

Hello Klugier,

Quote:Hello Oblivion,

I like the idea to add linter support for TheIDE, however I am not sure we should go for cppcheck. When I used it in the past, the tool had some problems like false positives. Also, the decision to relay on this particular one in the main code base is very difficult. We have very good alternatives such as clang-tidy and the true thing is that we are now basing on clang ecosystem like never before. With the usage of libclang and recent clang-format integration.

What about creating plugin system for TheIDE to allowing creation of extensions? Thanks to that there will be possible to support multiple linters. I understand that it will require a lot of work, but in the long term it is the way to go.

Klugier

Thank you very much for your comments.

CppCheck has matured nicely. Yes, it occasionally gives false positives, but so does clang-tidy. Not to mention cppcheck is famous for catching *interesting* issues where other solutions fail. The main reason why I opt for cppcheck is:

It is extremely simple to integrate and remove.

a) The ide/Linter package is almost self-contained package in the sense that there are only several Lines of code (7 lines of code in cpp files, and one line of code in ide.h: only menu entries, and config stuff) in TheIDE's codebase.

IMO that's not much of a maintenance burden. Removing it completely from the codebase if required won't take more than 15 seconds.

b) It uses it's own config file (JSON), meaning it does not mess with TheIDE's config file.

c) Implementation does not have to rely on low level, raw stuff, so future changes won't really break it unless U++ somehow gets broken.

d) It is activated only if the cppcheck executable is found.

If you look closely to the linter package, you'll notice that the main class (Linter) is actually an abstraction.

This is a stripped down version of my own linters system (I use a commercial solution besides cppcheck).

Hence the name is Linter, not CppCheck. We can definitely (re)add clang-tidy later.

As for the plugin system:

Writing a plugin system has always been a good idea, given that I intend to participate more in TheIDE's development this year, I'll likely look into it.

Best regards,

Oblivion

