Subject: C++ modules support is coming
Posted by Mindtraveller on Sun, 27 Sep 2015 10:58:25 GMT
View Forum Message <> Reply to Message

Well, it looks like an epic change of C++ is to come. After Clang, the latest Miscrosoft's C++ 2015 Update 1 will have modules support.
What will it give us? Better build times, better portability, easier attach of libraries.
Here's description of what C++ modules are:
https://github.com/isocpp/CppCoreGuidelines/blob/master/talk
s/Large-Scale-C%2B%2B-With-Modules.pdf
The proposal on possible implementation from Microsoft:
http://www.open-std.org/JTC1/SC22/WG21/docs/papers/2014/n421 4.pdf
The same from Clang:
http://clang.llvm.org/docs/Modules.html

A small descriprive article in Russian:
http://habrahabr.ru/company/infopulse/blog/267781/

Subject: Re: C++ modules support is coming
Posted by rxantos on Thu, 30 Mar 2017 09:04:27 GMT
View Forum Message <> Reply to Message

I don't understand.
What advantage do they give over static libraries?

Do they allow separation of implementation and declaration of templates?

If yes, I can see how they are useful. If no, then what exactly do they bring new?

Subject: Re: C++ modules support is coming
Posted by Klugier on Thu, 30 Mar 2017 21:01:54 GMT
View Forum Message <> Reply to Message

Hello,

The modules concept are easy. Modules will drop support for current double file system (header and implementation) and unify it in one file like it is in other languages like Java or other modern languages like golang.

The code will look like this (Upp exampele)

// Core.cpp:

```
module Upp.Core;

class File
{
public:
    // File methods...
};



// MyProject.cpp

import Upp.Core; // No include, however #include "" will be probably still possible for backwards compatibility.

CONSOLE_APP_MAIN // Macro handling is big mystery for me?
{
   File file;
   // Do something with the file...
}
```

The above example is only concept and it base only on my feature imagination. Of course the final implementation can be different, so you could study standard documentation to obtain more knowledge.

Sincerely,
Klugier
