Subject: Re: What do you think about this approach to making CodeEditor more user extendable?
Posted by cbpporter on Tue, 15 Dec 2015 13:16:01 GMT

View Forum Message <> Reply to Message

Things that I'm trying to fine tune by language:
- highlighting of #ifdefs
- highlighting of @region
- highlighting of nested comments
- highlighting of doxygen like comments
- highlighting things like literal constants a bit differently based on language
- highlighting ids based on other id rules
- highlight a few symbolic meta constants as non-keywords, but as literal constants, like true, for languages where true is not a keyword, but a literal constant and should be highlighted as an int.

And achieve this in a pretty general and fast way.

And generally speaking, the C like language highlighter is able to approximately, often very closely, syntax highlight a given language, but as it currently stands:
- it is not able to properly 100% highlight some inputs. The changes would be trivial to make it 100% but
- would not be compatible with C++.

That's why I proposed a structure with bool options, to make sure that highlighting is fast. What I'd need it to handle, as just one of the examples, is "0.Foo" to be highlighted as a literal int, member selection punctuation, id. And "7.0.Bar" as literal double, member selection punctuation, id. And "7.0f.Foo" as float and so on. For all those pure OOP languages.