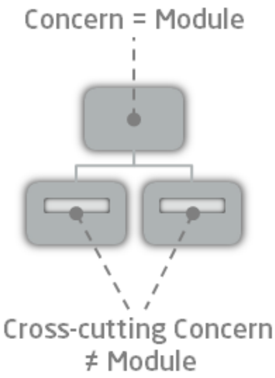
Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

Software Architecture Group
Prof. Dr. Robert Hirschfeld
Master's Project Proposal, Winter Term 2012/13

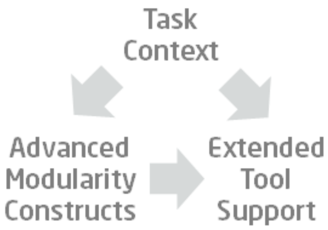
Recognizing Cross-cutting Concerns by Revealing Task Contexts

The Background

Programmers benefit from a modular system structure during software maintenance activities. Single modules are easy to create, comprehend, and extend with the given programming language and supporting tools. However, if one concern cannot be implemented in one module, programmers need to *scatter and tangle* source code across modules. By doing so, these concerns will crosscut other decompositions—including the dominant one. This situation is common for state-of-the-art languages like C#, Java, or Smalltalk. In the long term, code quality can decrease because the software becomes difficult to understand and modify.



There are ways to overcome those modularity challenges. However, programmers often fail to fully recognize cross-cutting concerns. Thus, they have difficulties in deciding when to choose advanced modularity concepts (e.g., AOP or COP) or make use of appropriate tool support during software maintenance activities.



Our Idea

When accomplishing tasks, programmers continuously work (e.g., explore, modify, run) with source code artifacts (e.g., classes, methods) that belong to cross-cutting concerns. However, these groups of artifacts form implicit *task contexts*, which are not apparent to programmers but could be very beneficial for software quality improvement.

If we make implicit task contexts explicit, programmers will be able to make informed decisions about using advanced modularity concepts or extended tool support.

Your Project Description

In the context of this master's project, you will create new tools that support developers in recognizing cross-cutting concerns: *Define, collect, refine*, and hence *reveal* task context information during software maintenance activities. Then, programmers should use that information to improve the code quality by recommending advanced modularity concepts (i.e., aspects) or by providing extended tool support. Project activities include

- studying related work (both literature and existing prototypes) and designing new conceptual solutions to reveal task contexts,
- creating new prototypes that implement these concepts in Squeak/Smalltalk, and
- evaluating these prototypes, for example, via small usability studies or interviews.

If you are interested, feel free to stop by and talk to us.

Contact

Software Architecture Group (http://www.hpi.uni-potsdam.de/hirschfeld)
Marcel Taeumel, Bastian Steinert, Prof. Dr. Robert Hirschfeld