



Software Engineering Seminar

Dynamic Program Slicing for Fault Localization

Description

In large software systems, keeping track of control and data flow is a very hard task. To this end, *program slicing* allows the developer to concentrate on the important parts of the program under consideration by essentially *hiding* other parts of the program from the developer that are not relevant to the problem at hand. That means: parts of a program that are not influenced by a chosen set of variables, or parts that do not, themselves, influence a specific variable or a set of variables are simply not shown to the user.

Various variants of program slicing (e.g. static, dynamic, ...) have been proposed so far. The aim of this topic is to examine and discuss techniques that employ *dynamic* program slicing to help with and to improve automated fault localization.

References

- [1] E. Alves, M. Gligoric, V. Jagannath, and M. d'Amorim. Fault-localization using dynamic slicing and change impact analysis. In *2011 26th IEEE/ACM International Conference on Automated Software Engineering (ASE 2011)*, pages 520–523, Nov 2011.
- [2] Franz Wotawa. Fault localization based on dynamic slicing and hitting-set computation. In *QSIC*, pages 161–170. IEEE Computer Society, 2010.
- [3] Jifeng Xuan and Martin Monperrus. Test case purification for improving fault localization. *CoRR*, abs/1409.3176, 2014.
- [4] P. Zhang, X. Mao, Y. Lei, and Z. Zhang. Fault localization based on dynamic slicing via jslice for java programs. In *2014 IEEE 5th International Conference on Software Engineering and Service Science*, pages 565–568, June 2014.

Contacts

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