Software Architecture, Interface Specifications, and Software Verification
- Three Cheers for Modularity

Manfred Broy
Fakultät für Informatik
Technische Universität München

Abstract

In the specification and verification of hierarchical architectures of component-based software systems modularity is a key idea. We introduce and discuss the following fundamental notions:

- component and its mathematical model
- open and closed system views
- specification of the interface of components
- export/import specifications
- applications

We discuss issues of software decomposition into components and the component based specification and verification. We show how we decompose a larger software system hierarchically in subparts called its components. This decomposition is called software architecture. To specify and verify a software system in a modular way we apply the following steps:

1. We specify and prove properties about the interfaces of its subparts.
2. We prove specified properties about the complete system by referring to the proved properties of the interfaces of its subparts.

Modularity is the only way to achieve a proper separation of concerns in reasoning about large software systems. We show the shortcomings of known approaches and how to overcome them.