**pasaje**

Aspect-oriented software development (AOSD) tries to solve the problem of separating the core functionality of a software system from concerns that have a more system-wide behavior and that tend to cut across the chosen decomposition of the software system. This problem is sometimes referred to as the “tyranny of the dominant decompositon” [37]. To overcome this prevalent decomposition [18], the AOSD paradigm provides new language constructs which allow cross-cutting concerns to be written down in a new kind of module named aspect.

AO

in order to apply aspectoriented techniques to legacy systems at use in industry, or to migrate those systems to an aspect-oriented solution, there is a need for tools and techniques that help in identifying the cross-cutting concerns in such systems and refactoring them into aspects. The study and development of such approaches is the objective of the emerging research domains of ‘aspect mining’ and ‘aspect refactoring’ : Aspect Mining is the activity of discovering, in the source code of a given software system, those cross-cutting concerns that potentially could be turned into aspects. We refer to such concerns as ‘aspect candidates’. Aspect Refactoring is the activity of actually transforming the identified aspect candidates into real aspects in the source code.