

PRIORITIZING TECHNICAL DEBT AS IF TIME AND MONEY MATTERED

CodeScene™

Powered by Empear

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”

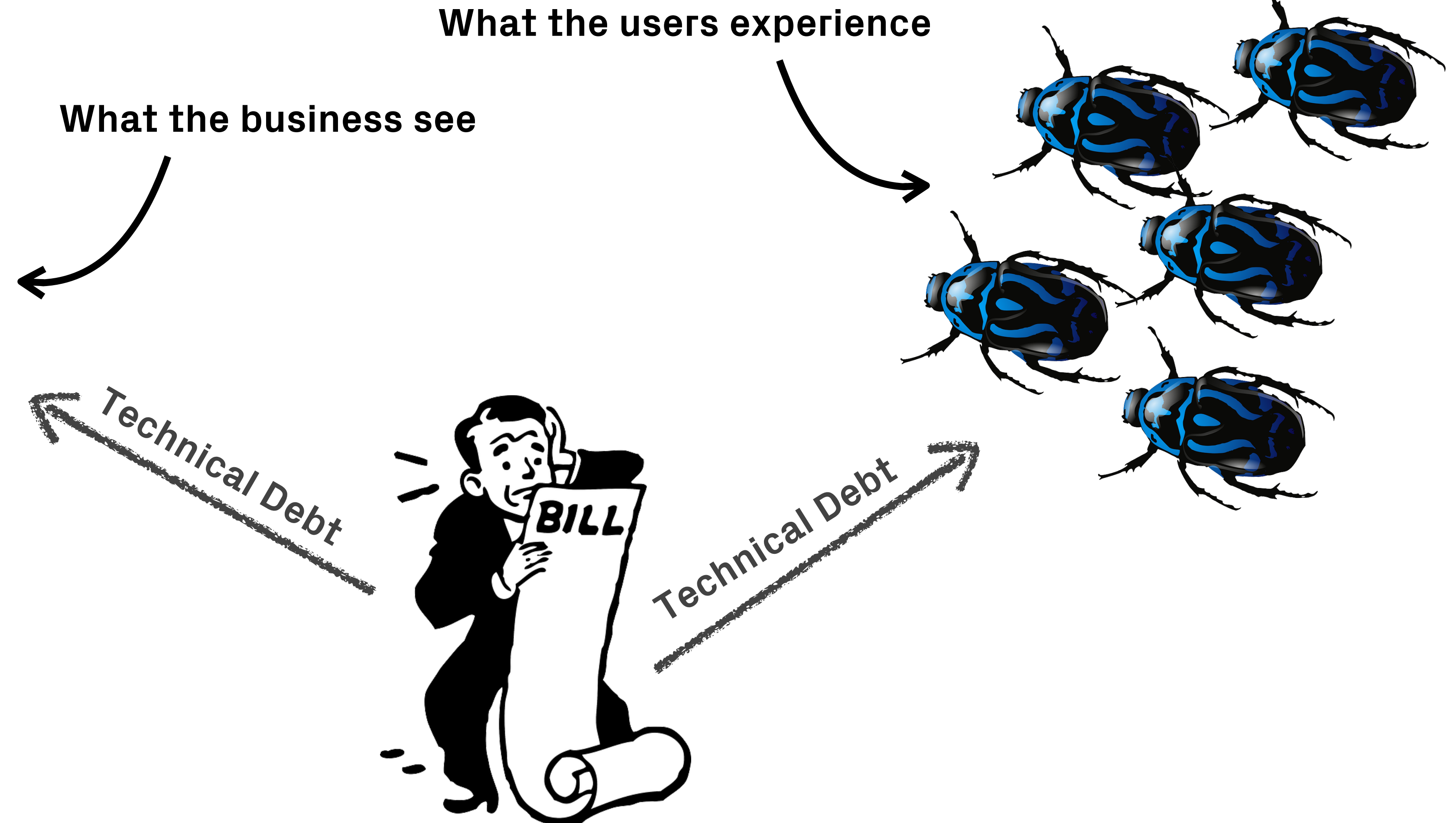
Are We Treating Symptoms Instead of the Real Issues?

Roadmap



Long Lead Times
Lack of Predictability

Product



QUANTIFYING TECHNICAL DEBT?



?

==



The background of the slide is a dense, repeating pattern of ancient Egyptian hieroglyphs. The hieroglyphs are carved into a light-colored, textured surface, possibly stone or papyrus, and are arranged in vertical columns. The symbols include various figures, animals, and geometric shapes, all rendered in a classic, stylized manner. The overall color palette is warm, with shades of beige and light brown.

4000 Years of Technical Debt?

4000 Years Ago => The Start of Recorded History

THE PERILS QUANTIFYING TECHNICAL DEBT

Q: What behaviour do we reinforce by quantifying technical debt?

important!





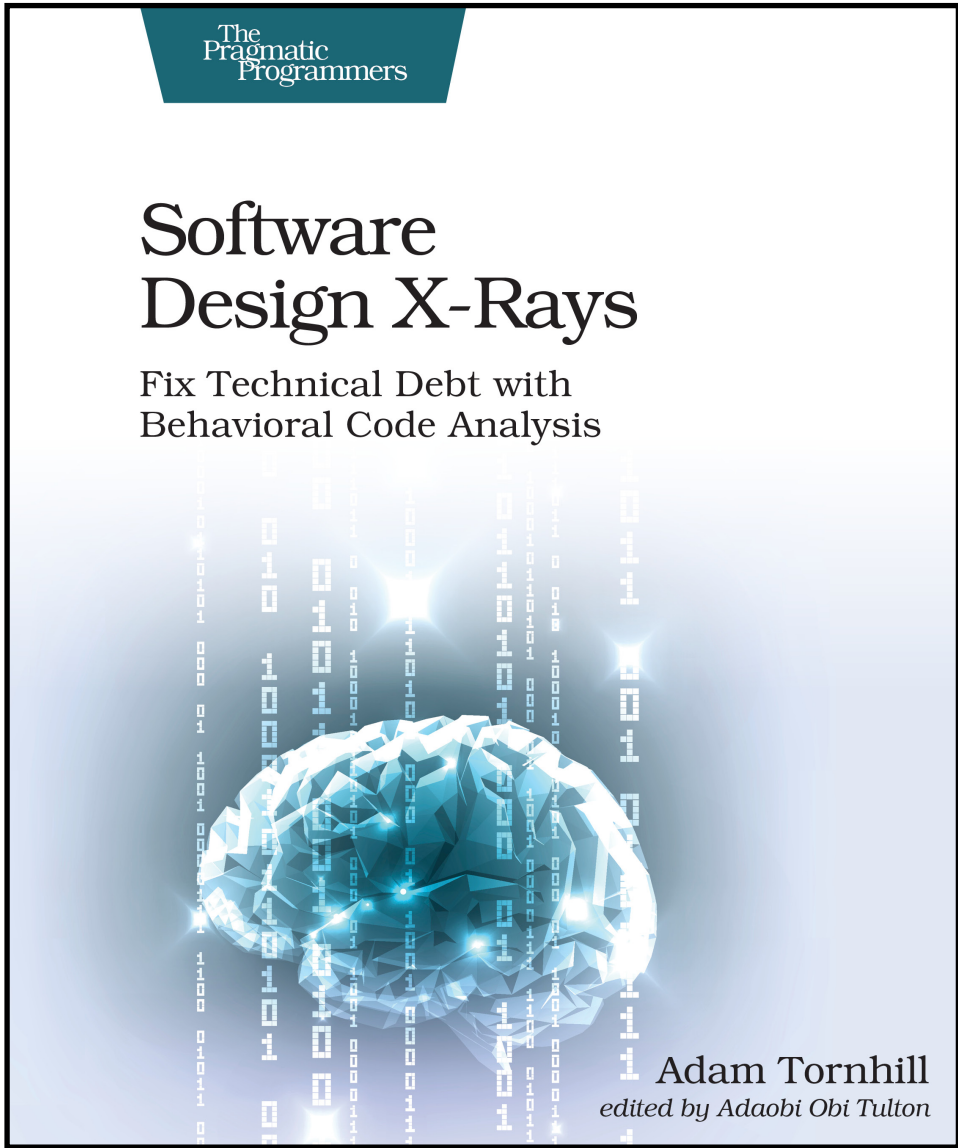
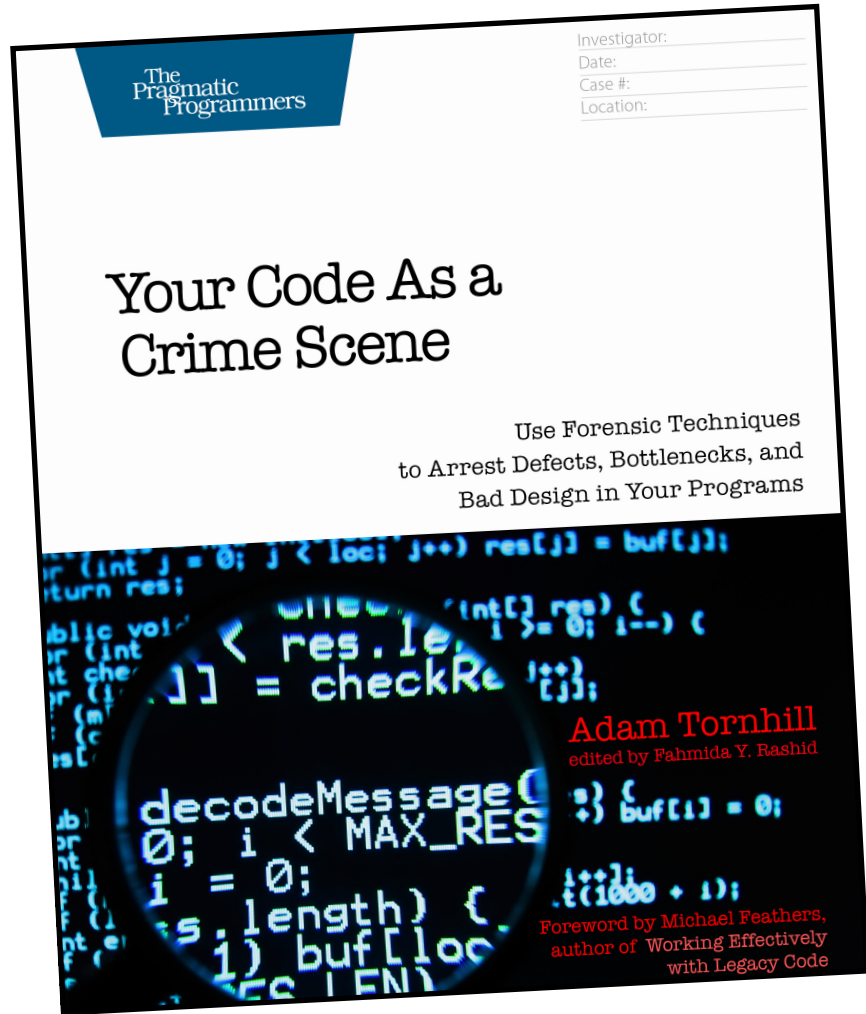
Quantifying Technical Debt Isn't Actionable.

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?*

Version-Control — A Behavioral Data Source



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

Fix behavior of StartsWithPrefix

0 27 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

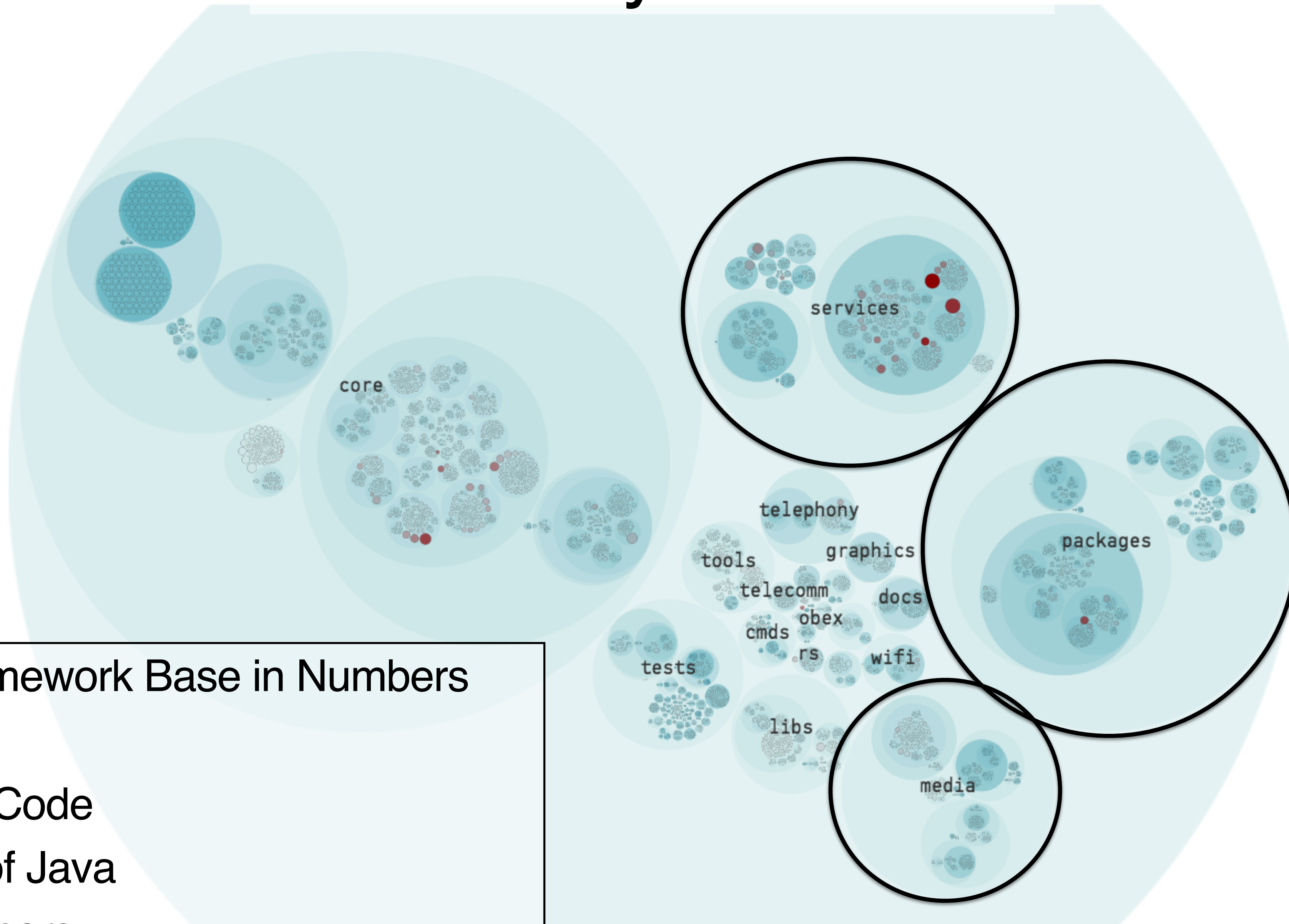
A Time Dimension

CASE STUDY: PRIORITIZING TECHNICAL DEBT

CodeScene™

Powered by Empear

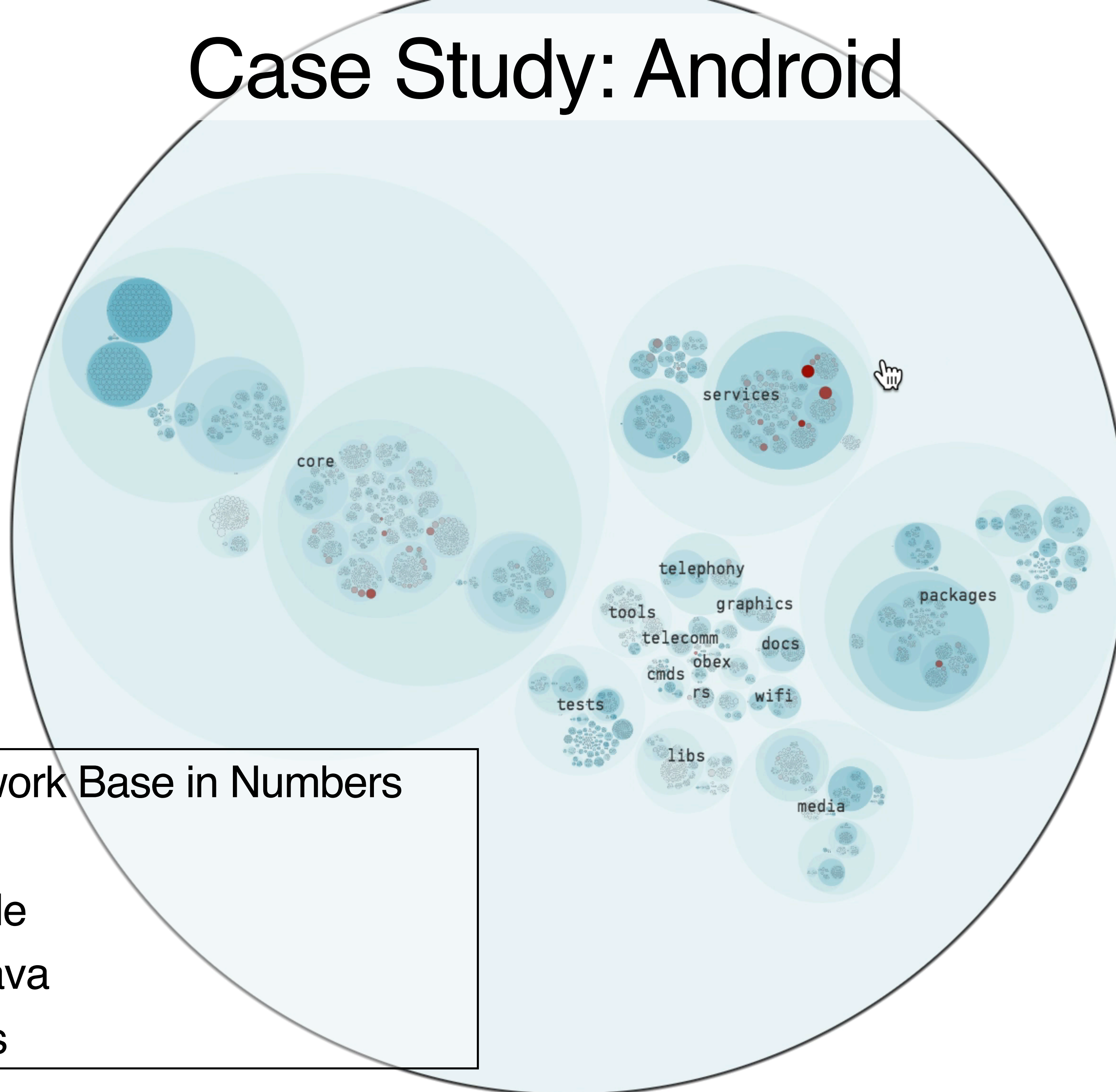
Case Study: Android



The Platform Framework Base in Numbers

3 Million Lines of Code
2,1 Million Lines of Java
2,000 Unique Authors

Case Study: Android



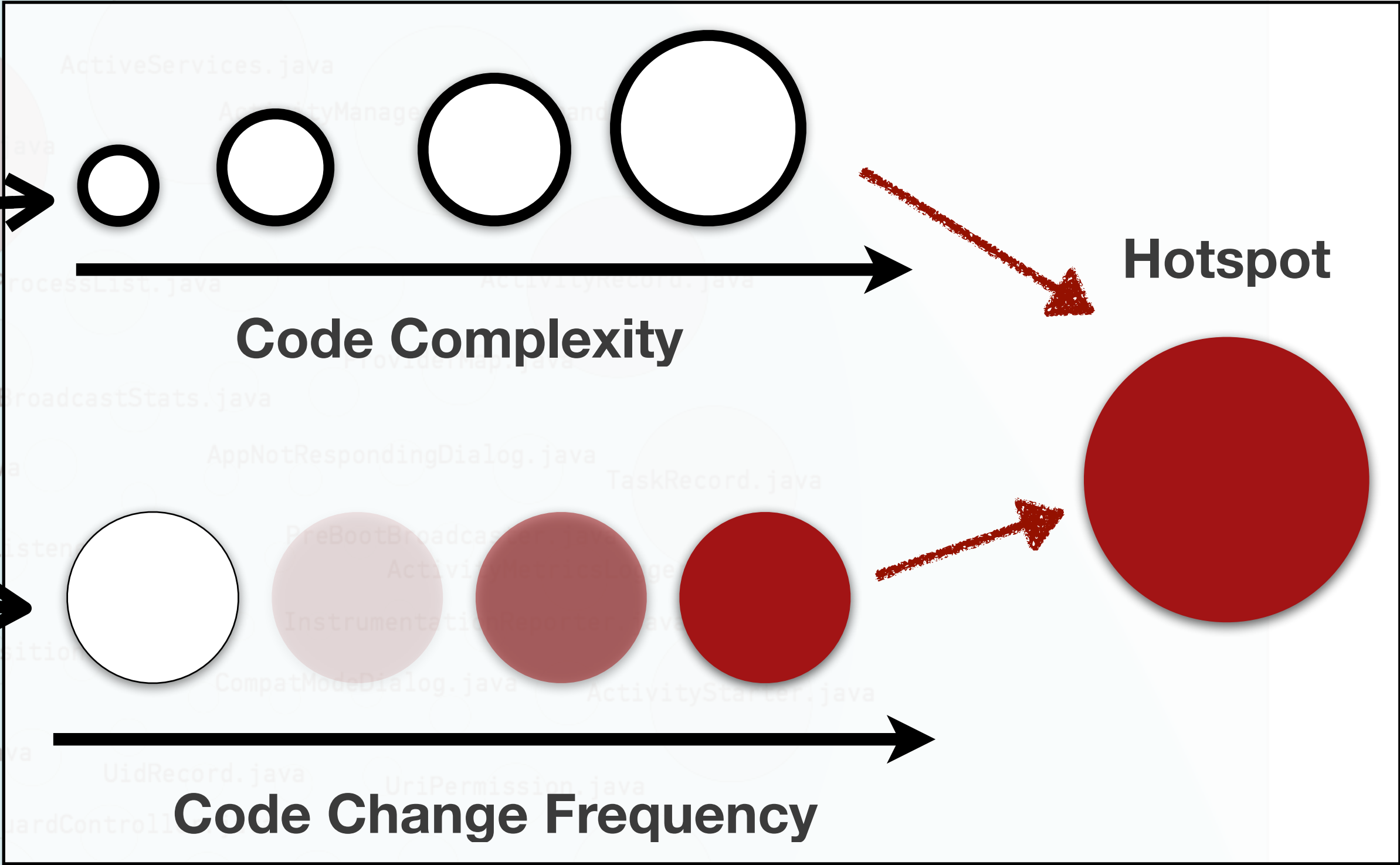
The Platform Framework Base in Numbers

3 Million Lines of Code
2,1 Million Lines of Java
2,000 Unique Authors

Principal

ActivityManagerService.java

Interest Rate



Investigate Hotspots: Beyond a Single Metric

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)

Symptoms of low Code Health



Low Cohesion, many responsibilities

Deeply Nested Logic, if-statements inside if-statements

Bumpy Road Code Smell, lack of encapsulation

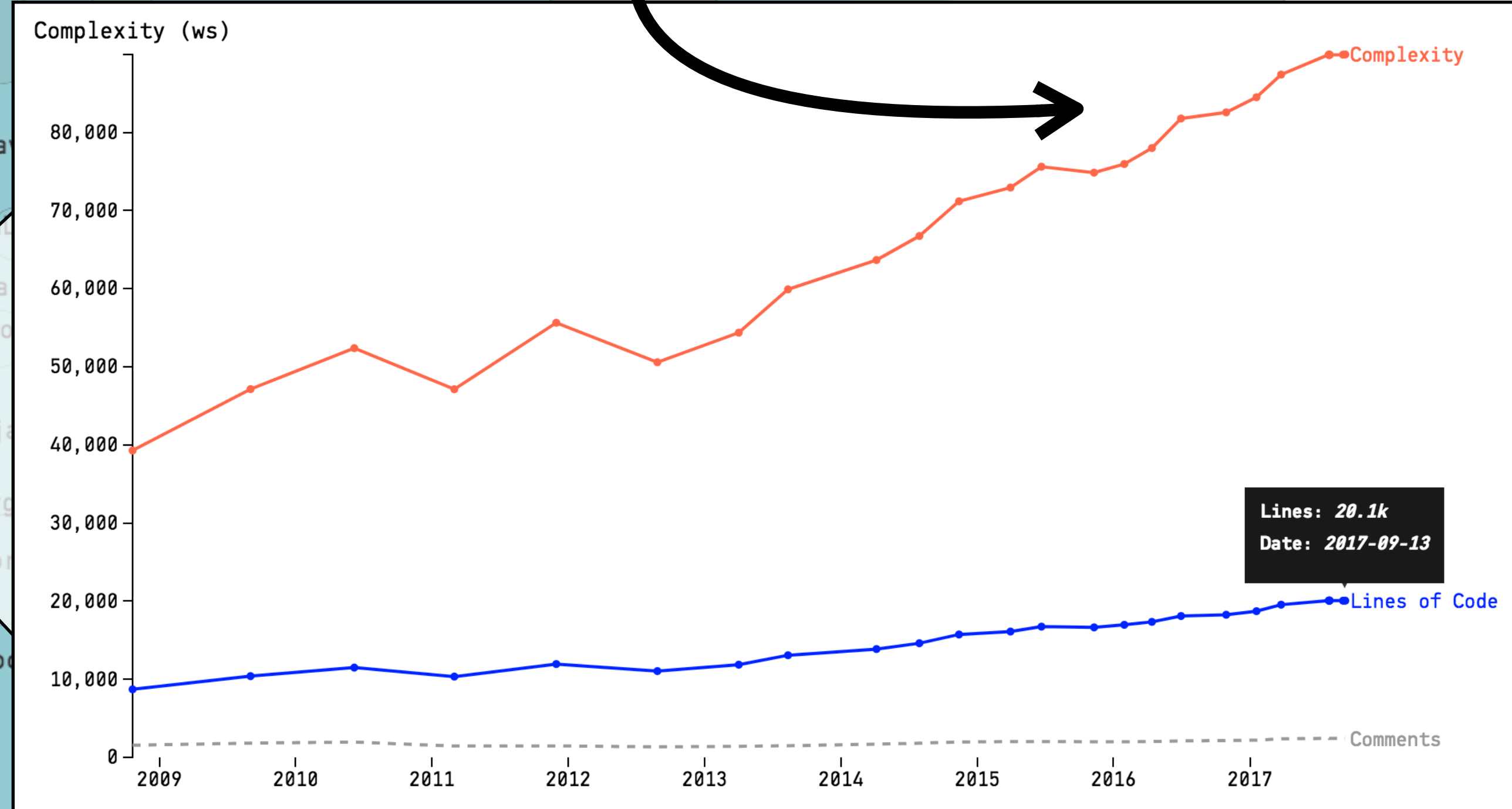
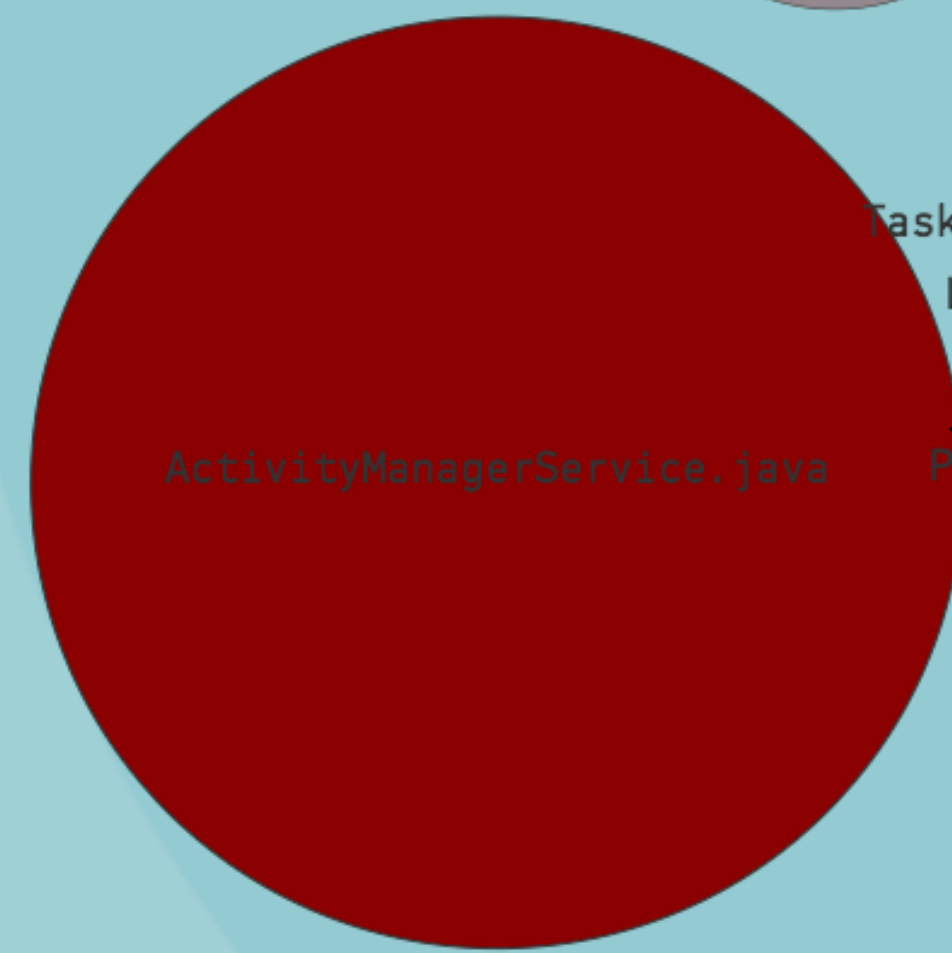
Primitive Obsession, missing a domain language

Excess Function Arguments, missing abstractions

Trends over Absolute Values

A Growing Problem

ActivityManagerService.java



Read More: <https://codescene.com/blog/code-biomarkers/>

Actionable Insights?

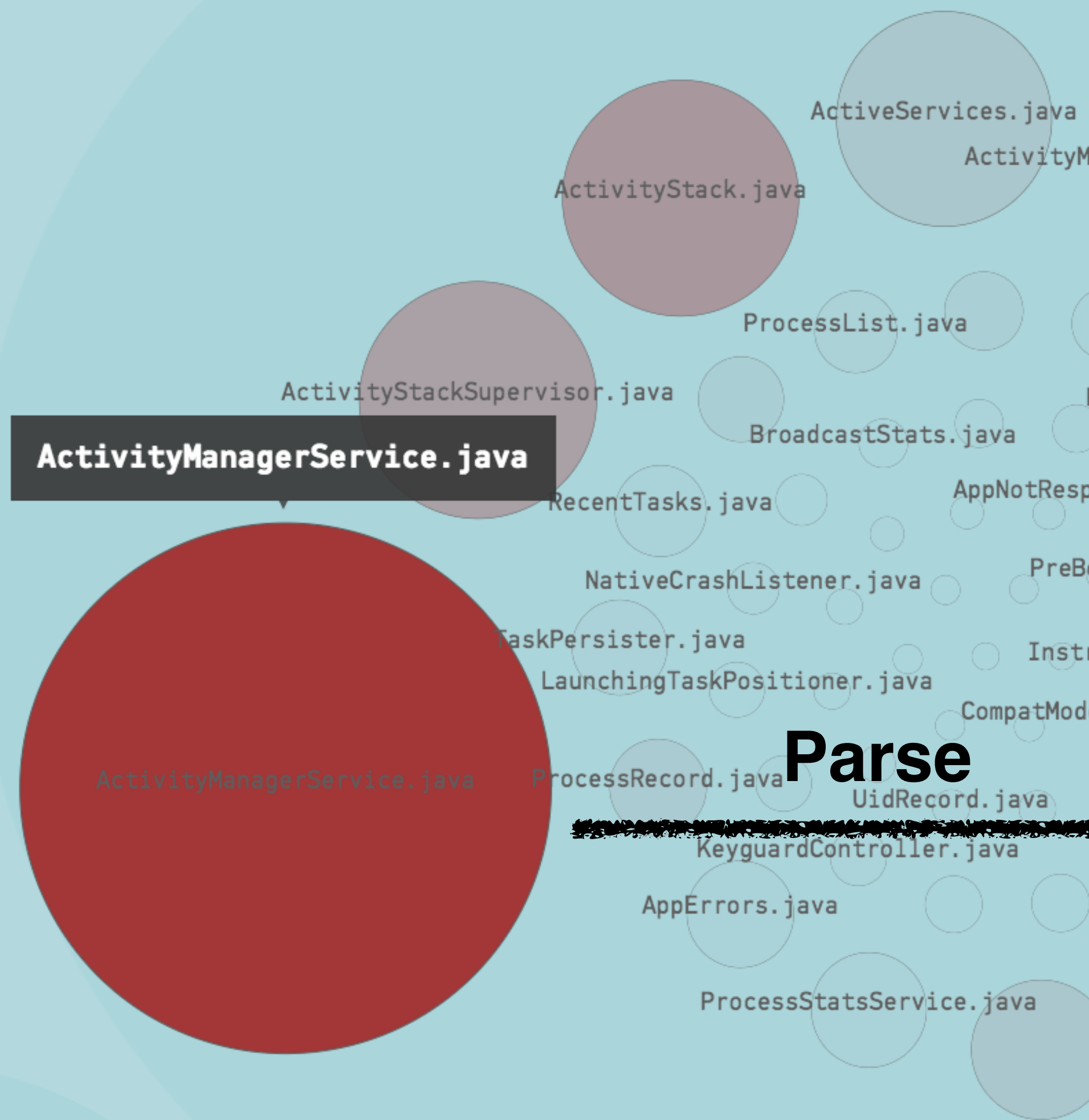
ActivityManagerService.java

20,000 Lines of Code!

74 Developers over the past 3 Months

Where Do We Start?

Hotspots: X-Ray `ActivityManagerService.java`

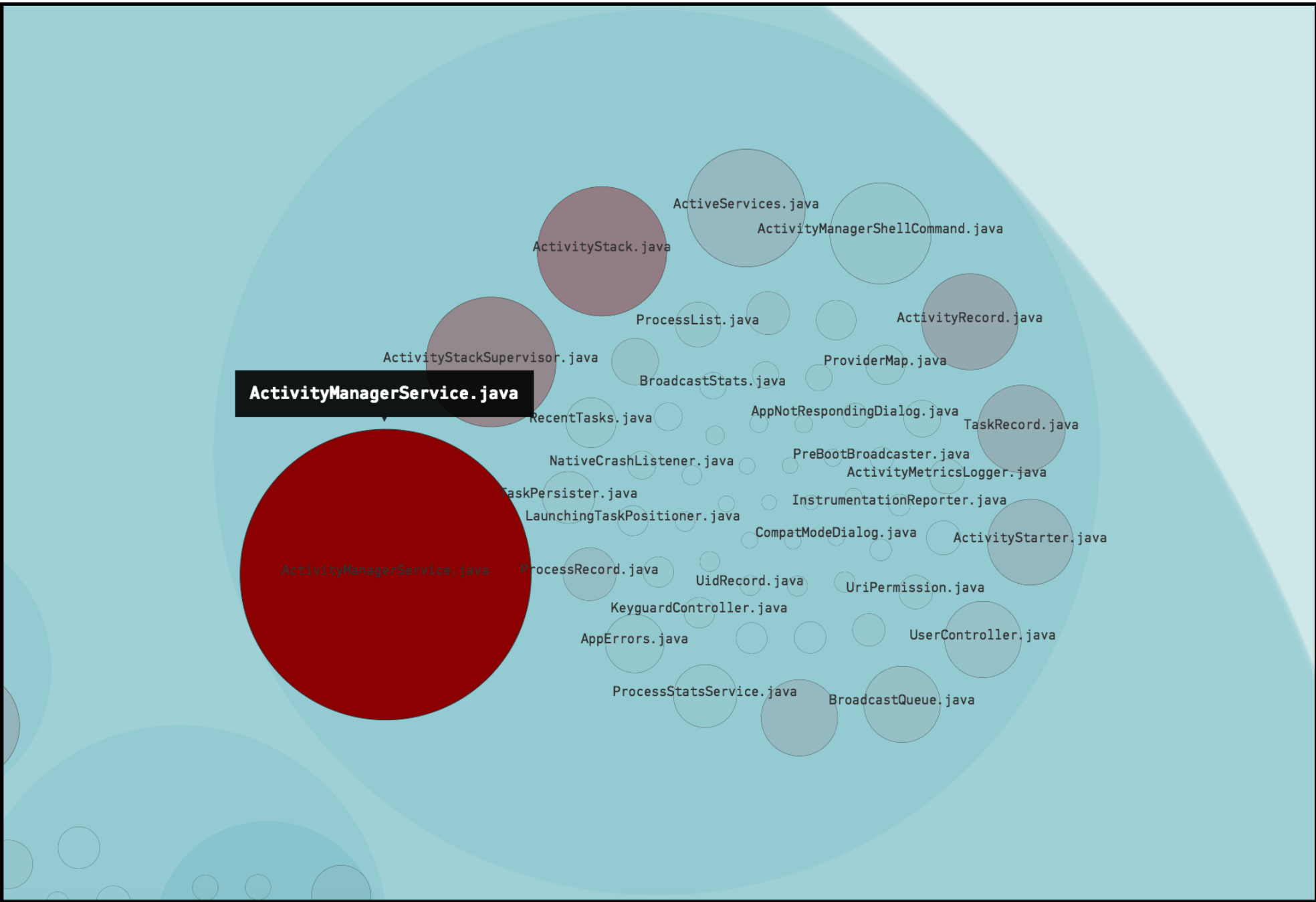


Function Level Hotspots



Recommended functions to improve.

X-Ray of ActivityManagerService.java



◆ Function	◆ Change Frequency	◆ Lines of Code	◆ Cyclomatic Complexity
ActivityManagerService.MainHandler.handleMessage	98	500	106
ActivityManagerService	75	160	12
applyOomAdjLocked	73	256	72
dumpStackTraces	73	188	40
dumpProcessesLocked	69	430	120
broadcastIntentLocked	60	647	171
enterPictureInPictureMode	60	71	7

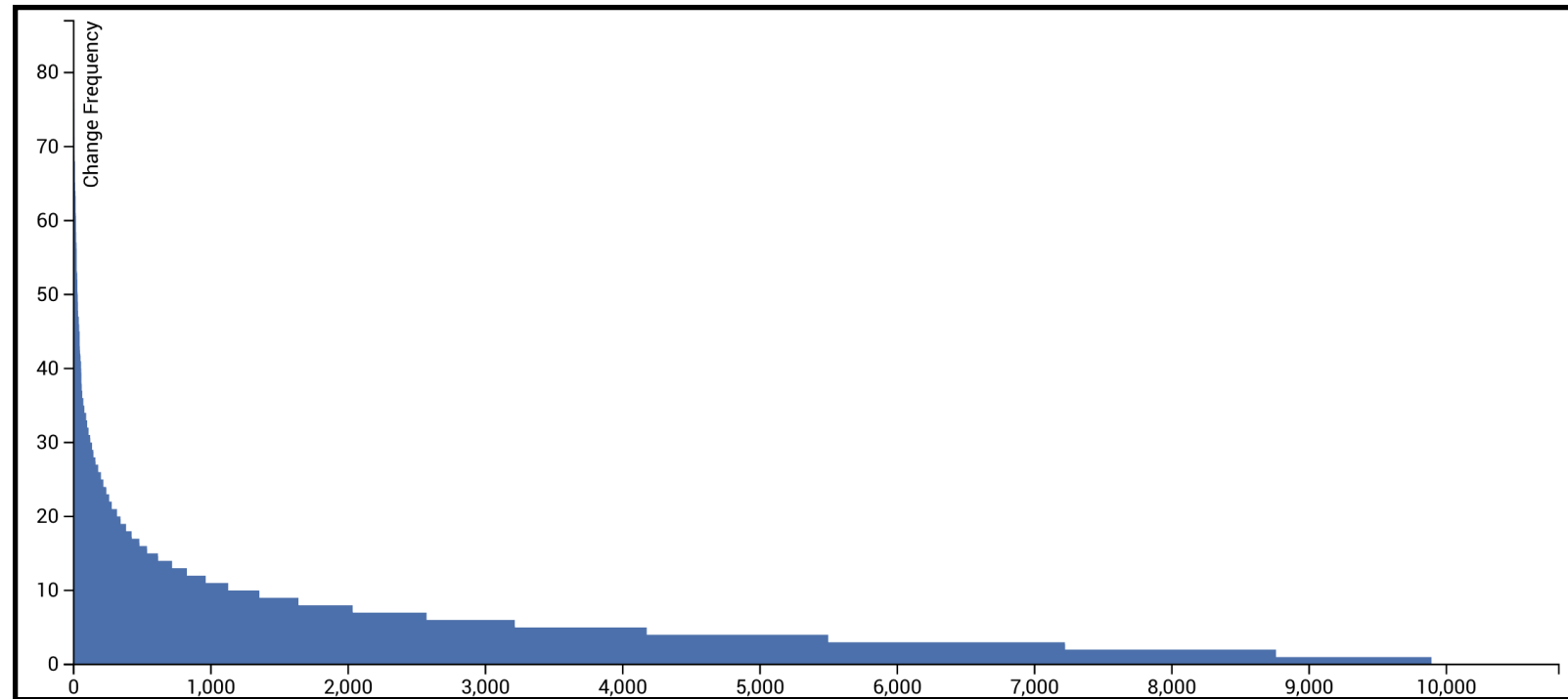
Why Hotspots Work

CodeScene™

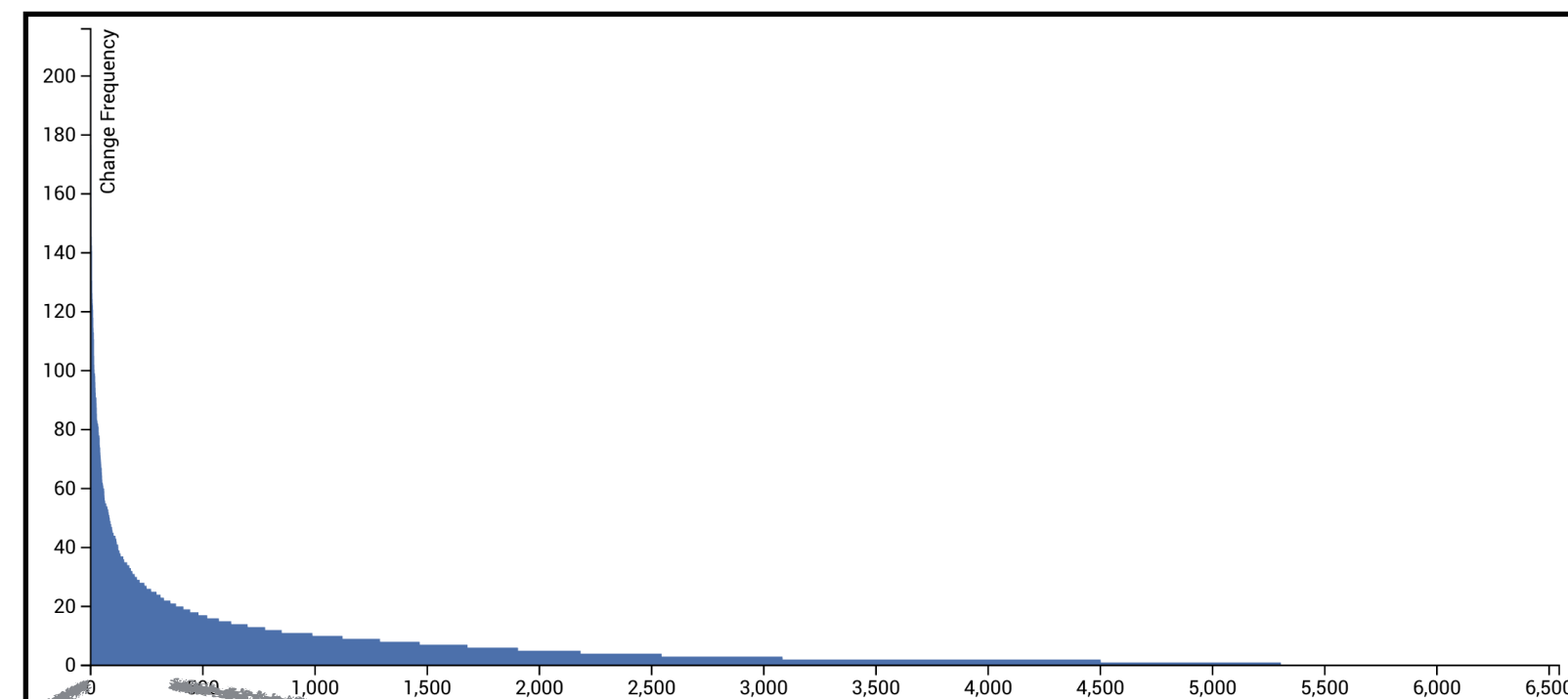
Powered by Empear

Why You Don't Have To Fix All Technical Debt

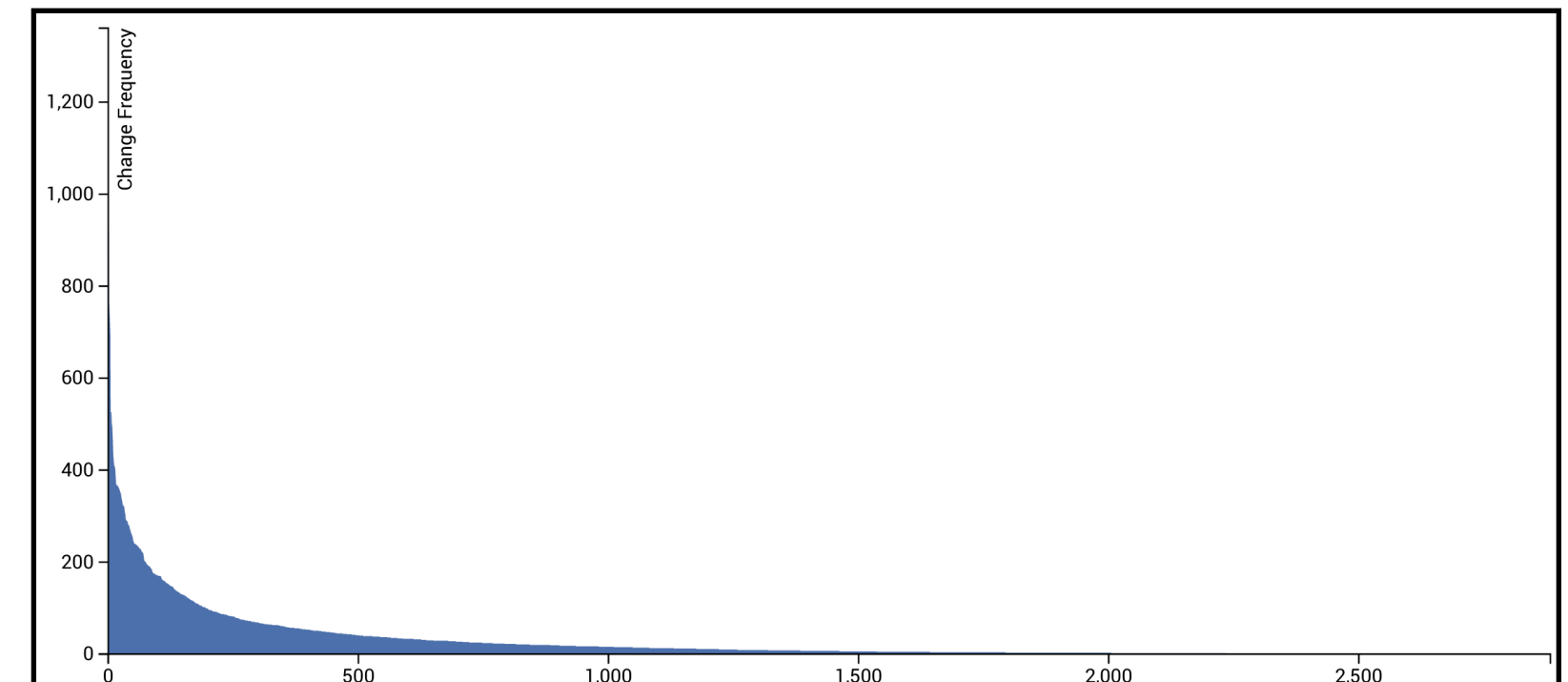
1 Year in Roslyn (C#, VB)



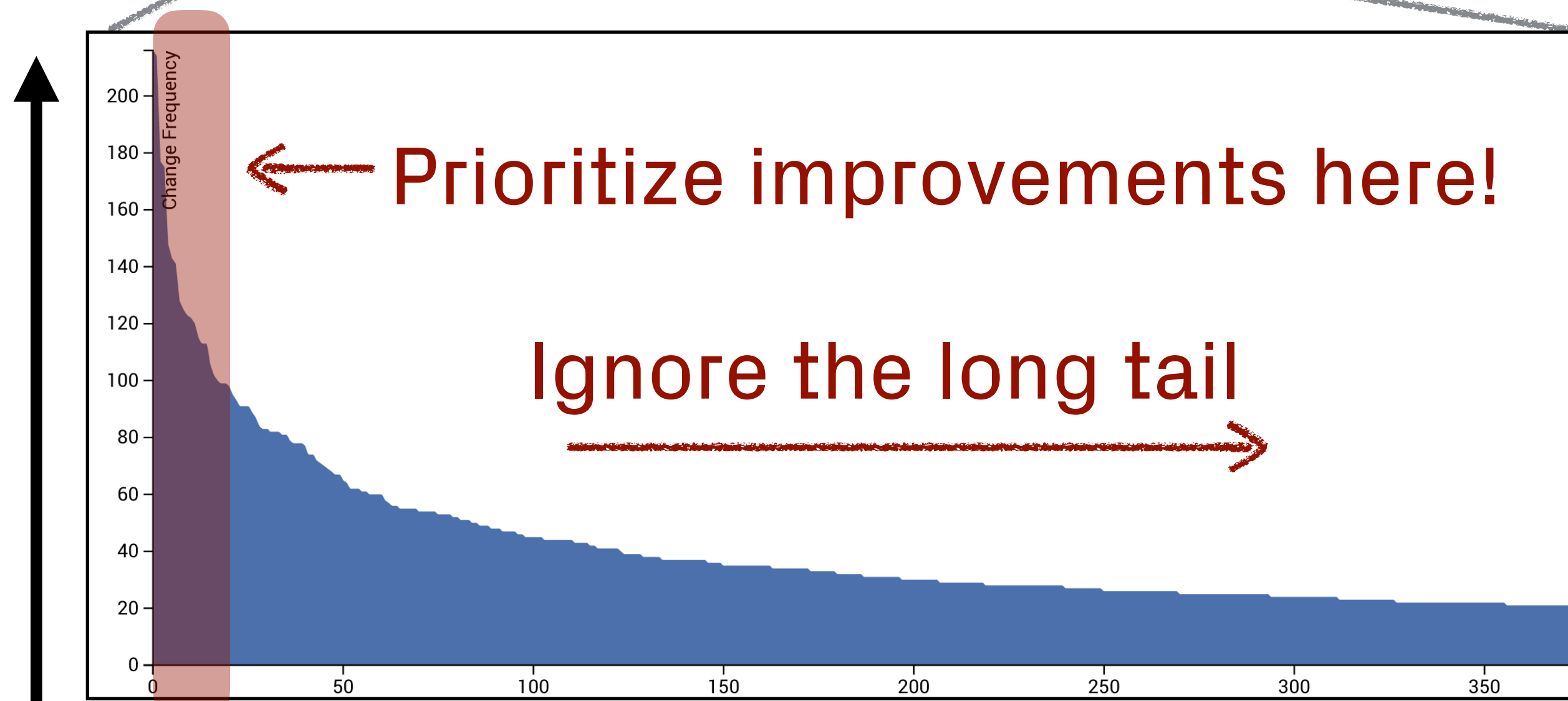
6 Years of Erlang



12 Years of Ruby on Rails



Change Frequency



Each file in the system

Code Quality In Context: Why you shouldn't fix all code issues

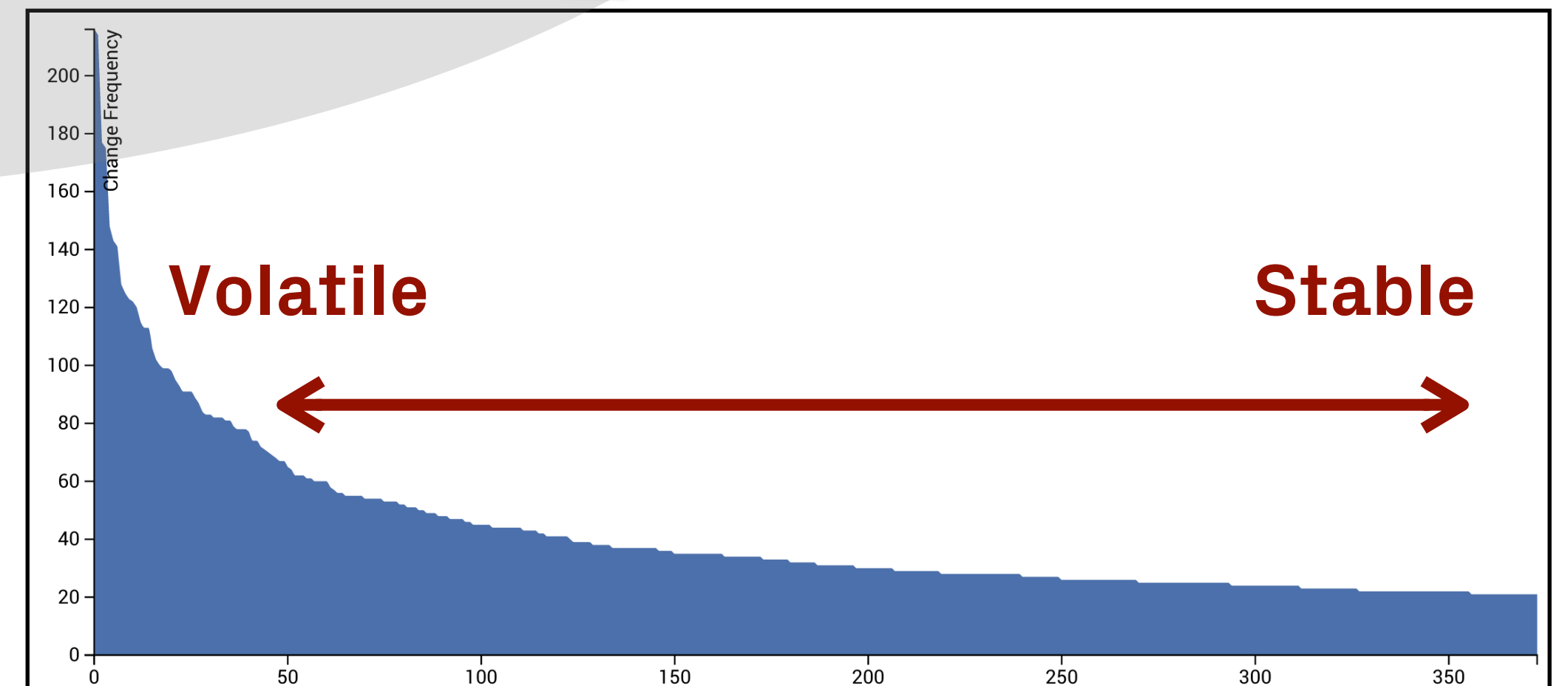
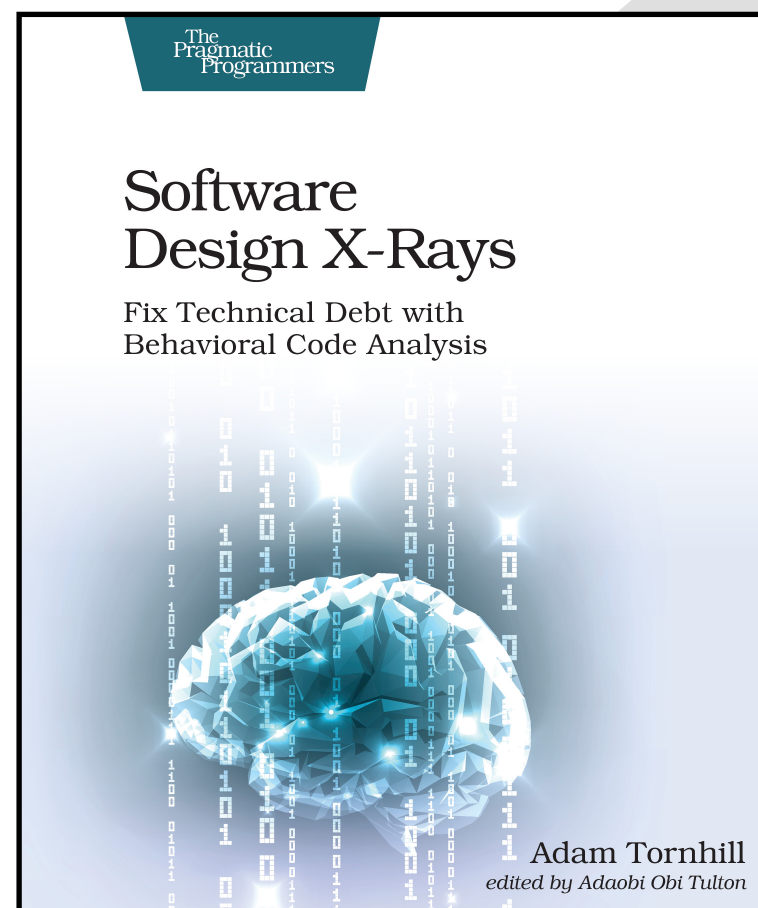
Your Best Bug Fix Is Time

The Three Generations of Code • 79

code. A team of researchers noted that a module that is a year older than a similar module has roughly one-third fewer faults. (See [Predicting fault incidence using software change history \[GKMS00\]](#).) The passage of time is like a quality verdict, as it exposes modules to an increasing number of use cases and variations. Defective modules have to be corrected. And since bug fixes themselves, ironically, pose a major risk of introducing new defects, the code has to be patched again and again. Thus, bugs breed bugs and it all gets reflected as code that refuses to stabilize and age.

Test Cases Don't Age Well

While old code is likely to be good code in the sense that it has low maintenance costs and low defect risk, the same reasoning doesn't



Read more: <https://adamtornhill.com/articles/code-quality-in-context/why-i-write-dirty-code.html>

The Legacy Code Link: Why Much Technical Debt Isn't Really Technical Debt

CodeScene™

Powered by Empear

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?

CodeScene™

Powered by Empear

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

- Active Contributors
- Former Contributors (knowledge loss)
- Simulated Knowledge Loss

Application Code

Test Code

src
benchmarkapps
benchmarks

test

~180 Contributors
350,000 Lines of Code
C#

Microsoft.AspNetCore.Mvc.IntegrationTests

test

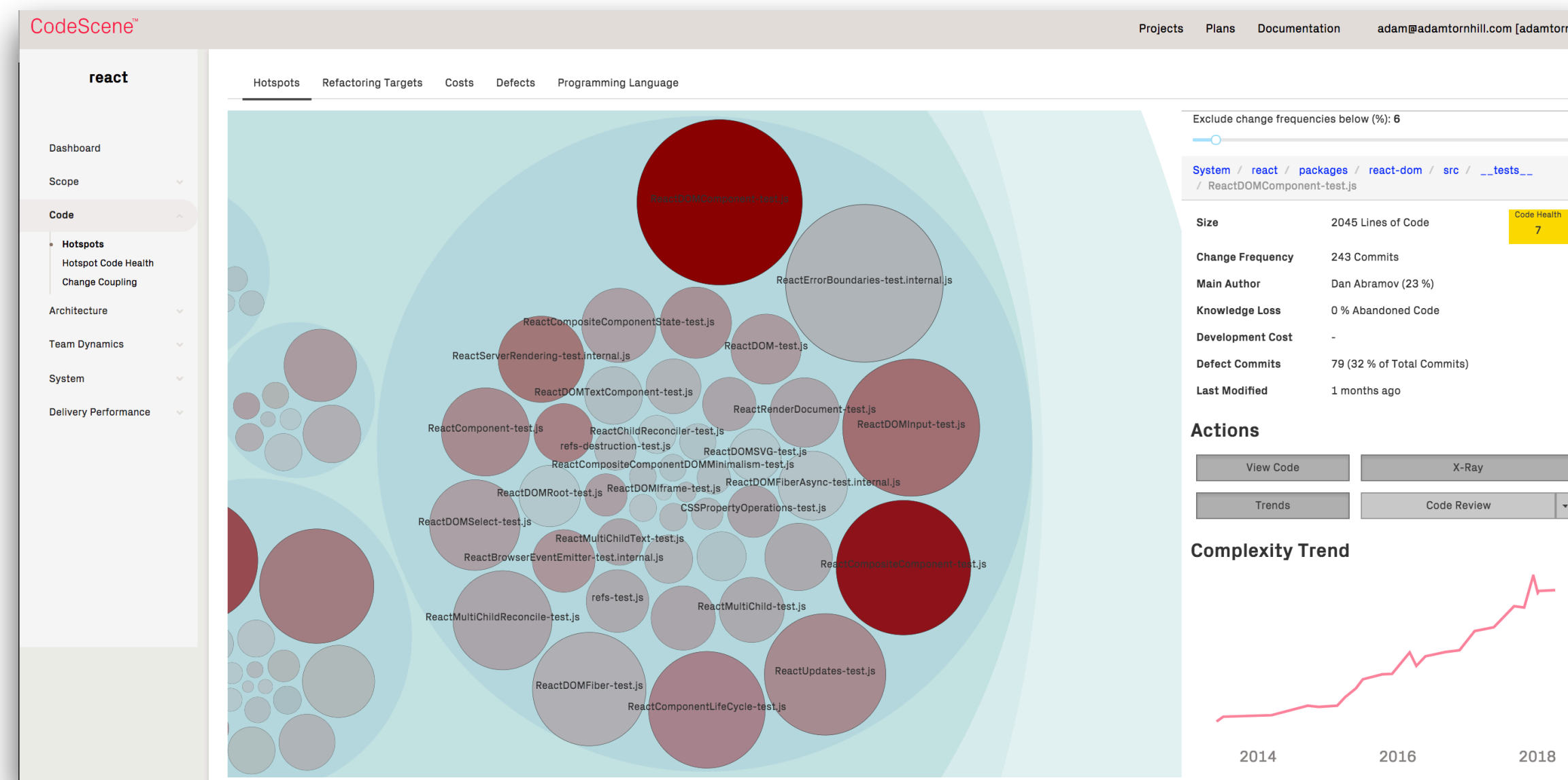
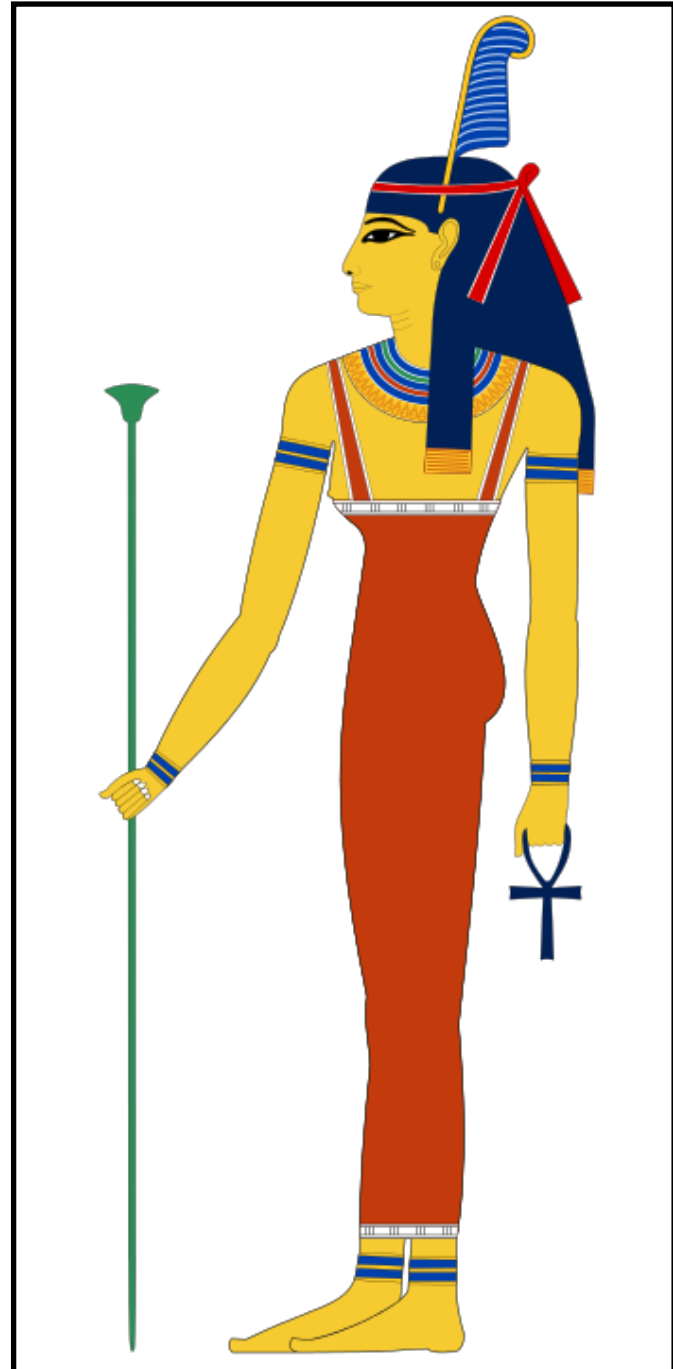
Code from <https://github.com/aspnet/Mvc>

A photograph of three large icebergs floating in a calm, blue ocean under a clear sky. The icebergs are white with jagged, textured surfaces. Their reflections are visible in the still water. The text is overlaid on the top part of the image.

There's More to Code Complexity than Code

Social Factors Influence how we Perceive a Codebase

Tooling: Try it on your own Code



<https://codescene.io/>



Track functions with
`git log -L :<funcname>:<file>`

Source Code:
<https://github.com/adamtornhill/code-maat>

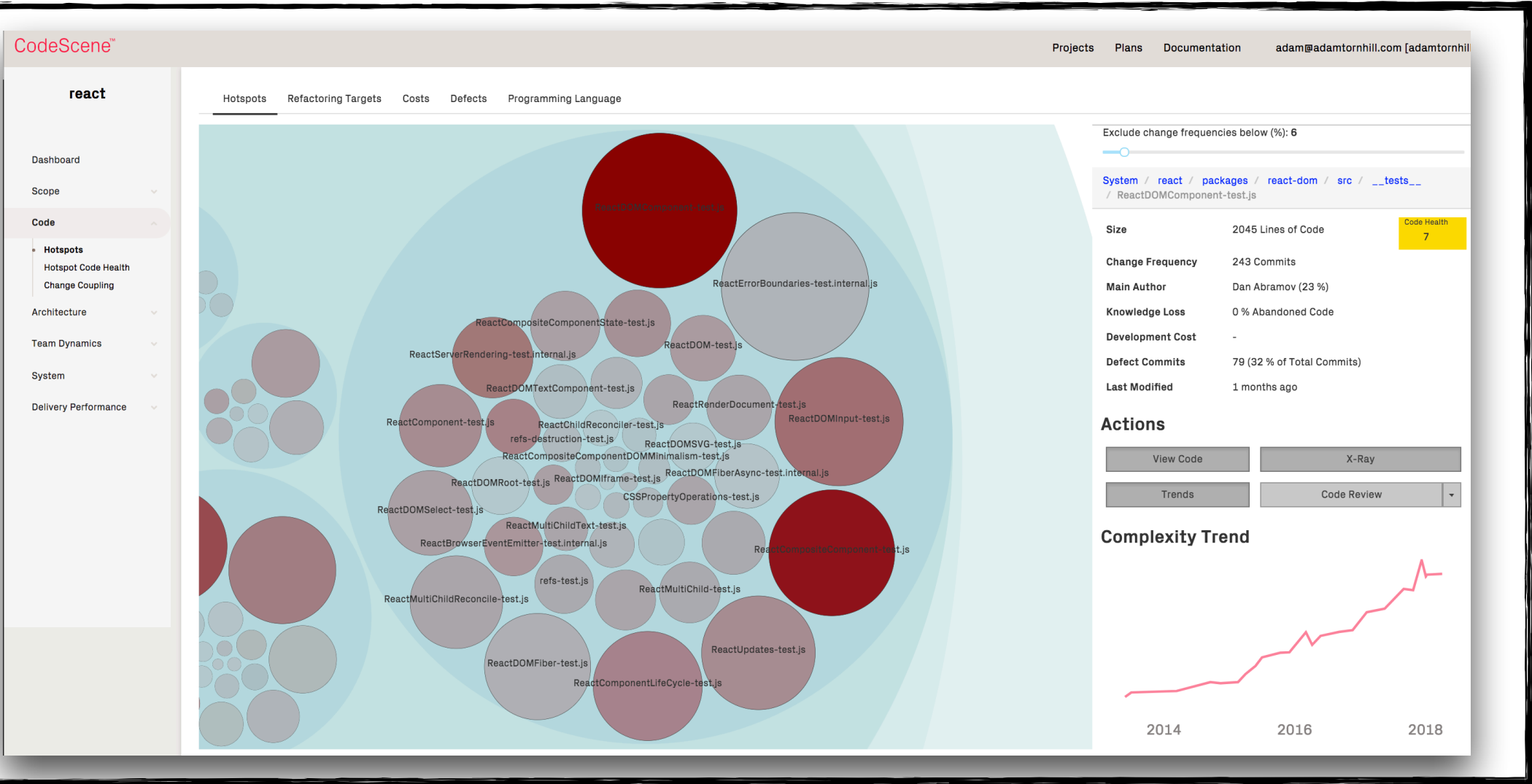
The background features a large, light blue circular area containing a complex network of smaller circles and clusters. These clusters are labeled with terms like 'mono', 'coreclr', 'libraries', 'installer', and 'tests'. The circles vary in size and color, with some being red and others blue, suggesting different levels of activity or importance in the analysis. The overall effect is a dense, interconnected web of data points representing code behavior.

Behavioral Code Analysis: A Communication Tool

Tools + Interactive Analysis

Examples:

<https://codescene.io/>



Blogs on Software Evolution, Technical Debt, and Programming

<https://www.codescene.com/blog/>

<https://adamtornhill.com/>

@AdamTornhill

LinkedIn: <https://www.linkedin.com/company/empear-ab/>

Twitter: <https://twitter.com/codescene>

