

Featureous: An Integrated Approach to Location, Analysis and Modularization of Features in Java Applications

Andrzej Olszak

To remain useful, each software application needs to continuously enhance and extend the functionality (or so-called features) according to the requests of its users. Because of that, software developers need to investigate and modify the source code of software applications by treating the individual functional requests of the users as the units of work.

However, the source code of object-oriented applications, such as Java-based applications, tends not to be structured in a way that explicitly represents features perceived by the users. Instead, the internal structures of object-oriented applications focus on creating units that encompass technical responsibilities. In such a case, the parts of source code that implement a user's feature are scattered over the internal structure of an application and are tangled with one another.

The structural mismatch between the user's and the developer's perspectives on how applications are organized makes it more difficult for a developer to discover where source code corresponding to a given feature is and how it relates to other features. In consequence, the implementations of user features in source code become more difficult to understand and modify for the developers. Therefore, it is essential that the structure of source code is made to be well-aligned with the user's functional perspective on software. In other words, it is essential that features are properly modularized in the source of applications.

To overcome these problems, this thesis proposes the Featureous approach. Featureous addresses the mentioned problems in Java applications in four steps. Firstly, Featureous provides a method for locating the parts of source code of an existing application that implement a given feature of interest. Secondly, Featureous uses this information as an input for conducting so-called feature-oriented analysis, which visualizes and measures the quality of modularization of features in the source code. Thirdly, Featureous supports the process of restructuring the source code to improve the modularization of features in source code by both a manual analysis-driven method, as well as an automated one that uses the mechanisms of multi-objective optimization. Finally, Featureous proposes a heuristic for automating large-scale measurement of feature modularization in source code of Java applications.

Featureous is implemented as a software tool that is integrated with the NetBeans interactive source-code development environment for Java. This implementation is freely available at the project's website: <http://featureous.org> . Based on the created tool, Featureous was evaluated by applying it to several medium and large open-source Java applications. The gathered experiences and measurements suggest that Featureous succeeds at efficiently locating features in unfamiliar source code, at aiding comprehension and modification of features, and at improving modularization of features in Java applications.