CodeScene

PRIORITIZING TECHNICAL DEBT AS IF TIME AND MONEY MATTERS

by Adam Tornhill, June 2022

Lehman's "Laws" of Software Evolution

Continuing Change

"a system must be continually adapted or it becomes progressively less satisfactory"

Increasing Complexity

"as a system evolves, its complexity increases unless work is done to maintain or reduce it"

M. Lehman, "Programs, life cycles, and laws of software evolution", 1980

Increasing Complexity: consequences and impact

Symptoms the business sees

Symptoms that the users experience

Roadmap

Long Cycle Times: time to market

Lack of Predictability: when will it be done?

Organizational problems

(not well-understood in the industry...yet)

Low innovation:

no time, no opportunities to experiment

Team

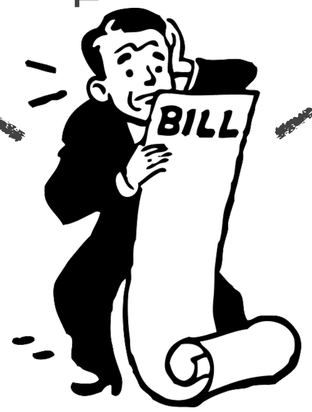
Developer attrition

Key Personnel dependencies

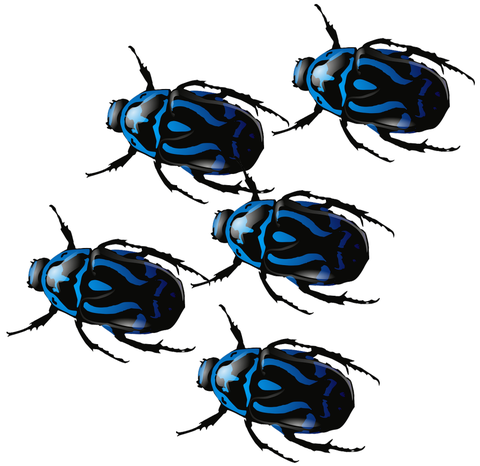
Technical Debt

Technical Debt

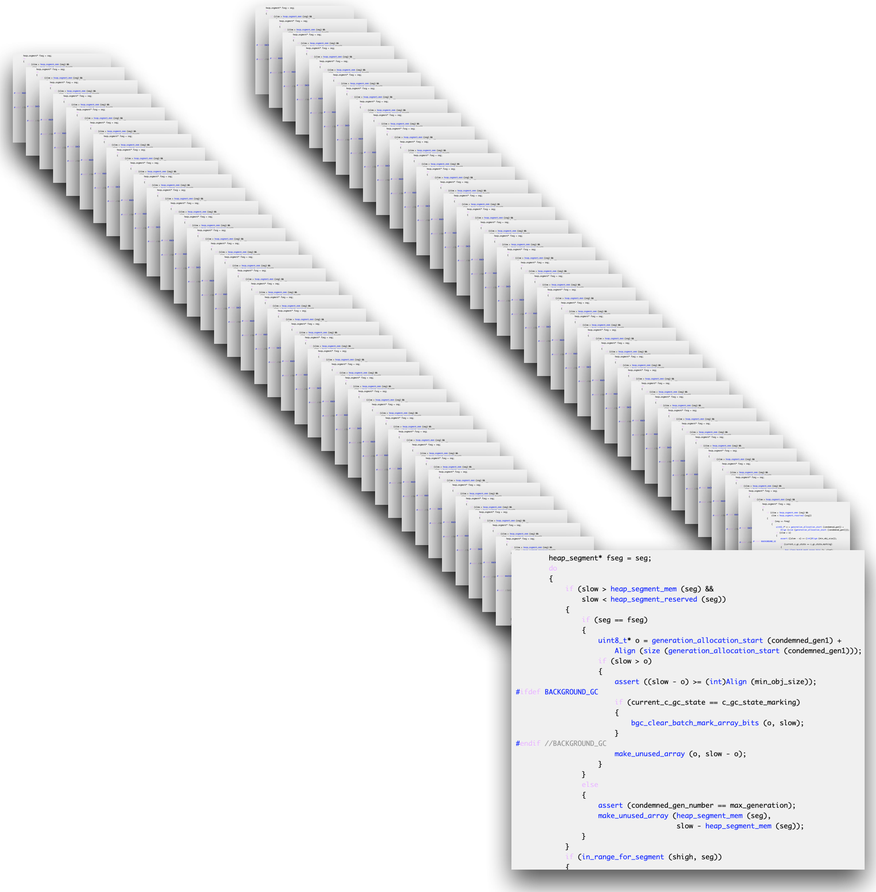
Technical Debt



Product



Quantifying technical debt via static analysis?

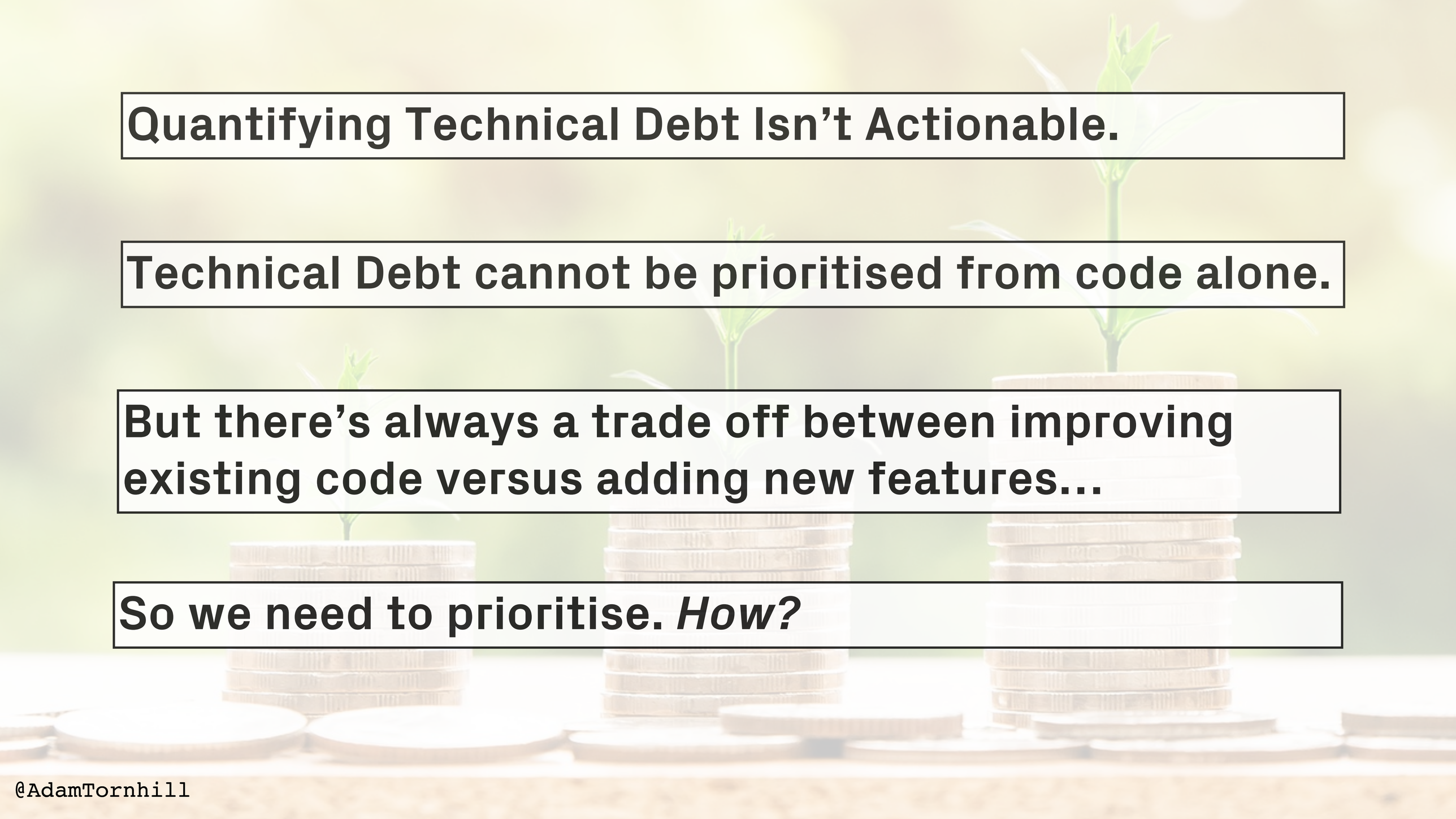


?

==

4000 Years of Technical Debt?

4000 Years Ago => The Start of Recorded History



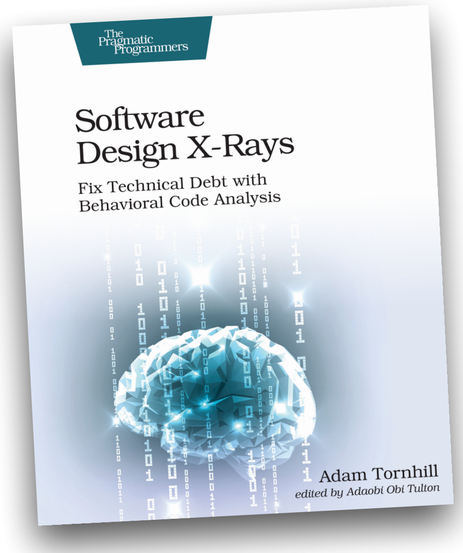
Quantifying Technical Debt Isn’t Actionable.

Technical Debt cannot be prioritised from code alone.

But there’s always a trade off between improving existing code versus adding new features...

So we need to prioritise. *How?*

What is a **Behavioral Code Analysis?**



Behavioral Code Analysis = code + people + context

While the code is important, it's even more important to understand how we — as a development organisation — interact with the system we're building.

Version-Control — **A Behavioral Data Source**

```
Commit: b557ca5
Date: 2016-02-12
Author: Kevin Flynn

    Fix behavior of StartsWithPrefix

    src/Mvc.Abstractions/ModelBinding/ModelStateDictionary.cs
    src/Mvc.Core/ControllerBase.cs
    src/Mvc.Core/Internal/ElementalValueProvider.cs
1   39   src/Mvc.Core/Internal/PrefixContainer.cs

Commit: fd6d28d
Date 2016-02-10
Author: Professor Falken

   Make AddController not overwrite existing IControllerTypeProvider

8   1    src/Core/Internal/ControllersAsServices.cs
48  0    test/Core.Test/Internal/ControllerAsServicesTest.cs
13  0    test/Mvc.FunctionalTests/ControllerFromServicesTests.cs

Commit: 910f013
Date :2016-02-05
Author Lisbeth Salander

   Fixes #4050: Throw an exception when media types are empty.

20  1    src/Mvc.Core/Formatters/InputFormatter.cs
```

Co-changing Files

Social Information

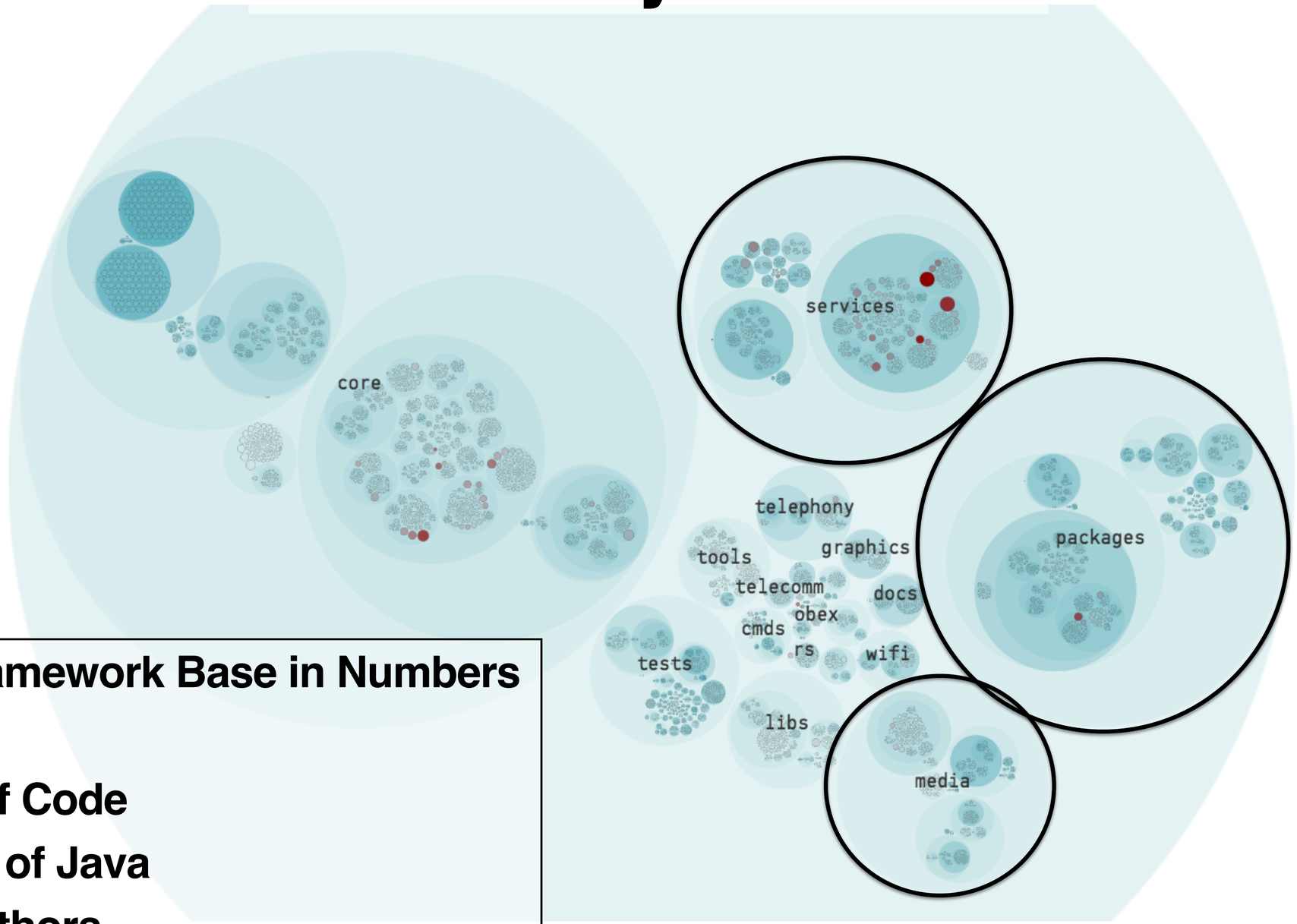
Progress on Tasks

A Time Dimension

CASE STUDY:

PRIORITIZING TECHNICAL DEBT

Case Study: Android



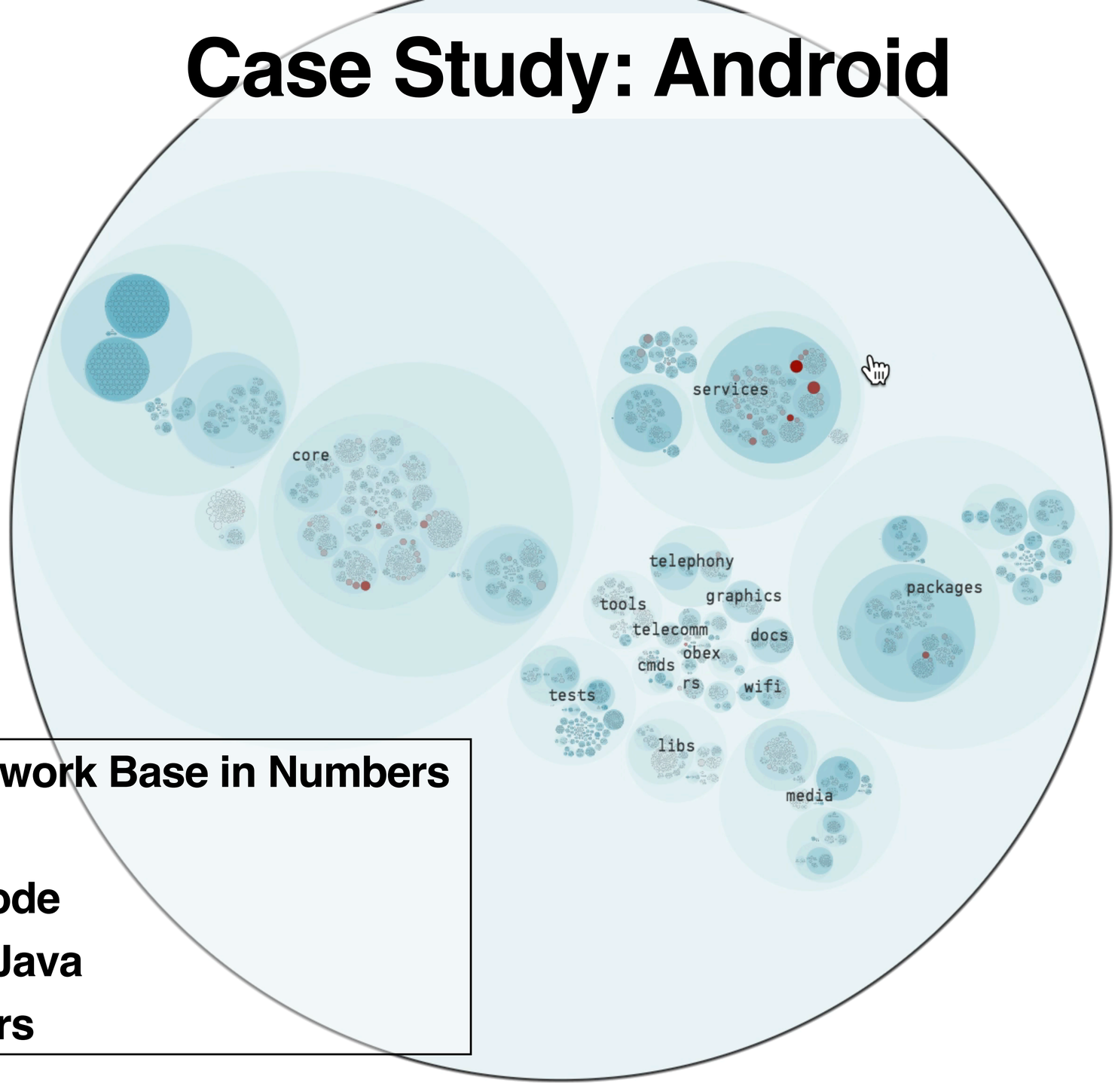
The Platform Framework Base in Numbers

3 Million Lines of Code

2,1 Million Lines of Java

2,000 Unique Authors

Case Study: Android



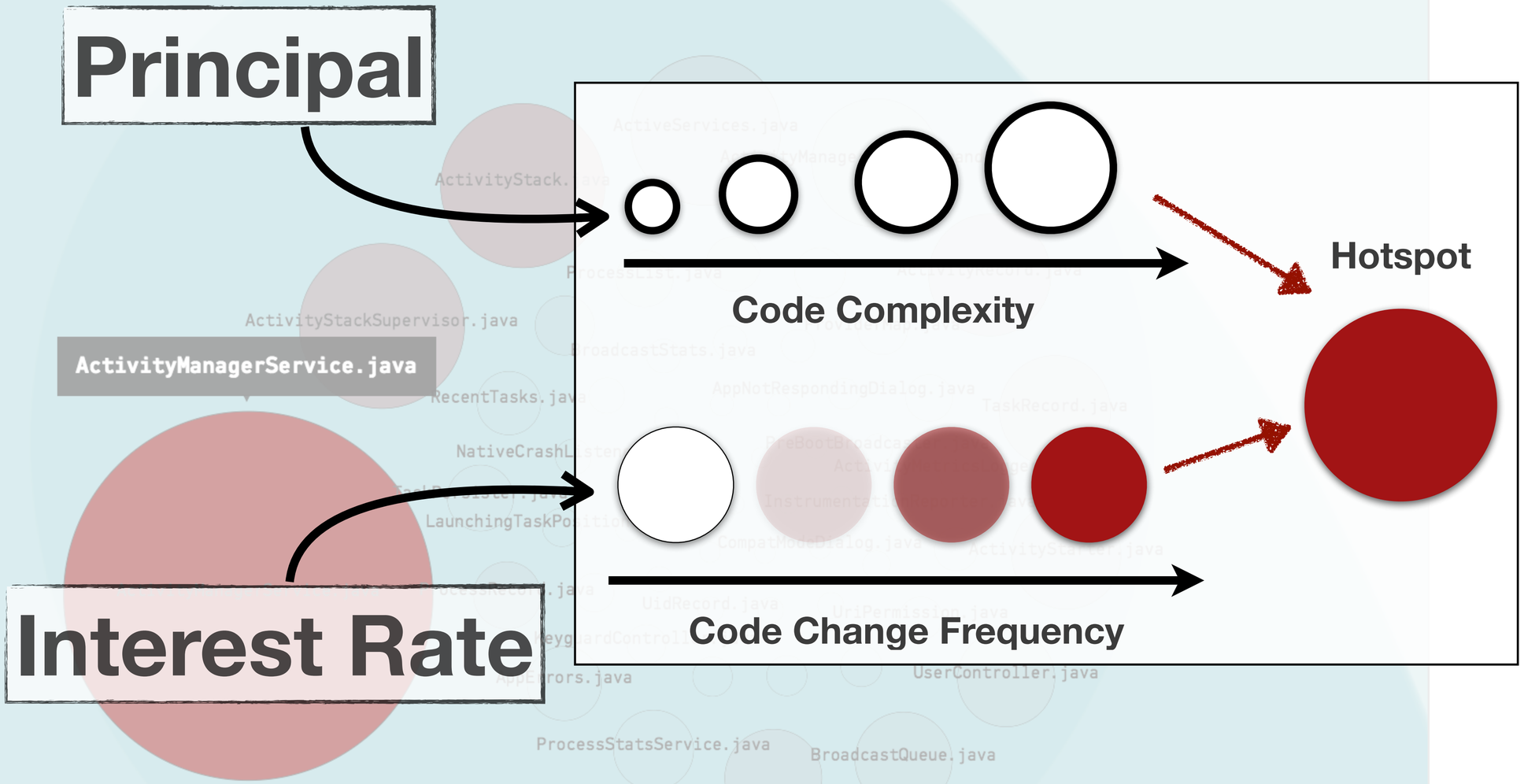
The Platform Framework Base in Numbers

3 Million Lines of Code

2,1 Million Lines of Java

2,000 Unique Authors

@AdamTornhill



@AdamTornhill

CAN WE MEASURE **“CODE COMPLEXITY”**?

Complexity Explained: is “code quality” really a thing?

“The assumption that fundamentally different views of complexity can be characterised by a single number is counter to the fundamental concepts of measurement theory.”

[..]

“the most promising approach is to identify specific attributes of complexity and measure these separately.”

Software Measurement: A Necessary Scientific Basis, N. Fenton (1994)

@AdamTornhill

Healthy Code: Beyond a Single Metric

Measure complexity via code health:

- Sample properties of the code that are known to correlate with increased maintenance costs and with higher risks of defects.
- Aggregate the metrics, normalize, and visualize.

Examples on Code Health Issues

Module Level:

Low Cohesion, many responsibilities

Brain Class, low cohesion, large class, at least one Brain Method

Function Level:

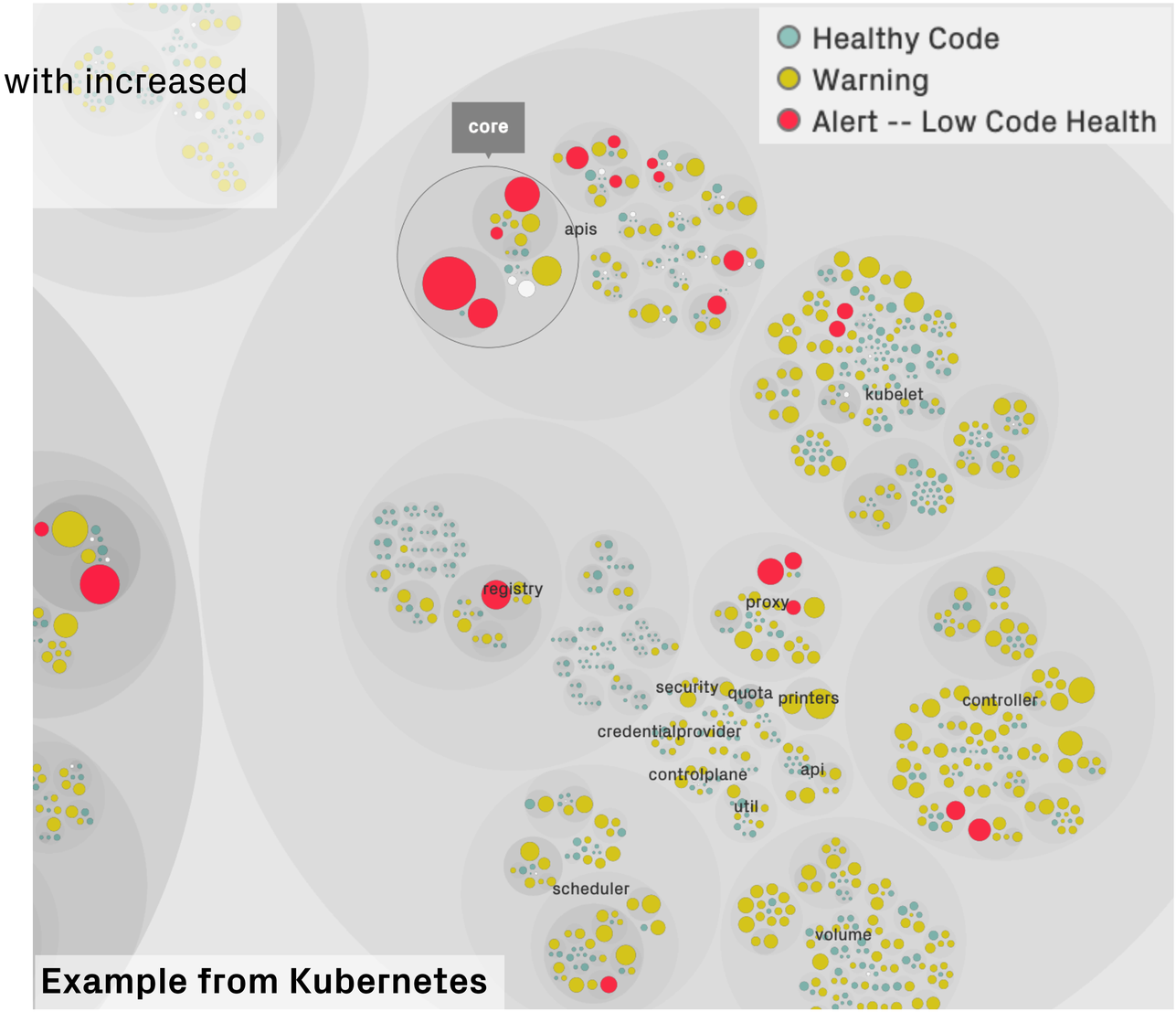
Brain Methods, complex functions that centralize the behavior of the module

Copy-pasted logic, missing abstractions, DRY violations

Implementation Level:

Deeply Nested Logic, if-statements inside if-statements

Primitive Obsession, missing a domain language



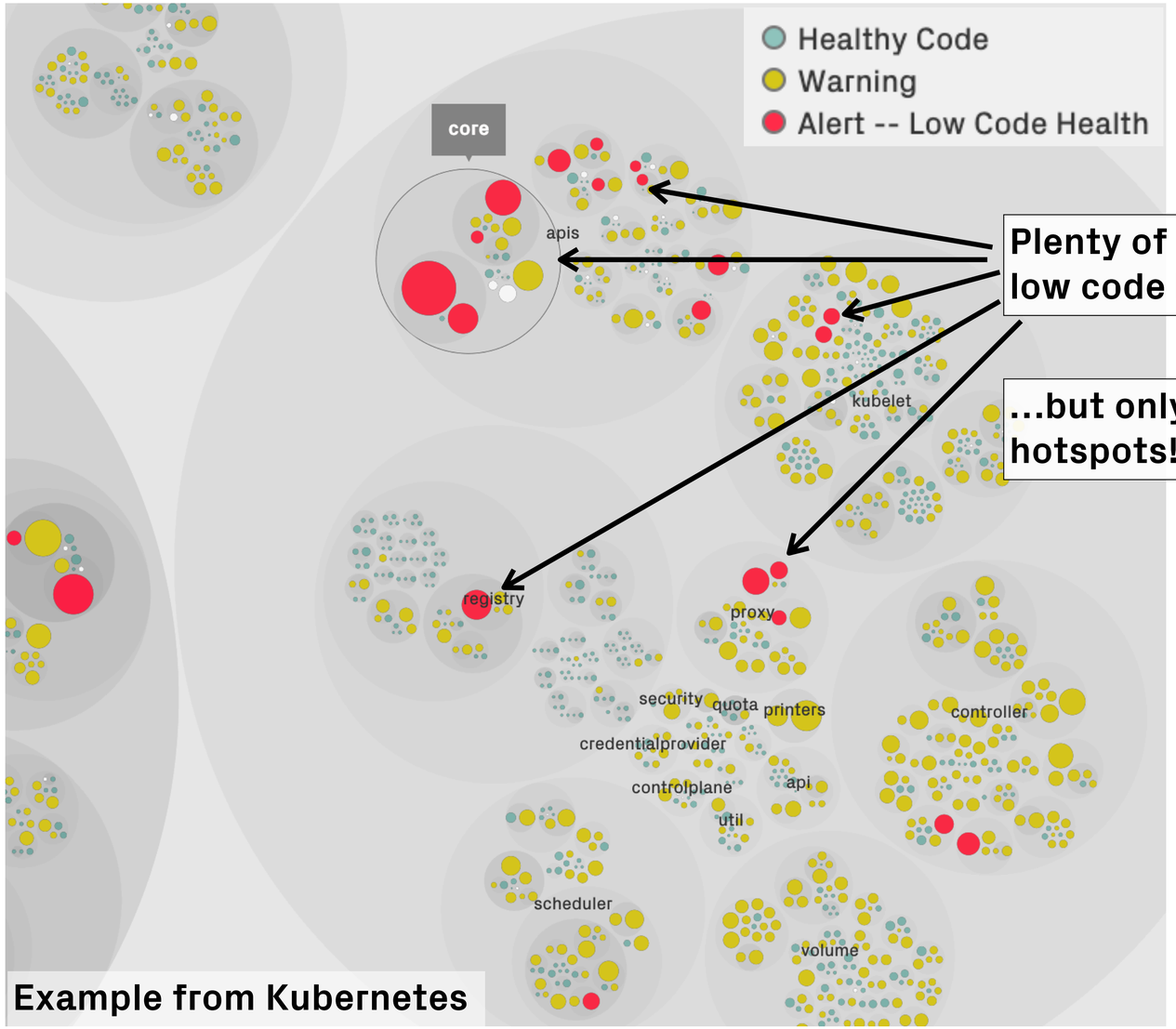
Example from Kubernetes

Learn More: https://codescene.com/blog/measure-code-health-of-your-codebase/

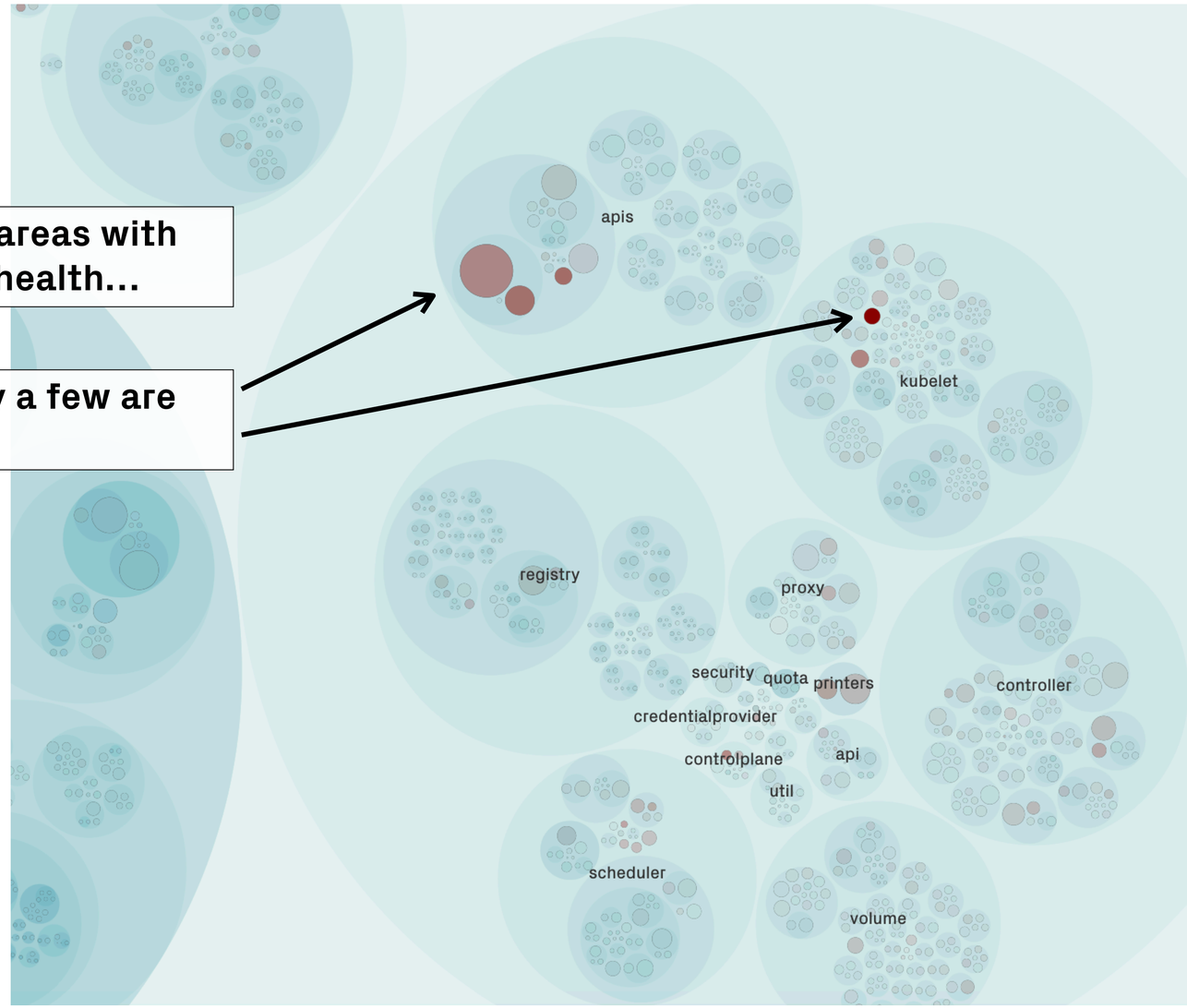
Code health: context matters

Quality: code health

Relevance: Development Hotspots

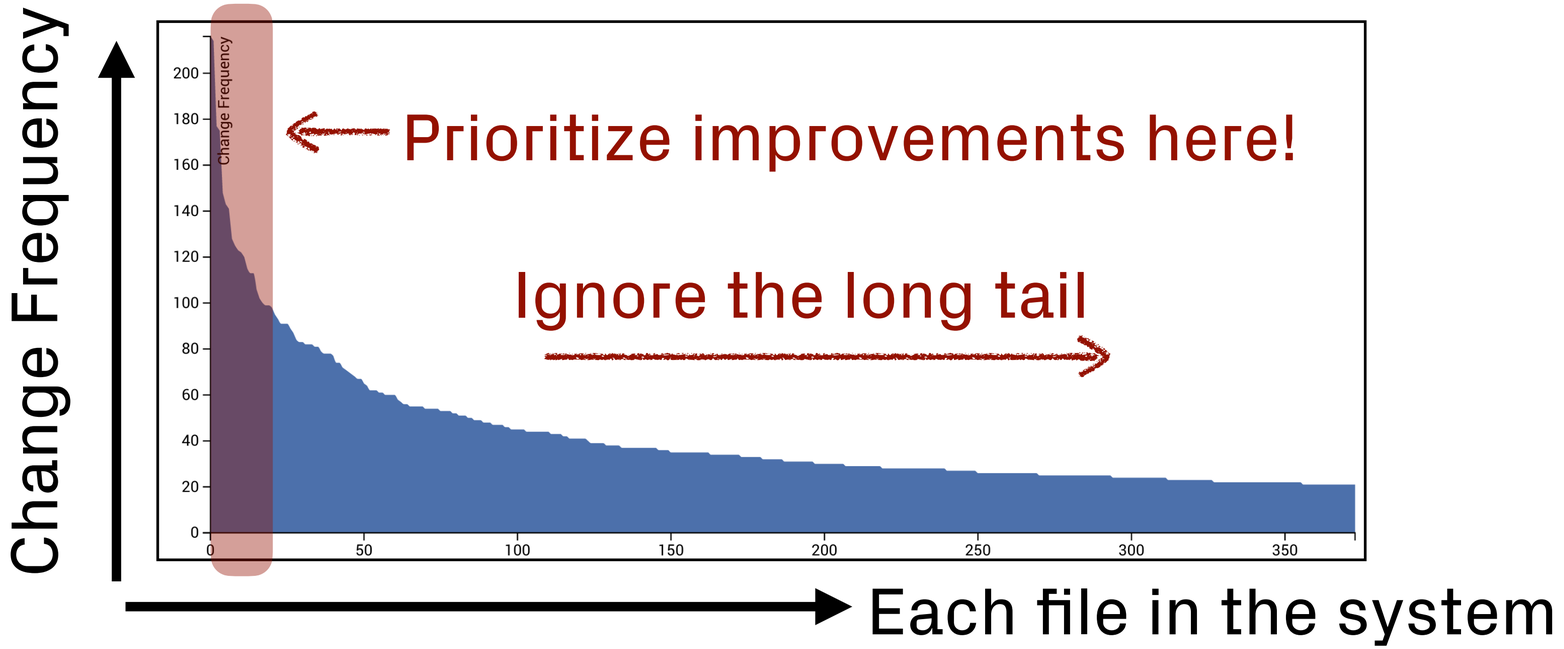


Example from Kubernetes



Finding **bad code** is the **easy** part.

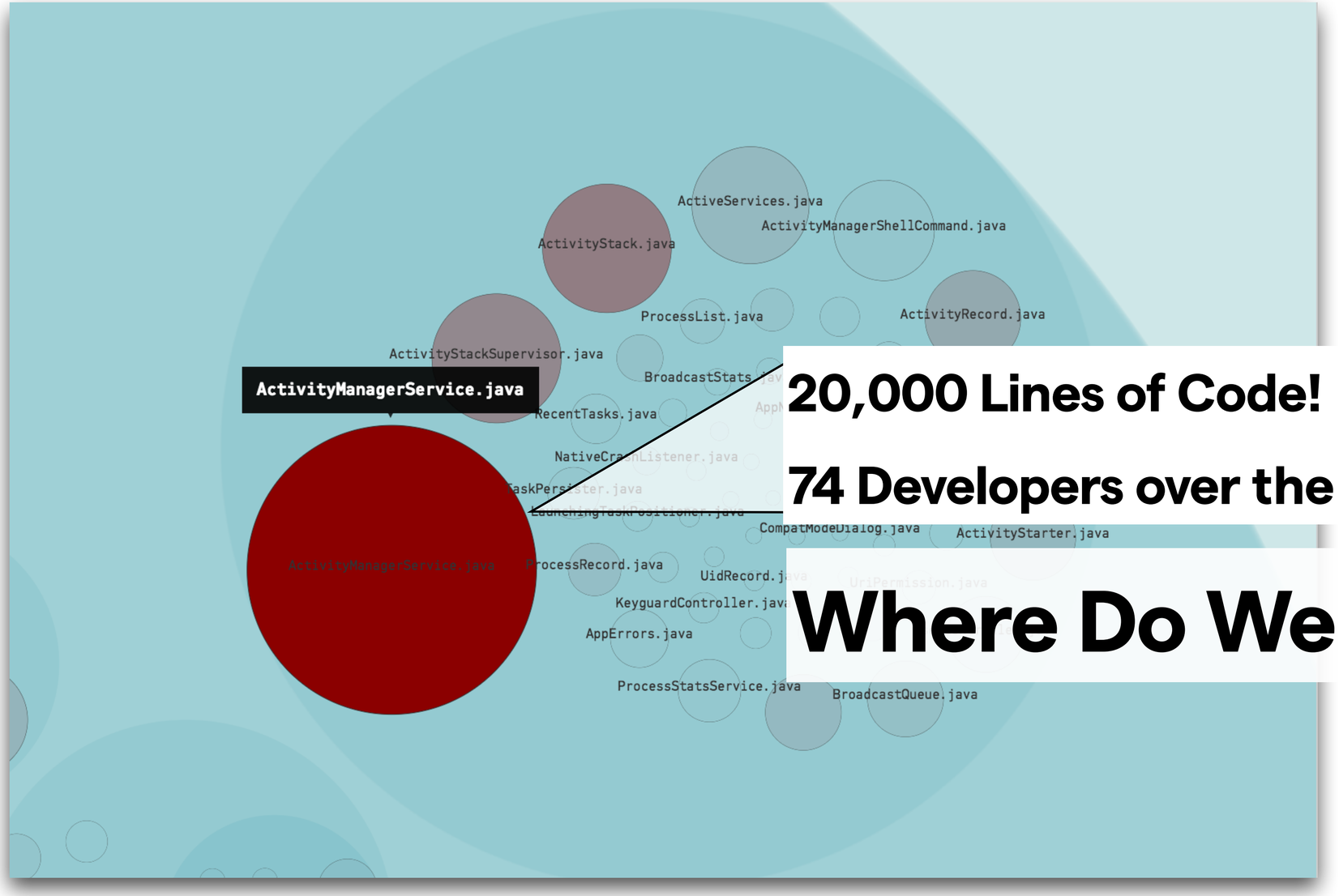
Hotspots: why you don’t have to fix all tech debt



Key take-aways:

- Most code is in the long-tail. This is low-interest debt.
- Hotspots only make up 2-4% of the total codebase, but attract 20-70% of all development activity!
- Code quality issues in a hotspot are expensive. This is high-interest debt.

Actionable Insights?

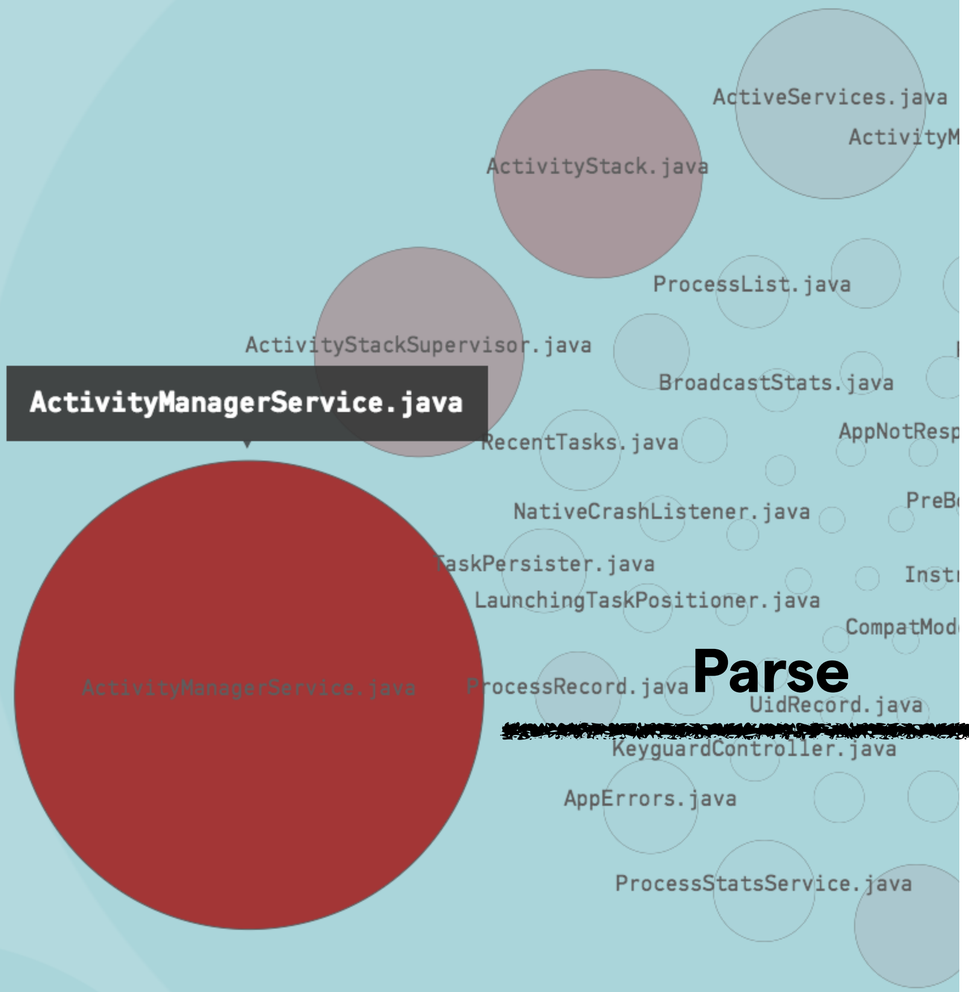


20,000 Lines of Code!

74 Developers over the past 3 Months

Where Do We Start?

Hotspots: X-Ray `ActivityManagerService.java`

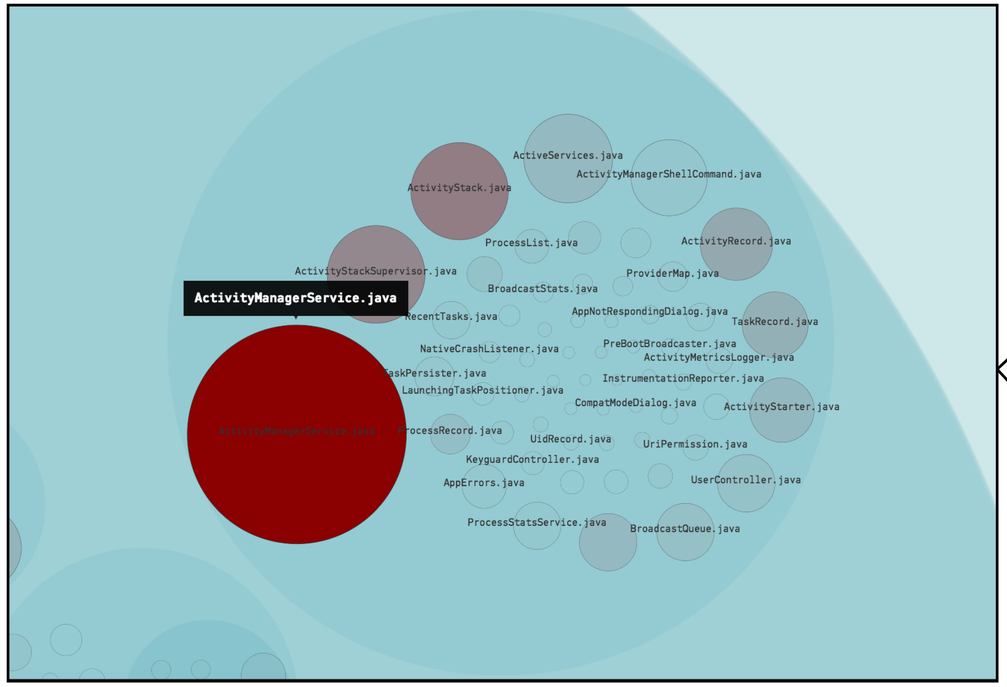


Function Level Hotspots

Recommended functions to improve.

From https://pragprog.com/book/atevol/software-design-x-rays

X-Ray of `ActivityManagerService.java`

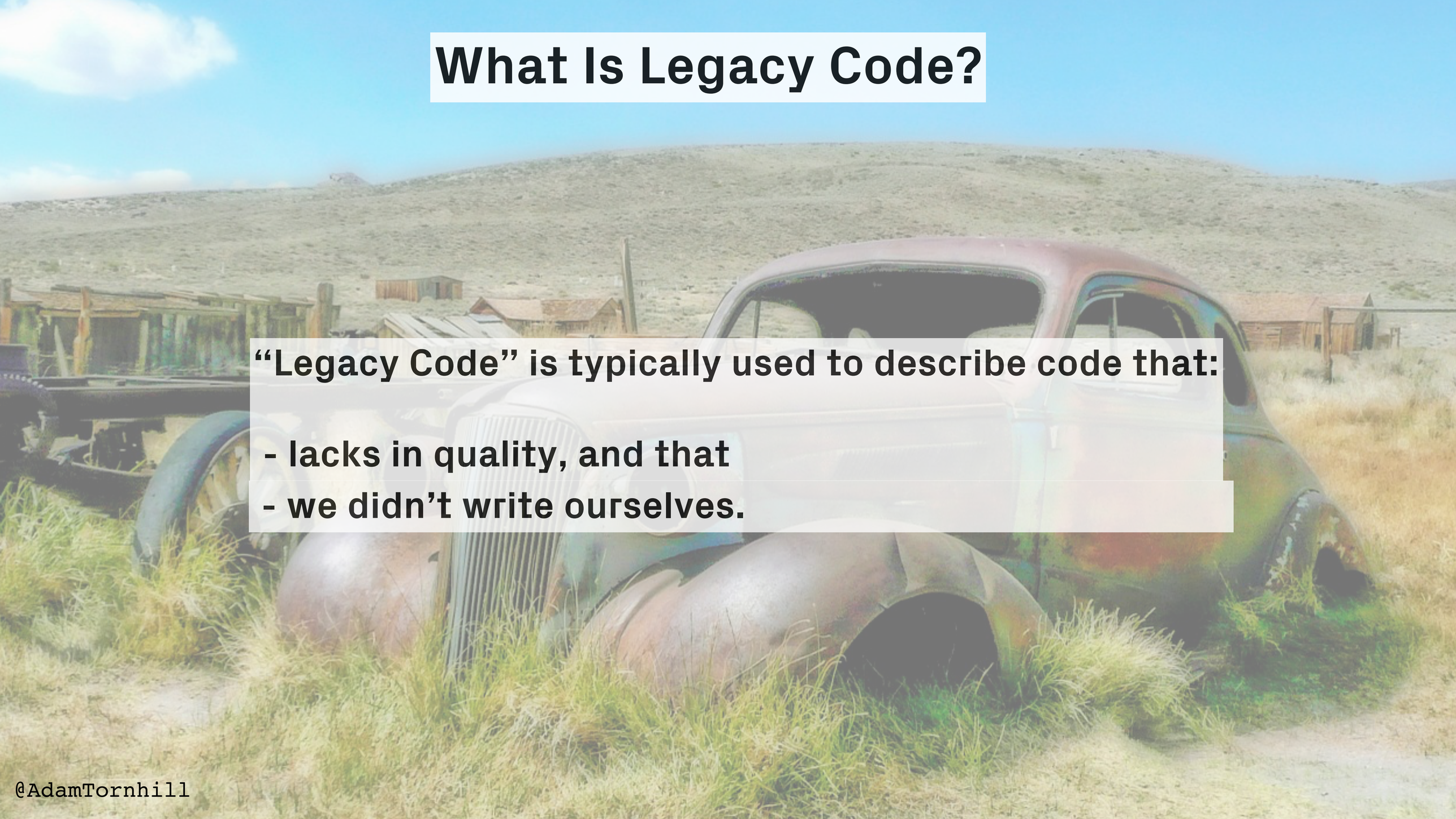


Function	Change Frequency	Lines of Code	Cyclomatic Complexity
`ActivityManagerService.MainHandler.handleMessage` View Complexity Trend | View Function Code	98	500	101
`ActivityManagerService` View Complexity Trend | View Function Code	75	160	13
`applyOomAdjLocked` View Complexity Trend | View Function Code	73	256	72

CodeScene

Tech Debt and People

What Is Legacy Code?



“Legacy Code” is typically used to describe code that:

- lacks in quality, and that
- we didn’t write ourselves.

The Technical Debt That Wasn't

Product #1	Product #2	Product #3
		?

Case Study:

How quick can you turn your current codebase into legacy code?

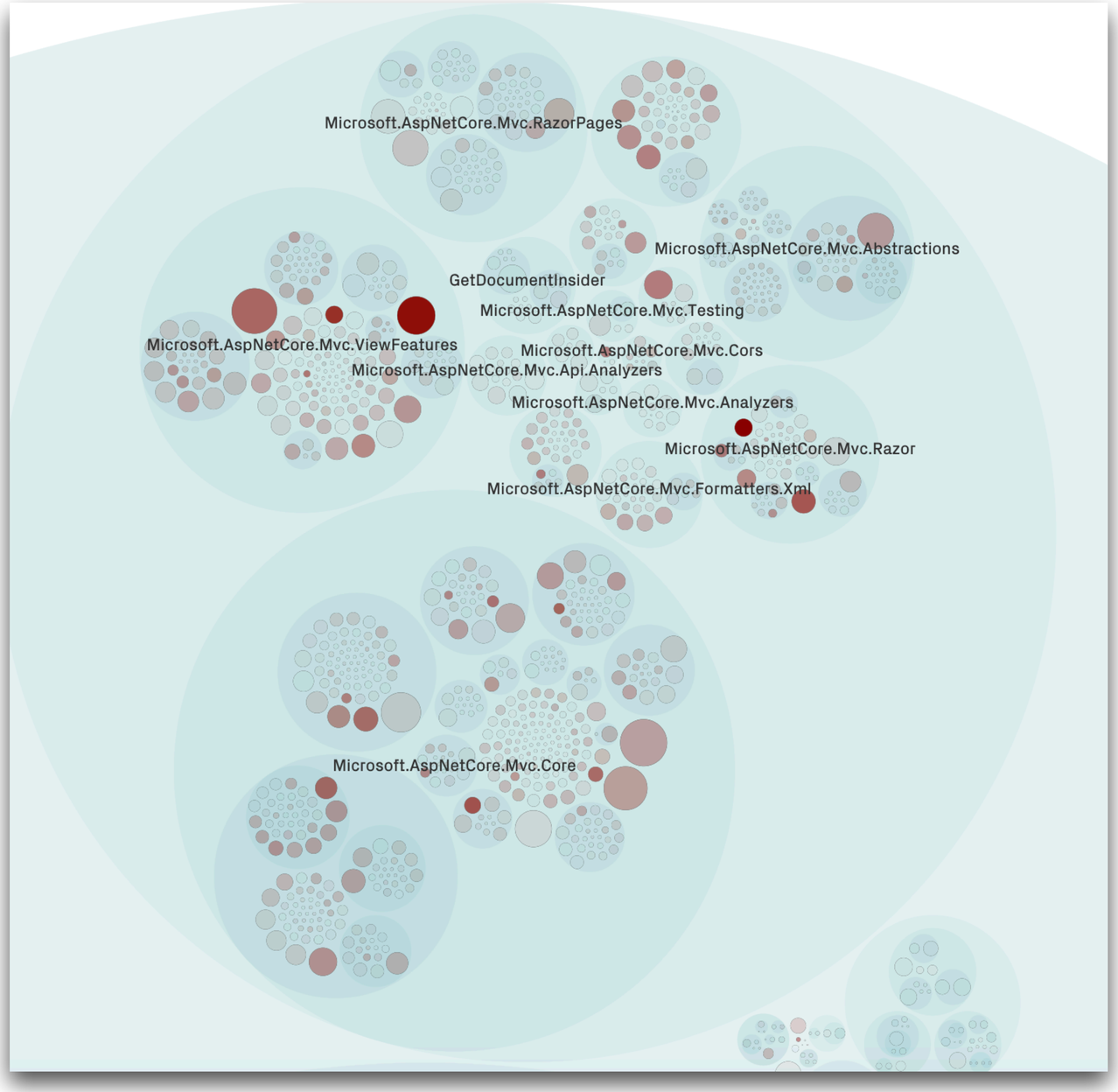
Case Study: ASP.NET MVC Core

ASP.NET MVC Core

~180 Contributors

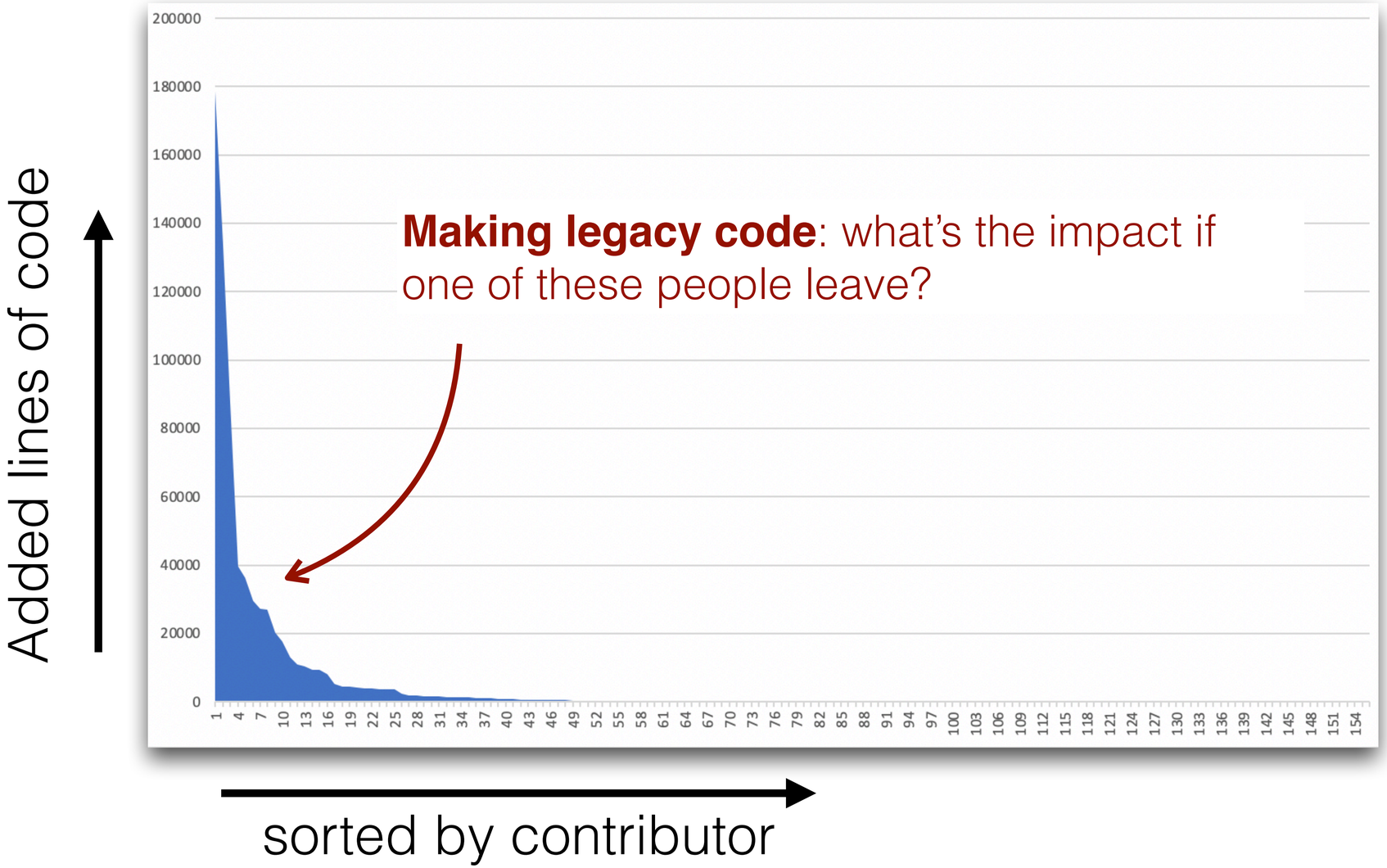
350,000 Lines of Code

Main language: C#



@AdamTornhill

Software Evolution: power laws are everywhere



Case Study: Off-Boarding

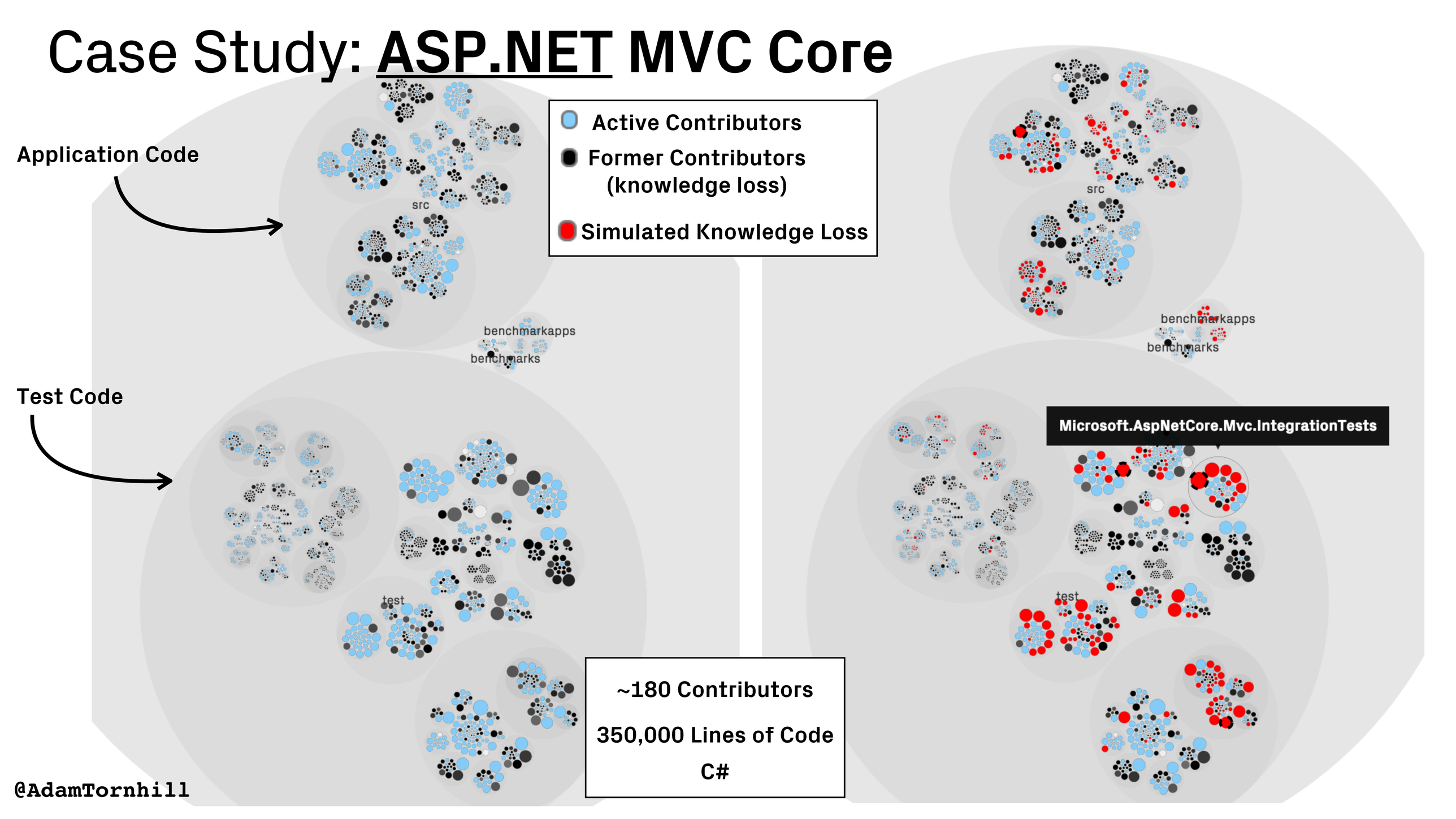
Identify the main developers behind each module

```
Commit: b557ca5
Date: 2016-02-12
Author: Kevin Flynn

    Fix behavior of StartsWithPrefix

     27   src/Mvc.Abstractions/ModelBinding/ModelStateDictionary.cs
     10   src/Mvc.Core/ControllerBase.cs
     1    src/Mvc.Core/Internal/ElementalValueProvider.cs
1    39   src/Mvc.Core/Internal/PrefixContainer.cs

Commit: fd6d28d
Date 2016-02-10
Author: Professor Falken

   Make AddController not overwrite existing IControllerTypeProvider

8    1    src/Core/Internal/ControllersAsServices.cs
48   0    test/Core.Test/Internal/ControllerAsServicesTest.cs
13   0    test/Mvc.FunctionalTests/ControllerFromServicesTests.cs

Commit: 910f013
Date :2016-02-05
Author Lisbeth Salander

   Fixes #4050: Throw an exception when media types are empty.

20   1    src/Mvc.Core/Formatters/InputFormatter.cs
```

Case Study: ASP.NET MVC Core

Application Code

Test Code



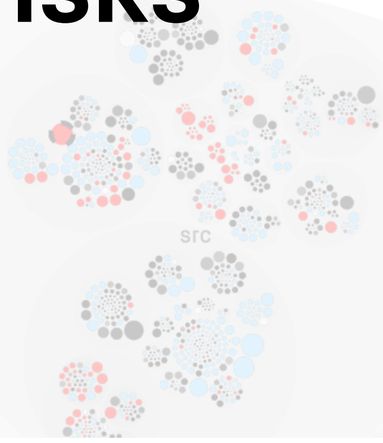
~180 Contributors

350,000 Lines of Code

C#

@AdamTornhill

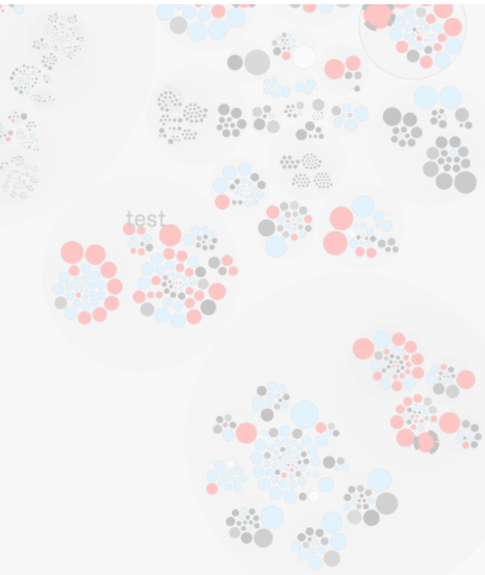
Mitigate off-boarding risks

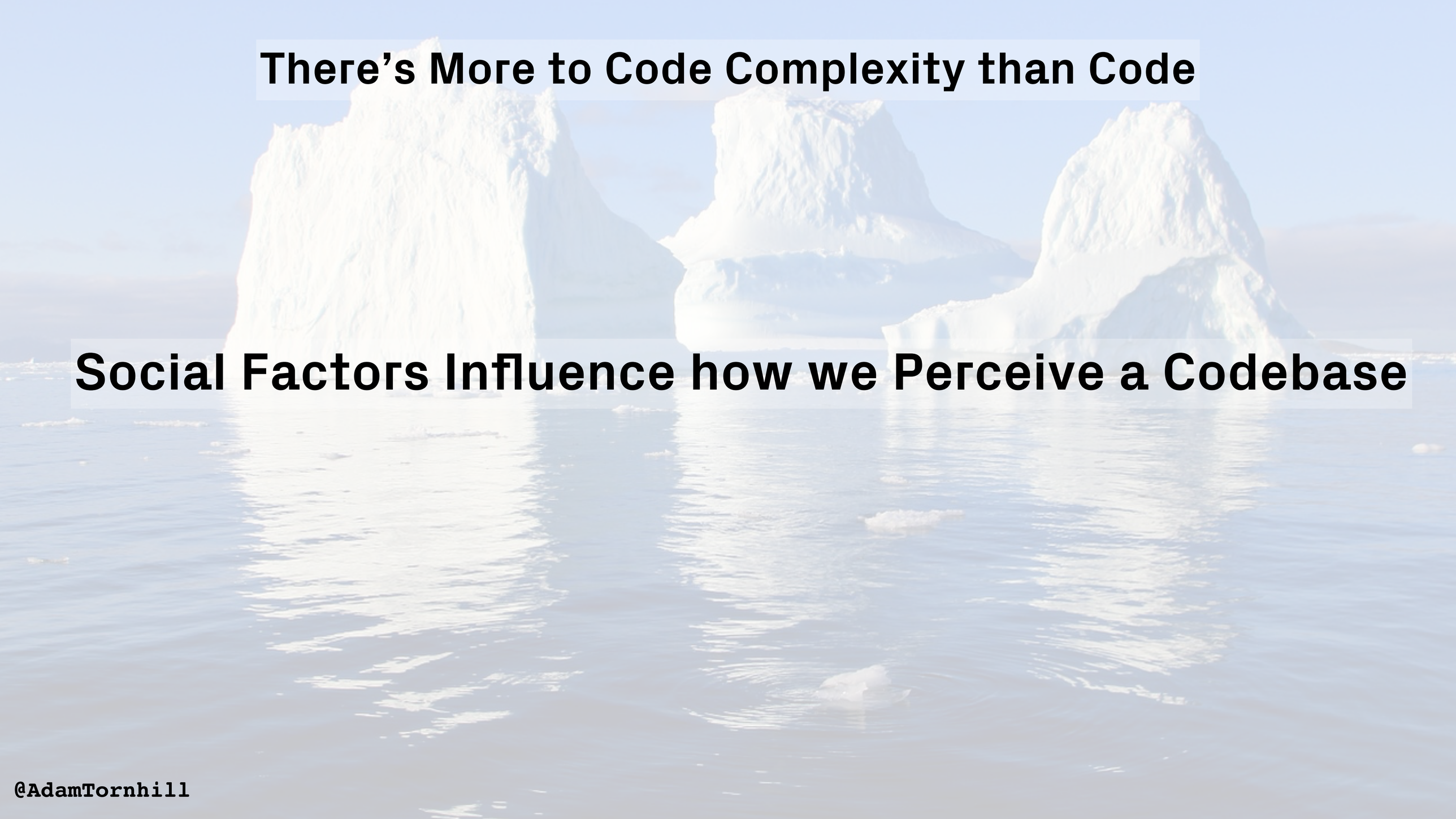


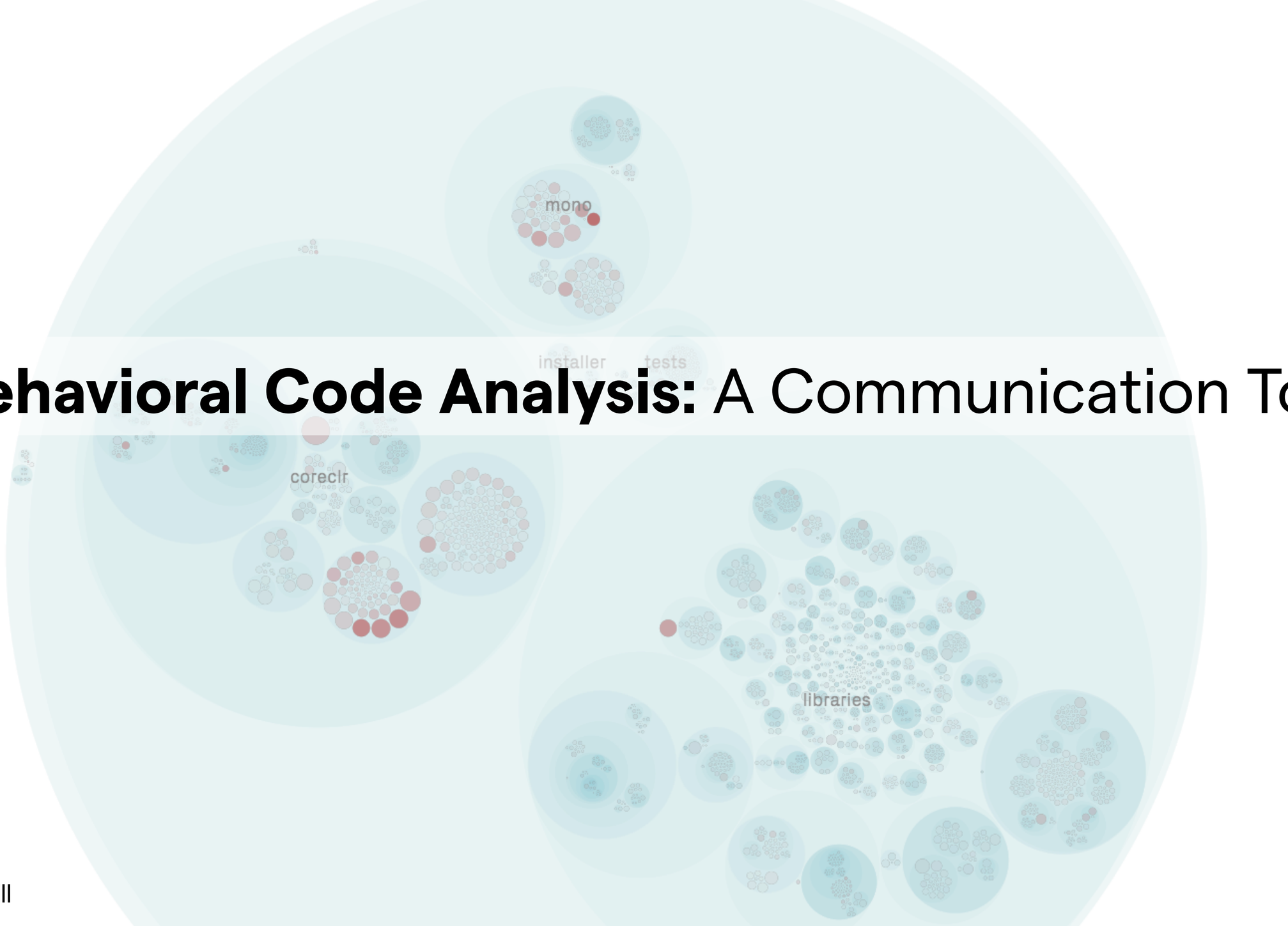
Identify risks by combining three properties:

```
knowledge loss + relevance (hotspot) + impact (complexity).
```

Limit the data to what's actionable.



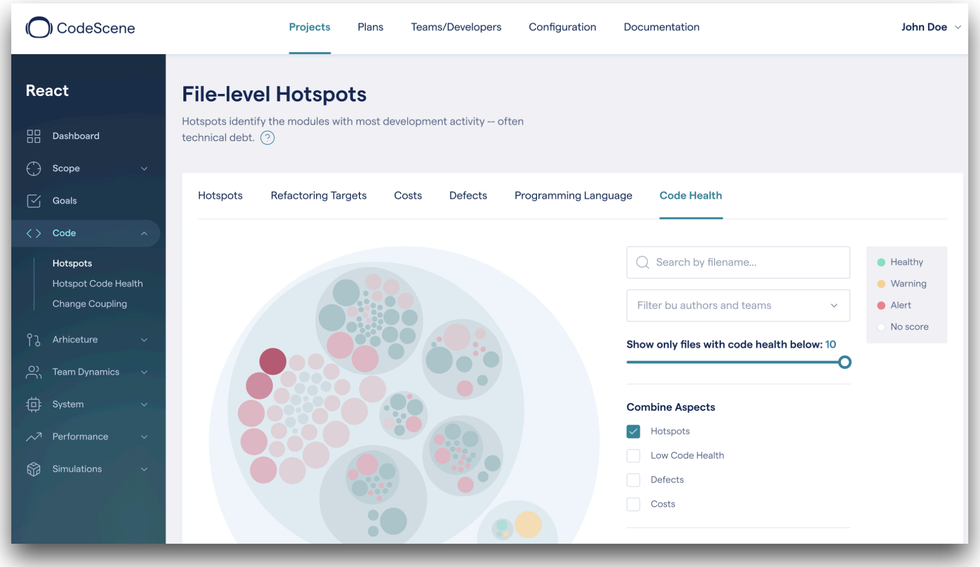

There’s More to Code Complexity than Code
Social Factors Influence how we Perceive a Codebase
@AdamTornhill



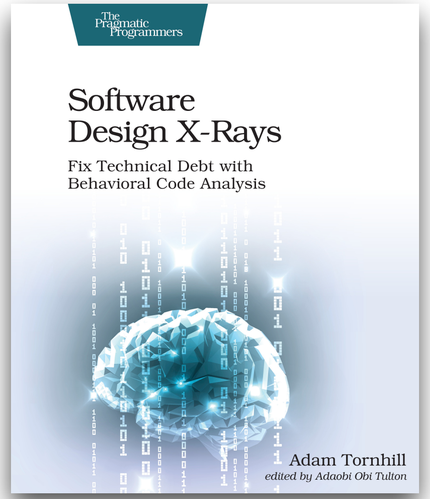
Behavioral Code Analysis: A Communication Tool

@AdamTornhill

Tools + examples: https://codescene.com/

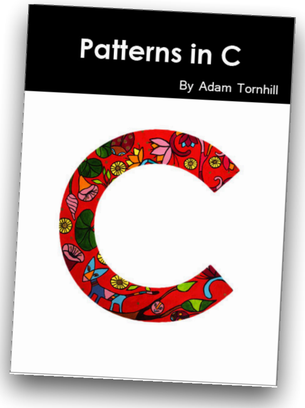


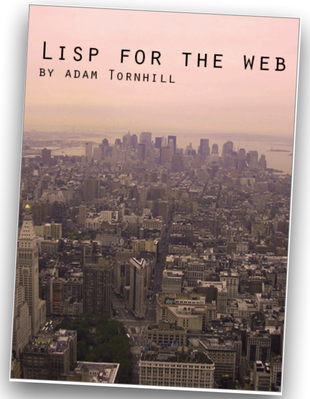
behavioral code analysis techniques, tech debt, teams, microservice analyses

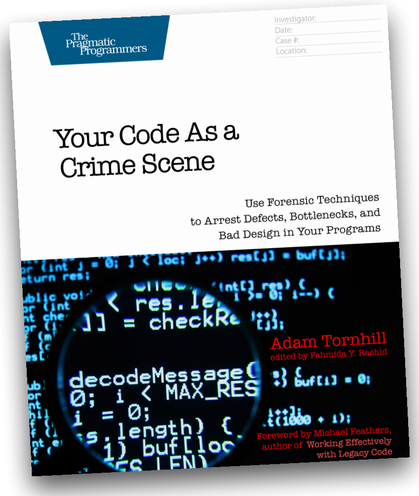


Blogs on Software Evolution, Technical Debt, and Code

- https://www.codescene.com/blog/
- https://adamtornhill.com/







Adam Tornhill

https://twitter.com/AdamTornhill

https://se.linkedin.com/company/codescene