

Socio-Technical Smells: How Technical Problems Cause Organizational Friction

Adam Tornhill
CodeScene
adam.tornhill@codescene.com

Keynote abstract

Successful software development requires that you keep code and people in balance so that one supports the other. It's a hard challenge since a piece of code doesn't reveal anything about its socio-technical context. Enter behavioral code analysis, an approach which combines code level metrics with data on how teams interact within the code. Armed with these techniques, we look to reduce organizational friction by focusing on a set of common challenges:

- Identify architectural coordination bottlenecks and understand the technical root causes.
- Visualize implicit dependencies between teams, act to decouple teams.
- Discover knowledge risks by measuring the Truck Factor. Learn how to mitigate it.
- Communicate the scaling risks inherent in Brooks's Law by showing data on how it impacts your delivery.
- Go beyond technical impact by knowing how bad code causes unhappiness, low morale, and increased attrition.