Automated Repair with Specifications

– SemFix –

Description

Recognizing an error in a software system may for example either be achieved by utilizing a test suite with the "right" set of test cases, or by attaching some kind of *formal specifications* to the system that is able to describe the control or data flow within the system. In the latter case, errors can be detected by ensuring that the specifications are not violated. Specifications can either be derived from the software system under consideration, or they can be specified by the user as a description (a model) of how the system *should* work.

This topic examines the tool SemFix which uses symbolic execution and constraint solvers to automatically repair bugs in software systems. It does so by deriving specifications for buggy functions based on the execution of an existing test suite. The specifications are set in such a way that all tests in the test suite pass only if the specifications are met.

References

[1] Hoang Duong Thien Nguyen, Dawei Qi, Abhik Roychoudhury, and Satish Chandra. SemFix: Program Repair via Semantic Analysis. *Proceedings - International Conference on Software Engineering*, pages 772–781, 2013.

Contacts

Simon Heiden (`heiden@informatik.hu-berlin.de`)
Software Engineering Group
Institut für Informatik
Humboldt-Universität zu Berlin