

The Scent of Design Smells: Should Developers Care About It?

FOUTSE KHOMH¹

Abstract: Design smells also known as Antipatterns identify “poor” solutions to recurring design problems. These design smells stem from experienced software developers’ expertise and are conjectured in the literature to negatively impact systems, for example by making classes more change-prone and-or fault-prone. They are generally introduced in systems by developers who do not have sufficient knowledge and/or experience in solving a particular problem or who have misapplied some design patterns. In practice, antipatterns are in-between design and implementation: they concern the design of one or more classes, but they concretely manifest themselves in the source code as classes through specific code smells.

In this talk I will present you a series of studies that we have conducted over the years to assess the impact of antipatterns and code smells on different software quality attributes. I will show you how anti-patterns and code smells affect the change- and fault-proneness of classes and their understandability. I will also show how certain antipatterns significantly impact the energy efficiency of mobile applications.

¹ Polytechnique Montréal