

Abstract

With C++20, `concepts` were introduced that allow `restricting` template parameter `types`. The objective of this project was to find novel refactoring operations related to `concepts` and to implement them as part of a language server.

An analysis has been performed on the language server protocol, `concepts`, and the clangd language server. This resulted in a few potential refactoring operations, two of which have been implemented as a tweak in the clangd language server.

The first implemented refactoring enables inlining *requires* clauses into the template declaration. It reduces the amount of code and, in most cases, makes the function signature easier to read.

The second implemented refactoring allows converting explicit template declarations into their abbreviated form using *auto* parameters, thus eliminating the template header above the function.

The two implemented refactor operations have been submitted upstream as pull requests to the llvm repository and, as of the writing of this paper (December 2023), are awaiting review. Once approved and merged, these new refactoring operations will become available to anyone using the clangd language server.