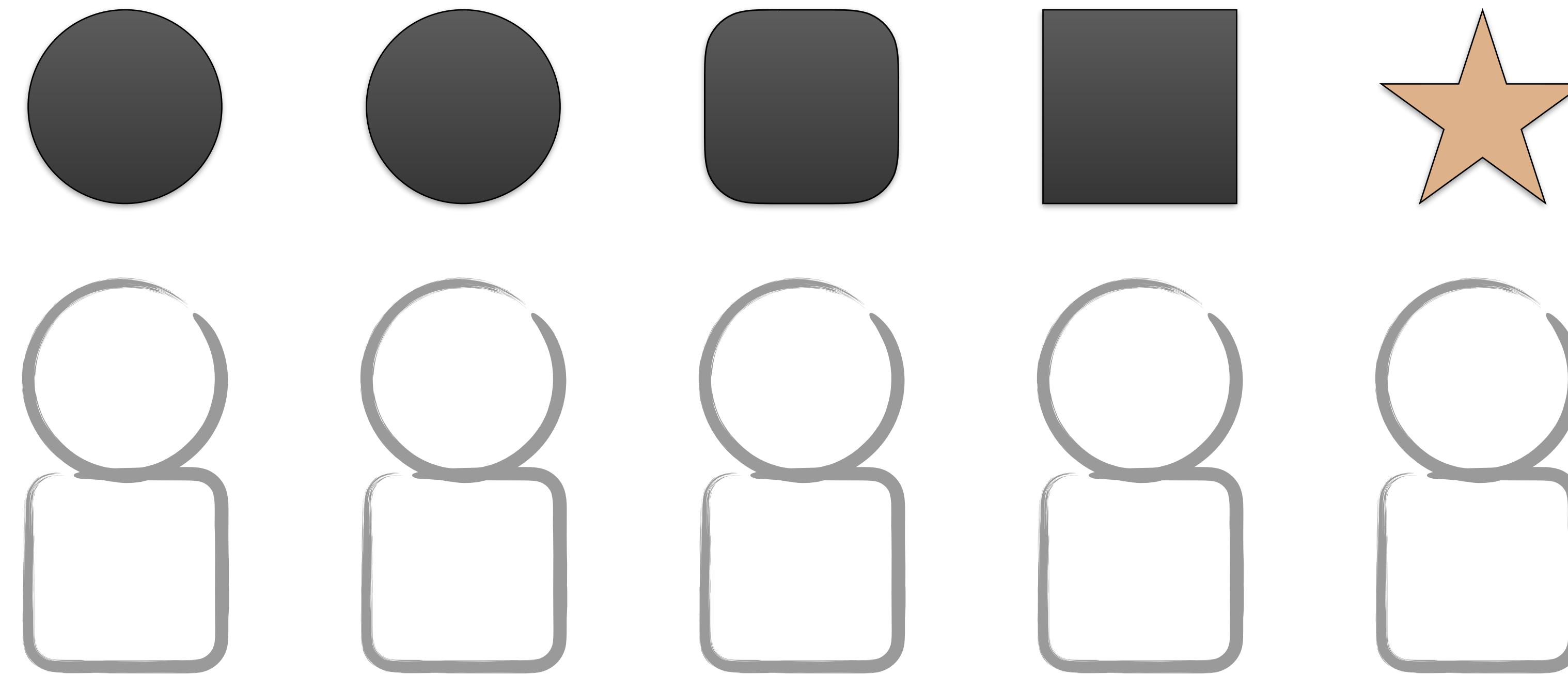
#githubuniverse #codereview



On target

Close enough

Need guidance



Open to new idea

Alternative solutions

New idea

Code review solves
Tribal knowledge development

“Architecture oral history requires
that the team is both willing and
able to retell the stories and keep
the oral history alive.”

Michael Keeling

Creating an Architecture Oral History, SATURN 2012

Code review audits
Application code

Code review audits
Application infrastructure

Code review audits
Application architecture

Code review forces us to
Write it down.
Make it searchable.

Agenda

- Quick anecdote
- What is code review?
- What problems does code review solve?
- **Quality attributes code review ensures**
- Tips for code reviews
- Limitations

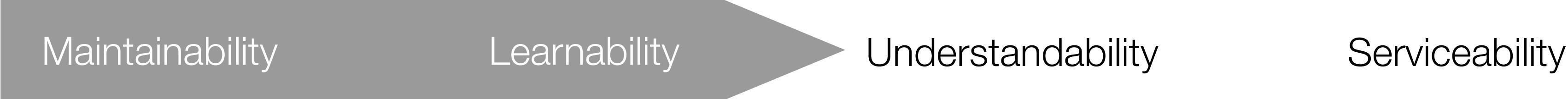
Code review ensures
Maintainability

Code review drives
Maintainability

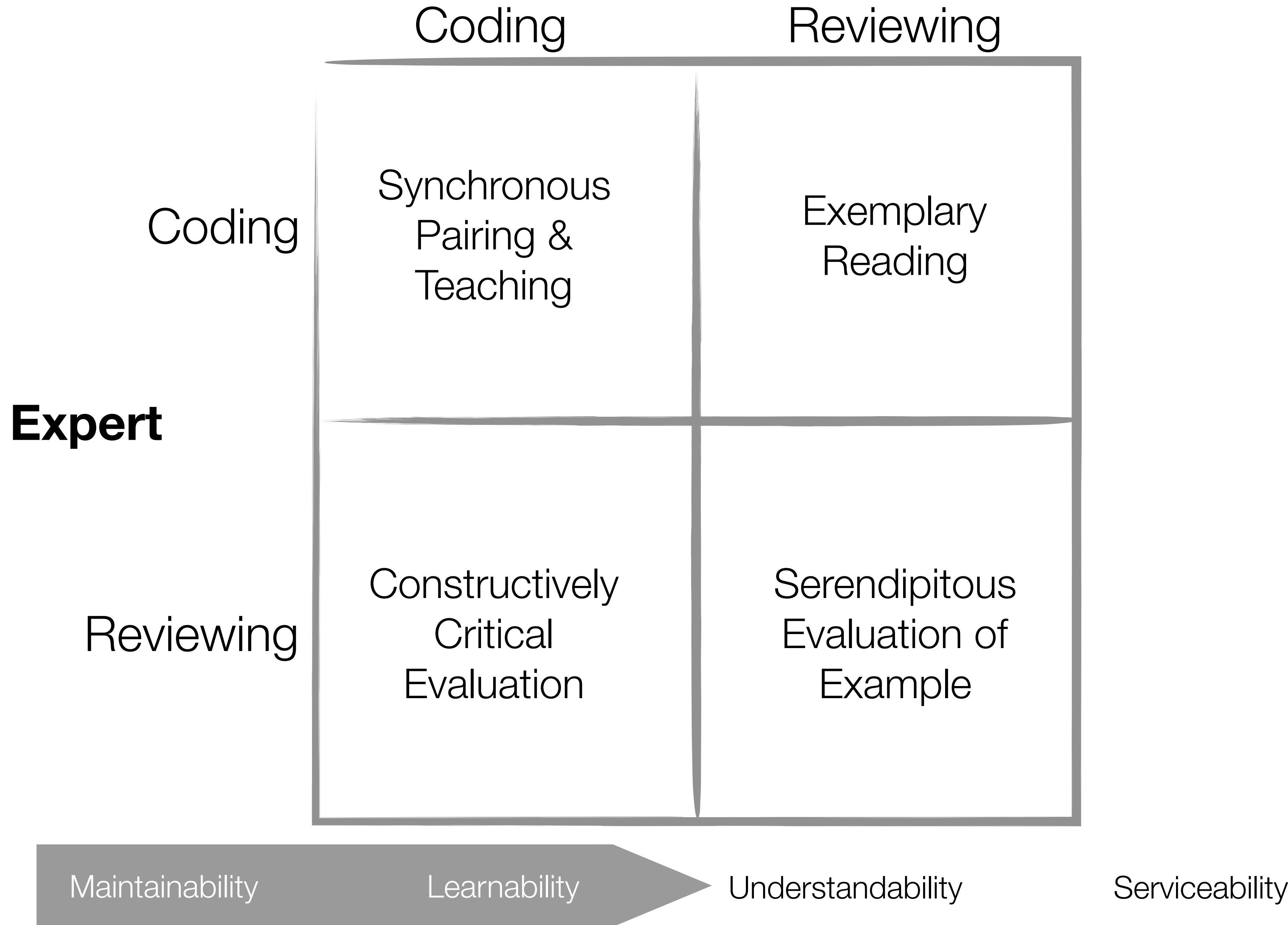
- Learnability
- Understandability
- Serviceability

Code review drives
Learnability

- Developing Code
 - Patterns & Conventions
 - Risks & Goals
- Developing People
 - Common Vocabulary
 - Teaching Moments



Learner



Code review drives
Understandability

- Establishes common yet evolving mental model
- Builds confidence in direction and design decisions
- Builds tribal knowledge
- Bonus: Enables elevator pitch

Maintainability

Learnability

Understandability

Serviceability

Code review drives
Serviceability

- Exposes addressable “gotchas”
- Exposes end-user interaction points
- Establishes consensus on supported workflows

Maintainability

Learnability

Understandability

Serviceability

“Given enough eyes, all bugs are shallow.”

Linus’s Law

Code review drives
Maintainability

✓ Learnability

✓ Understandability

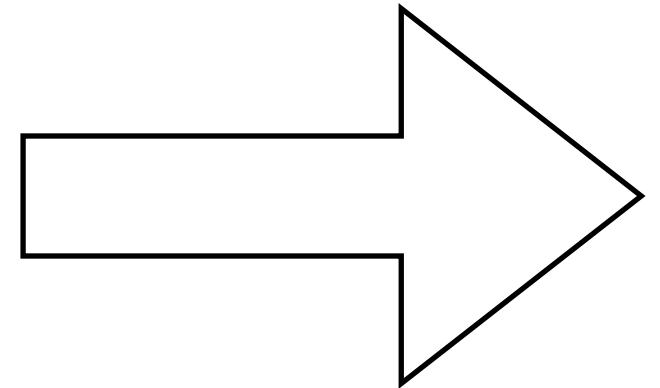
✓ Serviceability

First programming job out of school - B2B imprinting company

```
if ($customer == "spacely_sprockets") { do_something(); }
else { cry(); }
```

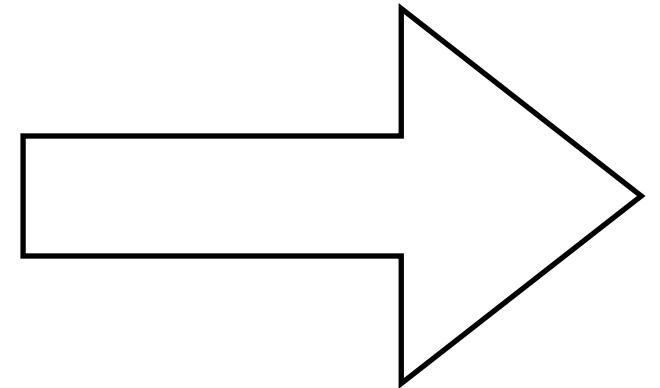
- Version control!
- **No code review** tooling or process
- Minimal pairing
- Continuous integration easily circumvented

Lack of code review



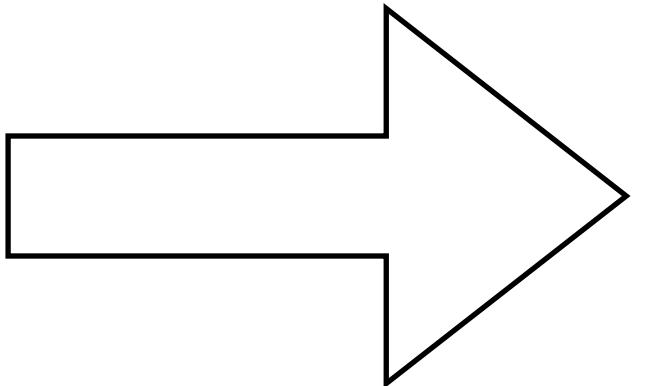
Lost Opportunities

Lack of code review



Lost Opportunities
Lost Revenue

Lack of code review



Lost Opportunities
Lost Revenue
Lost Job

Code review ensures
Compliance

Code review drives
Compliance

- Accessibility
- Auditability
- Idiomaticity

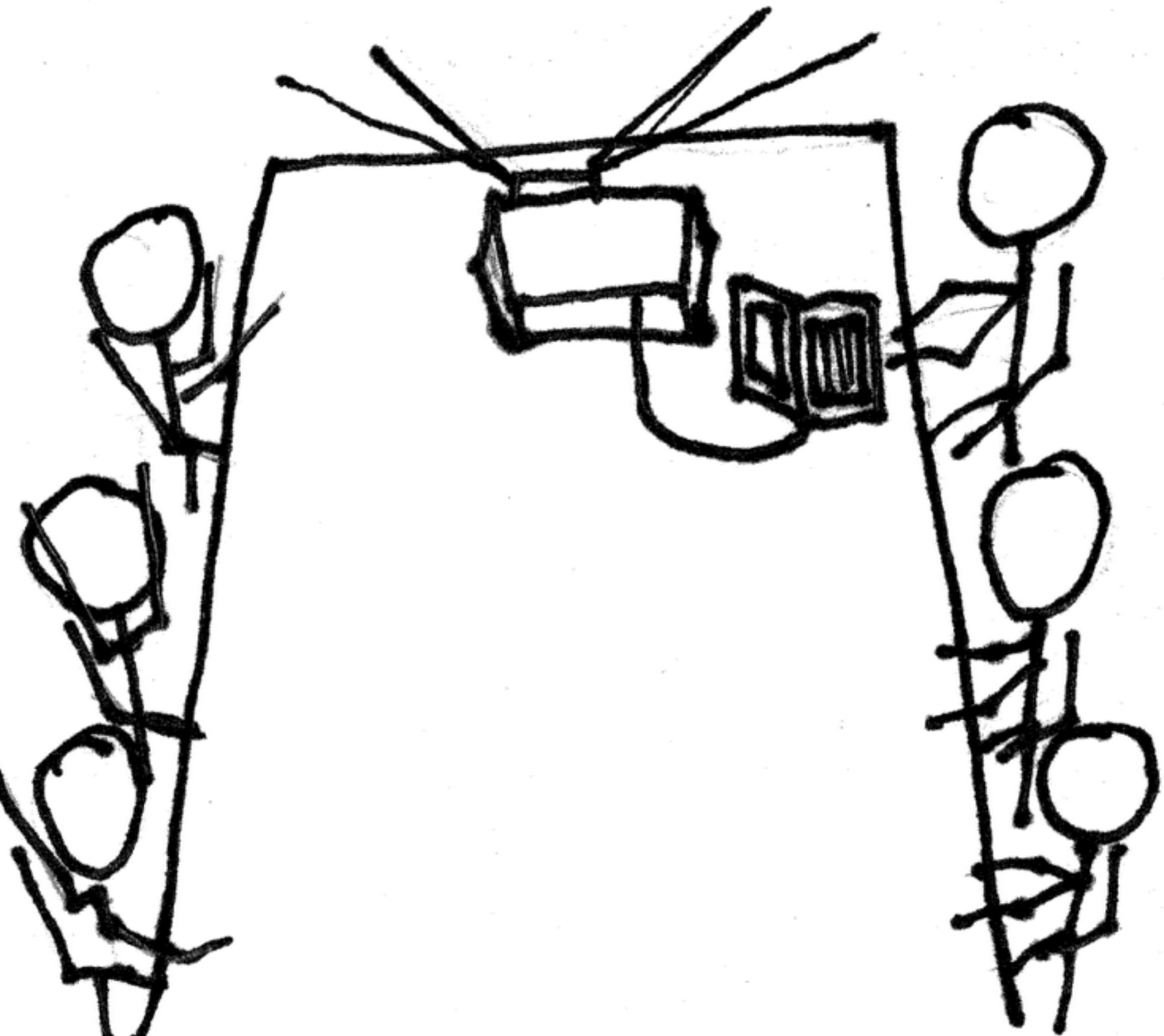
Second job out of school - Consulting

- Lone wolf working alongside other lone wolves
- No version control in proprietary software with custom “IDE” a.k.a. textarea.
- Last modified and modifier only
- No process of our own

First professional code review experience was group review

- Subcontractor on government project, 2010-2012
- Lone SME on platform
- Borland StarTeam + in house review system
 - My tools for version control integration
- Weekly merge window
- Round robin inspection

```
87 if(process.sucks()) {  
88 -   devs.foreach(suffer)  
+   process.map(devs,change)  
89 } else {  
90 -   continue  
+   new Retrospective(devs,process)  
91 }
```





Not a pleasant experience

- **Three to four hour** weekly round robin inspection
- Cutthroat mixture of competing contractors, subcontractors, and employees
- Embarrassment galore  Not a learning environment
- Immediate defensive posture
- “Merge next week” = you failed, possibly delayed project

\$1,450 per hour

\$1,450 per hour

\$5,800 per weekly meeting

\$1,450 per hour

\$5,800 per weekly meeting

\$290,000 per year

Effects?

- Waste
- “Get this over with.”
- Obstructionism
- Plenty of bugs
 - “I’ll fix that mistake later.”

Missed opportunities

- Accessibility expert was most vocal
- Project manager was vocal on contractual and HF matters
 - Both could have reviewed **asynchronously**
- Project was behind
 - Too many people could say No

Code review ensures
Security

Code review drives

Security

- Spot vulnerabilities
- Teach best practices
- Filter unnecessary code
 - YAGNI

Reviewers are like your lawyer

Screening and recommending actions to minimize risk, avoid preventable mistakes

Agenda

- Quick anecdote
- What is code review?
- What problems does code review solve?
- Quality attributes code review ensures
- **Tips for code reviews**
- Limitations

When should you integrate code review?

Context

- Project
- Technical

Keep reviews informal and short.

Tips for thorough code review

- Devote time
- Accept debt
- Identify churn
- Minimize pedantry
- **Make progress**

Major things we look for

- Algorithmic complexity
- Exception & error handling
- Exception, class, & variable naming
- Logging sufficiency & level
- Style conformation (automate!)
- Long lines & methods
- Readability
- Single purpose per commit

Most importantly

Does it work?

Is it tested?

Agenda

- Quick anecdote
- What is code review?
- What problems does code review solve?
- Quality attributes code review ensures
- Tips for code reviews
- **Limitations**

Code review cannot
Analyze dynamic structures

Code review cannot
Go on endlessly

Code review cannot
Solve political problems

Agenda

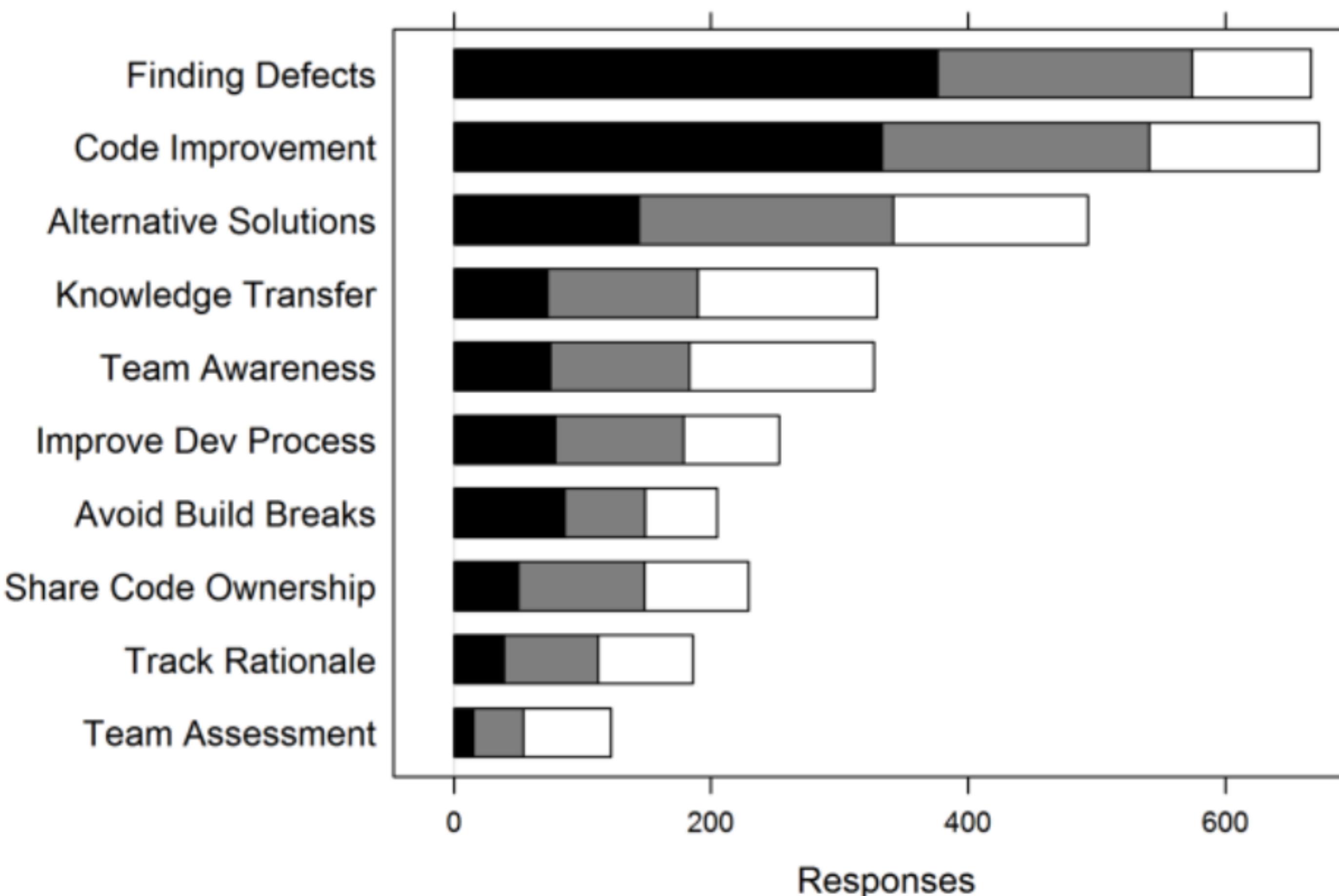
- Quick anecdote
- What is code review?
- What problems do code review solve?
- Quality attributes code review ensures
- Tips for code reviews
- Limitations



YOU DON'T HAVE TO TAKE MY WORD FOR IT

Ranked Motivations From Developers

Top Second Third



Code Review

is systemic examination of proposed changes to a codebase.

solves mental model synchronization and **tribal knowledge development**.

ensures maintainability, compliance, & security.

must be short, thorough, and automated where possible.

will not solve all human problems, but *some* is better than *none*.

— Pittsburgh —

CODE & SUPPLY Co.

—

www.codeandsupply.co
@codeandsupply



@ColinDean

[github.com/
colin dean/talks](https://github.com/colin dean/talks)

speakerdeck.com/colin dean



FIN

Attributions

- Westminster College picture: <https://www.flickr.com/photos/westminstercollege/15759678054/in/album-72157649340620016/>
- RMU picture: <http://cfbarchitects.com/higher-education/selected-projects/academic-buildings-libraries-learning-commons/robert-morris-university/>
- Pittsburgh picture: probably Dave DiCello
- “Their first code review” <http://classicprogrammerpaintings.tumblr.com/post/142702963264/their-first-code-review-william-frederick>
- Bass, Len; Paul Clements, and Rick Kazman. *Software Architecture in Practice*. Addison Wesley, 2013.
- Wiegers, Karl E. *Peer Reviews in Software*. Addison Wesley, 2012.
- Cohen, Jason, Steven Teleki, and Eric Brown. *Best Kept Secrets of Peer Code Review*. Smart Bear Software, 2006.
- Wilhelm, Alex and Alexia Tsotsis. Julie Ann Horvath Describes Sexism and Intimidation behind Her Github Exit. TechCrunch, 2014 March 15. Retrieved 2016 April 26. <http://techcrunch.com/2014/03/15/julie-ann-horvath-describes-sexism-and-intimidation-behind-her-github-exit/>
- Bacchelli, Alberto and Christian Bird. *Expectations, Outcomes, and Challenges Of Modern Code Review*. Proceedings of the International Conference on Software Engineering, Proceedings of the International Conference on Software Engineering, May 2013.
- and others mentioned in the slides

No, really. Fin. Srsly.

Third out of school and current job - Engineering

- Highly disciplined team using Java, Scala, and Groovy
- Git + Gerrit
- Constructively critical feedback
 - No criticism without alternative solution and reasoning
 - Wide experience range: 1-2 yrs to 25+ yrs
- Team split in late 2014, I was asked to be tech lead

Github Enterprise in 2016

- All new projects
- Same workflow as public Github

Code Review Tools

