Software Architecture Conformance Checking

By: Razieh Rahimi
Supervisor: Dr. Ramtin Khosravi
Field: Software Engineering
Date: 3/15/2010

Abstract of MS Thesis

Software architecture is a way to deal with the complexities involved in software development processes and to achieve desired quality attributes. However, several factors cause the architecture implemented in the code to diverge from the intended designed architecture. Such erosion results in serious consequences such as complication in maintenance, change and evolution of software systems. To prevent divergence from the architecture, architecture conformance methods are adopted to check the compliance of code with the architecture.

In development of today’s enterprise applications, a number of domain-specific languages are used along with the main (general-purpose) programming language. Using domain-specific languages together with general-purpose languages results in description or implementation of the domain or application logic in different languages. Therefore, checking certain architectural rules needs inspecting artifacts of several languages and discovering relationships among them. As a result, there is a need for a conformance checking method to support multi-language source code checking.

In this research, a model-based approach is proposed to evaluate cross-language rules. Our proposed approach models the artifacts of different languages uniformly which is, in turn, independent of the languages used in the development and can be extended to support newly designed domain-specific languages and cross-language rules that cover the artifacts produced.

The proposed approach extends the expressiveness of rule description languages and makes it more effective in application to real-world software projects. Using this method continuously throughout the development and maintenance phases of software systems development significantly increases the conformance degree of code to the designed architecture.