Abstract

Aspect Oriented language aims to make cross-cutting concerns clearly identifiable with special linguistic construct called aspects. In order to analyze the properties of an aspect one should consider the aspect itself and the part of the system it affects. This part is just a slice of the entire system and can be extracted by exploiting program slicing algorithms. However they will behave correctly in isolation, but when interaction changes an aspect’s behavior or disables and aspect, we will term it as aspect interference. We will propose an approach to detect aspect interference, Aspect composition are modeled by using graph production system for modelling aspect-language semantics. This graph is transformed into runtime-state representation. Combined with the production system (also with proper tool) the execution of the aspect is simulated. This simulation results in LTS(labelled transition system) that can be used to analyze verify different behavior at join points.

References

- Mehmet Aksit, Arend Rensink, and Tom Stajien, "A graph-transformation-based simulation approach for analysing aspect interference on shared join points," AOSD'09 March 2-6, 2009, Charlottesville, Virginia, USA.
- Tom Stajien, Arend Rensink, "A Graph Transformation-Based Semantics for Analysing Aspect Interference."

Index Terms

Computer Science
Languages

Keywords
Program slicing  Static slicing  Dynamic slicing  Aspect Interference  cross cutting concerns