Towards a Prioritization of Code Debt: A Code Smell Intensity Index

Francesca Arcelli Fontana, Vincenzo Ferme, Marco Zanoni, Riccardo Roveda

ESSeRE Lab, DISCo, University of Milano-Bicocca

2 Oct 2015, Bremen, Germany
Code smells and Technical Debt

Code smells: symptoms of problems at code or design level that can be resolved through the right refactoring steps

*Sources of Technical Debt*

The management of code smells involves different decisions:

- which code smells to refactor?
- what to refactor first? (prioritization)
Prioritization: Code smell Intensity

What is Intensity:

- a number in the range 1–10
- used to rank code smells

Intensity can be computed for detection strategies based on metrics and thresholds, e.g. (Shotgun Surgery),

\[ \text{CC} \geq \text{HIGH} \land \text{CM} \geq \text{HIGH} \land \text{FANOUT} \geq \text{LOW} \]

Intensity considers where the evaluated smell is placed in the metric distributions.

Example instance

- Metric values: CC = 8; CM = 10; FANOUT = 6
- Intensity: \( \frac{7.75 + 7.75 + 10}{3} = 8.5 \) (High) **Why?**
Metric distribution and thresholds (1)

We associate five points to Intensity value ranges on the distribution of each metric:

1. Very Low: [1, 3.25);
2. Low: [3.25, 5.5);
3. Mean: [5.5, 7.75);
4. High: [7.75, 10);
5. Very High: [10, 10].

Figure 1: FANOUT Intensity points for Shotgun Surgery
Metric distribution and thresholds (2)

In the example:

- **CC ≥ HIGH(5):** CC = 8 → High (7.75)
- **CM ≥ HIGH(6):** CM = 10 → High (7.75)
- **FANOUT ≥ LOW(3):** FANOUT = 6 → Very High (10)

→ \( (7.75 + 7.75 + 10)/3 = 8.5 \) (High)

<table>
<thead>
<tr>
<th>Metric</th>
<th>VERY-LOW</th>
<th>LOW</th>
<th>MEAN</th>
<th>HIGH</th>
<th>VERY-HIGH</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>CM</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>6</td>
<td>13</td>
</tr>
<tr>
<td>FANOUT</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
</tbody>
</table>
Can we use code smells and Intensity to enhance existing measures of Technical Debt?

How do developers behave when dealing with code smells of very different intensity levels?
Thank you!

marco.zanoni@disco.unimib.it