

Reverse Engineering of Object Oriented Code

Abstract

Bc. Peter Kubov, FIT VUT, xkubov06@stud.fit.vutbr.cz

Bc. Richard Mička, FIT VUT, xmicka11@stud.fit.vutbr.cz

Maintaining up-to-date documentation of a codebase is a challenging task. During the lifetime of a program, its implementation can vary a lot. The structure can sway away from the initial design and keeping the design and documentation corresponding to the current state is a time-consuming task. Availability of up-to-date documentation can, however, play a critical role in the time needed to alter an already developed system. We will show an option where up-to-date UML documentation can be obtained from the source code of a system utilizing a reverse engineering approach.

This talk will be divided into two parts. In the first part, we will present a code analysis framework - Object Flow Graph (OFG). Such a framework is capable of tracking the lifetime and the usage of the objects in a source code. With the modification of this structure, we will derive algorithms used to recover information used in particular UML graphs.

In the second part, we will present multiple reverse engineering algorithms to obtain selected parts of the class and object diagram. We will demonstrate the process on an already developed codebase. For the class diagram, we will introduce methods of extraction of inter-class relationships in presence of weakly typed containers and interfaces. We will show how an interaction diagram showing the behavior of objects can be reconstructed by altering the object flow graph introduced in the first part of the talk.

Algorithms and reverse engineering methods in this talk are based on techniques developed by Paolo Tonella and Alessandra Potrich presented in the book Reverse Engineering of Object Oriented Code.