**AO**

Aspect Oriented Programming (AOP) is used to reduce complexity, increase readability, and improve modularity in software systems. In large software systems, complexity, readability, and modularity remain major obstacles[8], [12]. These systems, which could benefit the most from refactoring into

AOP, are the last systems that are actually refactored, due to the amount of time and effort that would be required.

AM y AR

Typically, programmers decide to exploit AOP when they are implementing a new task in a program that, because of its nature, is going to require adding code in locations scattered throughout many functions, classes, or files. However, the debugging, testing, and maintenance of large legacy systems also could be eased considerably if already existing code for these kinds of tasks could be identified and refactored into AOP style[8], [12].