

Adding C# Support to an Architecture Consistency Checking Tool

Context: One goal of defining the software architecture for a software system is to provide a 'blue-print' for the implementation of the system, describing which components the system consists of and between which components dependencies are allowed. Architecture consistency checking (ACC) tools help to detect architectural inconsistencies caused by introducing violating dependencies, i.e. dependencies that have been prohibited by the architecture. By reporting and visualizing violating dependencies, ACC tools help software architects and developers to implement the envisaged architecture correctly and to become aware of potential architectural problems.

Task: The task of this dissertation is to extend JITTAC, the "Just-In-Time Tool for Architecture Consistency". JITTAC is an Eclipse plugin implemented in Java and allows software architects to define the intended software architecture of a system graphically. While the software developer is editing the source code of the system, JITTAC parses the source code, extracts dependencies from it, and compares them with the specified architecture. JITTAC gives instant feedback by marking the relevant line of code, in case the developer has introduced violating dependencies, e.g. by accessing a class that the current code is now allowed to depend on.

Up till now, JITTAC is able to parse and analyze Java source code. The goal of this dissertation is to extend JITTAC to support code written in C#. This requires the candidate to realize components that parse C# code, extract the relevant dependencies, and translate them into JITTAC's internal dependency model. The implementation should be evaluated by applying the extended JITTAC tool to a case study system; the possibility to collaborate with industry during implementation and evaluation exists.

Requirements: Interested students should have a solid knowledge of object-oriented programming, very good Java programming skills and a basic understanding of the C# programming language. Experience in developing plugins for the Eclipse platform is helpful but not mandatory.

Additional resources: <https://www.youtube.com/watch?v=BNqhp40PDD4>