C++20 feature: Modules

Adam Graliński

C++ FFFE, April 2021

Why modules?

Pay only for what is being used approach is **awesome**

- holds for runtime
- kind of does not hold for compilation time
  - long headers with lots of template instantiating
    - such as Boost
    - such as STL

Why modules?

Problem: Long compilation time penalty
Solution:

version	*solution*	*pros*	*cons*
C++17 and before	**precompiled headers**	some*	lots*
C++20	**modules**	lots	still not widely supported

How to use?

https://en.cppreference.com/w/cpp/language/modules

```
// helloworld.cpp
export module helloworld;  // module declaration
import <iostream>;         // import declaration

export void hello() {      // export declaration
    std::cout << "Hello world!\n";
}

// main.cpp
import helloworld;  // import declaration

int main() {
    hello();
}
```

Compiler support

You'll need a modern compiler with support for modules.

(note: at the time of writing this, modules support is partial/incomplete for every existing compiler. However, it is complete enough to compile the example code.)

- `-std=c++20` or `-std=c++2a`
- **clang**: works out of the box since 11.0. See P1766R1.
- **gcc**: works.
  - P1766R1 is **not** supported, but many others are
  - requires g++ version 11
  - requires extra flag: `-fmodules-ts`
- **MSVC**: basic support is claimed
  - VS 2019 with Modules (16.8 preview 3) or later required
  - I was unable to verify this :-)

How to compile?

1. Precompile the module code to binary module interface

```
clang++ -std=c++2a -c helloworld.cpp -Xclang
        -emit-module-interface -o helloworld.pcm
```

2. Compile c++ sources and link them

```
clang++ -std=c++2a -fprebuilt-module-path=. main.cpp helloworld.cpp
```

Taken from Arthur O'Dwyer 's excellent article.

Granular control over exports

```
export module MyModule;
void Foo() { /* do important stuff */ }
export void Bar() {
  Foo();
}

import MyModule;

int main() {
  Bar(); // OK
  // Foo(); // won't compile!
}
```

What next?

https://eel.is/c++draft/module
https://blog.ecosta.dev/en/tech/explaining-cpp20-modules

Thank you!