

How Proemion Used CodeScene and Conflux to Optimize Team and System Boundaries for Flow

Proemion, a leading telematics provider, used CodeScene and Conflux guidance to optimize team performance and flow. By adopting Team Topologies principles, they identified and addressed issues like declining code quality and refactoring hotspots. CodeScene provided actionable insights, improving team alignment and minimizing dependencies, leading to faster time-to-value and better engineering efficiency.

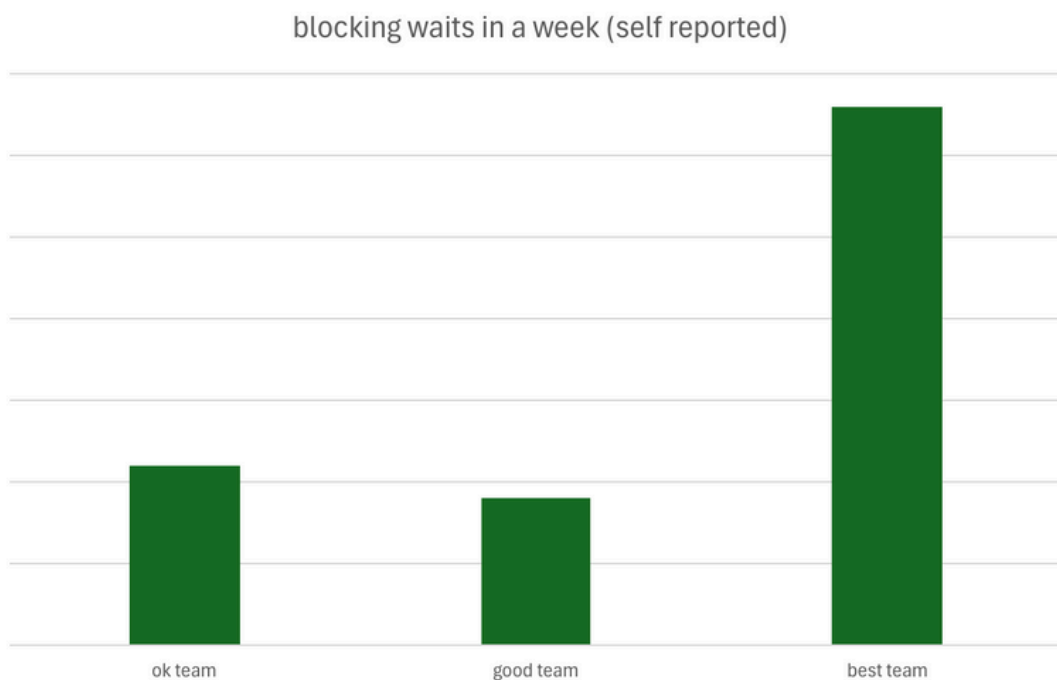


Establishing Clear Boundaries for Team Autonomy and Collaboration at Proemion

At Proemion, 60 engineers across 10 teams develop and scale the flagship DataPortal to meet growing demand. To ensure sustainable growth, Head of R&D Umberto Nicoletti adopted fast flow principles like Team Topologies.

As the engineering team grew, establishing clear boundaries for flow became essential for autonomy and collaboration. In 2024, inspired by a talk from Conflux CEO Matthew Skelton, they adopted CodeScene to evaluate codebase health.

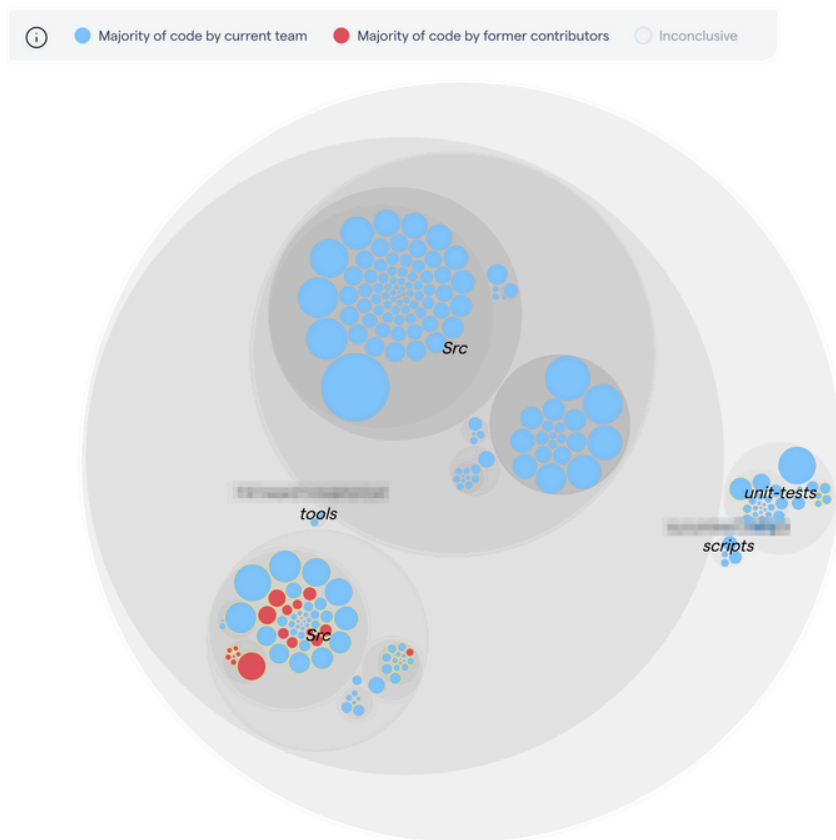
Nicoletti initially assessed team independence by counting "blocking waits," but the results were counterintuitive: the highest-performing team reported the most blockers. He later realized this reflected the team's maturity and awareness of its challenges.



Using Team Analysis in CodeScene to Map to the Team Topologies Principles

CodeScene was used to compare the context of different teams, providing clarity and pinpointing the problems faced by the lower-performing team.

“CodeScene aligned perfectly with our observations,” said Nicoletti. “It revealed the underperforming team faced multiple challenges: declining code quality, refactoring hotspots, knowledge loss, and low effectiveness despite team expansion.”

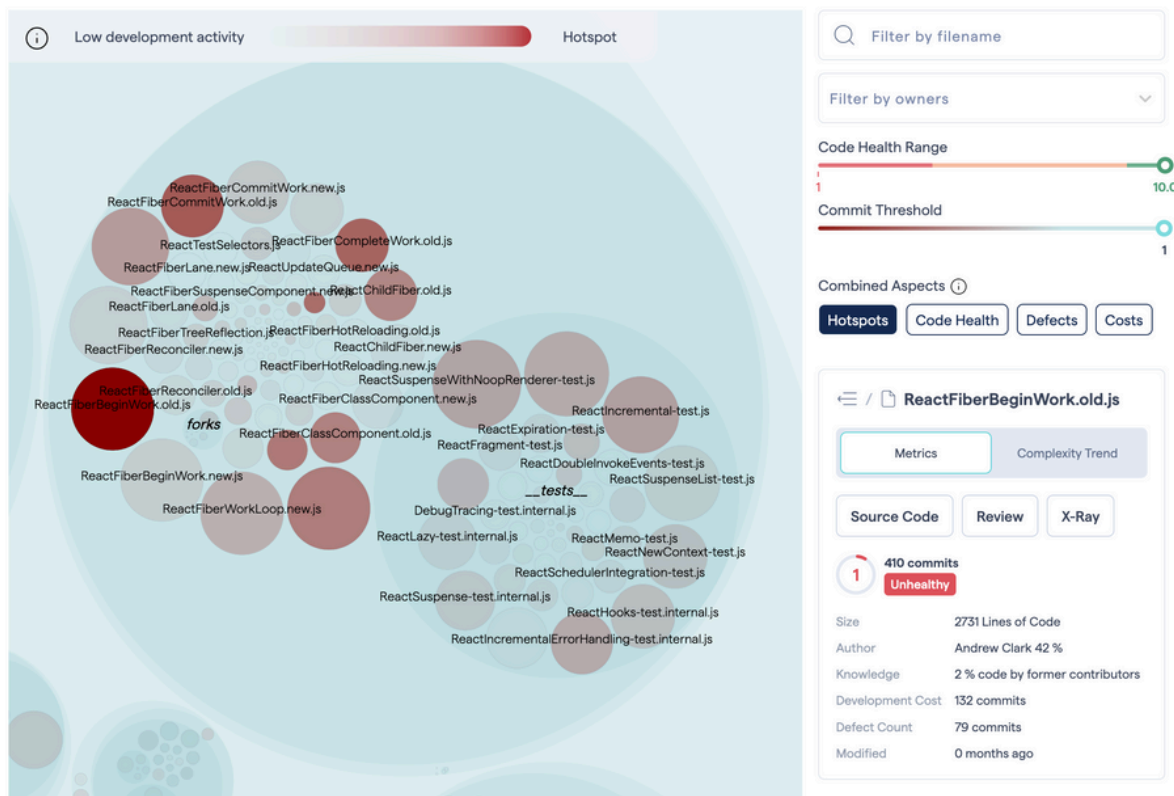


It visualizes team boundaries through code activity patterns, making it easy to spot misalignments between software and team architecture, inspired by Team Topologies.

Nicoletti also used CodeScene to assess team boundaries and cross-team dependencies, confirming that transitioning from waterfall to stream-aligned teams had minimized blocking waits. “It confirmed that we didn’t have major blocking issues after the reorganization,” said Nicoletti.

What Makes CodeScene Unique Compared to Tools Like SonarCloud?

“CodeScene’s focus on organizational dynamics is invaluable for team leads, managers, and CTOs,” said Nicoletti. “It helped us identify that the lower-performing team wasn’t addressing hotspots, a key insight that tools like SonarCloud miss.”

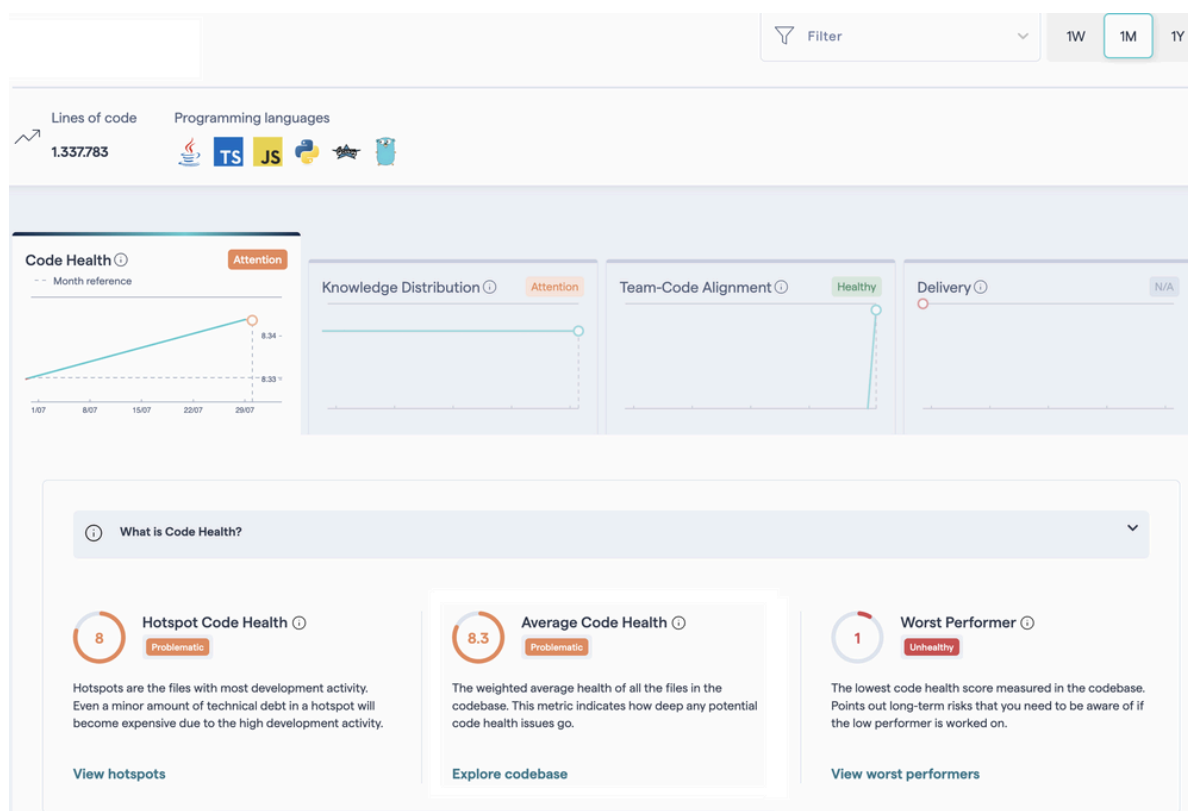


Hotspots let you visualize large codebases by combining a code health perspective with temporal and organizational data

Hotspots: These indicate the priority of issues, calculated from development activity in the code. They highlight areas that need attention based on how frequently they’re worked on.

Low Code Health: This reflects the severity of a hotspot. The lower the code health, the redder the circle representing the hotspot. Hotspots help identify maintenance problems — complex, frequently worked-on code that can be difficult to manage.

Clear Software Metrics Reflecting Engineering Efficiency and Code Health



CodeScene’s dashboard showed steadily increasing code quality and team alignment. This data matched the perceived team(s) performance as they work on the codebase that has been powering the Proemion telematics platform for over 15 years.

CodeScene provides actionable insights for reducing time-to-value

With guidance from Conflux, Proemion’s R&D department identified and addressed performance issues in their software delivery teams. “CodeScene supports team performance concerns with data and actionable next steps, excelling in revealing the dynamics that impact software delivery,” said Nicoletti.

Visit [Conflux for more information](#) or contact the [CodeScene Sales team](#) for further details