

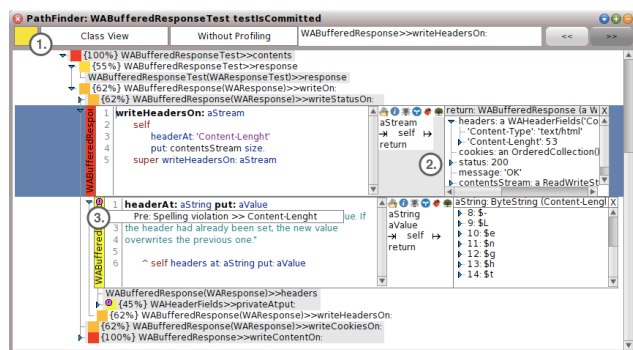
The Reflective Programmer Improved Navigation in Back-in-time Debuggers

Background

Our research efforts are based on continuous reflection on our programming practices. Developing software is often complicated by the lack of appropriate abstractions or insufficient tool support for understanding the relevant details of large software systems. We approach these and related issues by advancing programming tools, devising new modularity abstractions, and providing run-time support. Consequently, our goal is to improve the comprehension and design of complex software systems.

Description

In this master's project, you will advance our existing back-in-time debugger called PathFinder with new navigation concepts for browsing run-time behavior. Although PathFinder allows developers to explore specific test case execution in full detail, the tool misses features to answer critical questions such as “where does an object state come from?” For that reason, developers often require more time than necessary to browse the large amount of run-time data. The goal of this master’s project is to provide advanced run-time navigation concepts in order to make back-in-time debugging more comfortable. Among others, we are interested in a new query mechanism to search for specific program entities; object tracing to follow specific states through call trees; backward stepping to navigate in all directions; identification of behavioral patterns to summarize repetitive method calls; and hot recompilation and statement-level exploration to allow a full debugger experience. Project activities include:



The goal of this master’s project is to provide advanced run-time navigation concepts in order to make back-in-time debugging more comfortable. Among others, we are interested in a new query mechanism to search for specific program entities; object tracing to follow specific states through call trees; backward stepping to navigate in all directions; identification of behavioral patterns to summarize repetitive method calls; and hot recompilation and statement-level exploration to allow a full debugger experience. Project activities include:

- Studying related work (both literature and existing prototypes) and elaborate further solutions to browse program behavior
- Designing and implementing improved navigation concepts for PathFinder such as a comprehensive query mechanism, object tracing, and backwards stepping
- Conducting small usability studies to identify and resolve flaws in the programmers’ experience using your solutions on a regular basis

References

- <http://www.hpi.uni-potsdam.de/hirschfeld/publications>

Contact

Software Architecture Group

- Prof. Dr. Robert Hirschfeld
- Michael Perscheid