

O trabalho apresenta o conceito de Hotspot Patterns, que são problemas recorrentes de arquitetura em sistemas complexos responsáveis por altos custos de manutenção. Os autores formalizam cinco padrões principais, com base na Design Rule Theory de Baldwin e Clark:

1. Unstable Interface – interfaces centrais e influentes que mudam frequentemente, tornando-se pontos frágeis no sistema.

2. Implicit Cross-module Dependency – módulos que deveriam ser independentes, mas apresentam acoplamentos ocultos revelados pelo histórico de co-mudanças.

3. Unhealthy Inheritance Hierarchy – hierarquias de herança que violam

princípios de projeto (ex.: classe pai dependendo de filho, ou clientes dependendo tanto da classe base quanto de todas as subclasses).

4. Cross-Module Cycle – ciclos de dependência entre módulos, que aumentam a complexidade e a propensão a erros.

5. Cross-Package Cycle – ciclos de dependência entre pacotes, que quebram a hierarquia esperada da arquitetura.

Para detectar esses padrões, foi desenvolvido o Hotspot Detector, uma ferramenta que combina informações

estruturais (dependências) e históricas (comundanças extraídas de logs de versão). Ela gera matrizes de estrutura (DSM) e identifica automaticamente onde esses problemas ocorrem.

A avaliação foi feita em nove projetos open source da Apache e um projeto comercial. Os resultados mostram que:

Arquivos envolvidos em hotspots são muito mais propensos a erros e mudanças do que a média.

Quanto mais hotspots um arquivo acumula, maior sua taxa de bugs e mudanças.

Entre os padrões, Unstable Interface e Cross-Module Cycle foram os que mais contribuíram para a propensão a falhas.

Um estudo de caso industrial confirmou a utilidade prática: arquitetos e desenvolvedores validaram os problemas detectados e iniciaram refatorações guiadas pelas descobertas da ferramenta.