

Marcel Hess

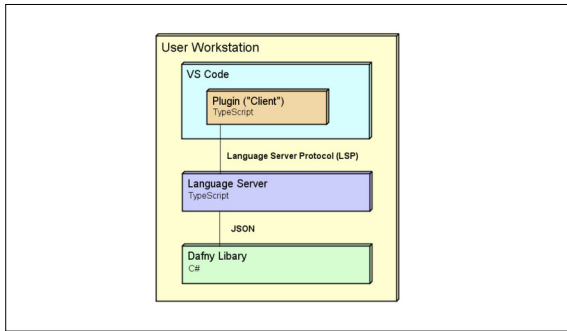


Thomas Kistler

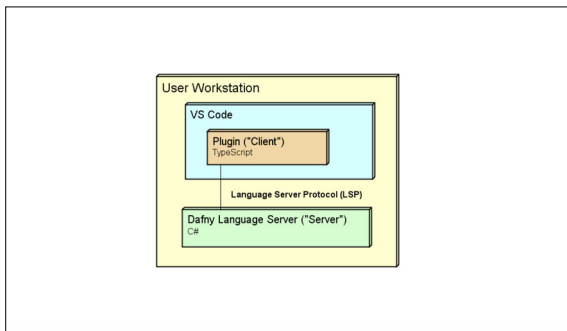
Students	Marcel Hess, Thomas Kistler
Examiner	Thomas Corbat
Subject Area	Software Engineering - Core Systems

Dafny Language Server Redesign

Reimplementation of the Dafny Language Server in C#



Deployment Diagram of the Old Solution
Own presentation



Deployment Diagram of our New Solution
Own presentation

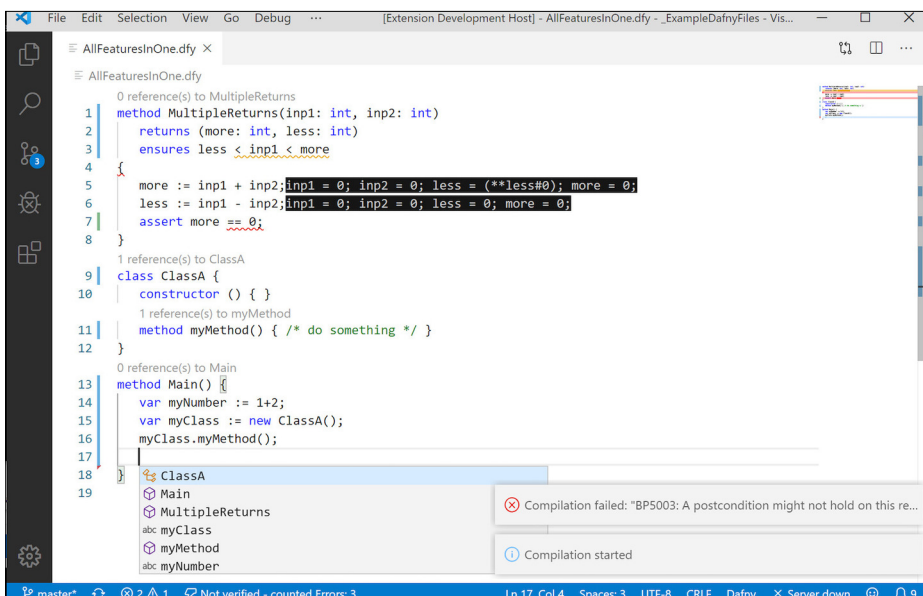
Initial Situation: Dafny is a formal programming language to prove a program's correctness with preconditions, postconditions, loop invariants and loop variants. In a preceding bachelor thesis, a plugin for Visual Studio Code has been implemented to access Dafny-specific static analysis features. For example, if Dafny cannot prove a postcondition, the code will be highlighted and a counter example is shown. Furthermore, it provides access to code compilation, auto completion suggestions and various automated refactorings. The plugin communicates with a language server, using Microsoft's language server protocol, which standardizes communication between an integrated development environment (IDE) and a language server. The language server itself accesses the Dafny library, which features the backend of the Dafny language analysis, through a proprietary JSON-interface.

Objective: The goal of this thesis was to redesign the language server to be directly integrated with the Dafny library and deprecate the proprietary JSON-based interaction. Subsequently, this facilitates implementing new language features and offering them via the language server protocol. In order to have direct access to the Dafny library the language server is reimplemented in C#.

Result: The preexisting language server could be superseded by our new, rewritten server. Since it is integrated with the Dafny library, all its functionality can be accessed directly. This renders the proprietary interface between the old language server and the Dafny library obsolete. Most features of the previous bachelor thesis are supported with the new language server. These include:

- Compilation and Code Verification
- Syntax and Error Highlighting
- Suggestions for Auto Completion
- Code Lens
- Go to Definition

With the new language server further features can be added in a convenient and uniform way. This enables integration into further IDEs, other than Visual Studio Code.



Features Postcondition Violations, Counter Example, Auto Completion, Code Lens and Compilation
Own presentation