From Structured Heaps to Encapsulated Runtime Components

Arnd Poetzsch-Heffter
poetzsch@informatik.uni-kl.de

Abstract

In OO-modeling and programming, state is structured into a set of objects. Objects can reference each other forming a graph with objects as nodes and references as edges. During the execution of object systems, the graph structure changes when objects are created or references are stored. As objects can access other objects only by following references, means to control the graph structure are very important to understand, develop, specify, or verify object systems.

In the last years, several techniques - in particular extended type systems - have been studied and developed to guarantee properties of the object graphs. We review two of these techniques, namely ownership and universe types and explain their fundamental role for semantics-based encapsulation and modular verification.