

---

Subject: Re: Very first impressions and.... [FEATURE REQUESTS]

Posted by [mirek](#) on Wed, 12 Sep 2007 19:52:16 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

[quote title=mdelfede wrote on Wed, 12 September 2007 14:59]luzr wrote on Wed, 12 September 2007 18:40

file from preprocessor kept in sync with the file I'm workin on....

How do you want to keep it in sync?

Note that rescanning after #define is typed is not enough.

Any change in file changes potential preprocessor output. This is not even related to macros, even if there are no macros, any change in file changes preprocessor output.

Quote:

I've not looked at your parser code, but I guess it does a first scan on file when ide opens, and then it keeps scanning only the changes you make to file, am I wrong ?

Yes, you are wrong

All project files are scanned at ide startup(\*). Then modified files are scanned before they are leaved or if the information is needed.

(\*) in fact, all information is cached, so in reality only changed files are scanned at startup, unchanged file information is just loaded.

Quote:

So, at ide opening you add the preprocessor pass before the full scan pass. Then, when you work on file, there are 2 possibilities :

1- you make changes not related to # preprocessing directives, so your code work as usual.

Any change in file is related to preprocessor.

That is the problem you seem to miss (or maybe I do not understand your method).

Quote:

Quote:

The preprocessor is the simple part... The hard part is caching logic.

Of course, but that is already done. It should only a matter of keeping a second buffer (aka symbol table in memory) and switch to it when the long job is done. It should be not too difficult to insert in current code.

If only things would be so simple

Note:

What you suggest might partially work, if preprocessor would be capable of exporting all defined macros together with preprocessor output. In that case, you could split the processing to processing of headers, which would yield two files (preprocessed files and set of macros) and the main file, then you could e.g. include only macros for processing of main file, which would be much faster. But hey, this is what I call "partial preprocessing".

Not that there is another problem: You are not only interested in information contained in the .cpp file, but also in things defined in headers...

Mirek

---