Create namespaces for modules without wrapping files in namespace blocks

Document #:	PXXXXRX
Date:	2022-09-09
Project:	Programming Language C++
Audience:	Evolution

Reply-to:	Maximilian Downey Twiss <creatorsmithmdt@gmail.com>

1 Introduction

This paper proposes to allow the creation of namespaces for modules without wrapping files in namespace-blocks through specifying a namespace-name in a module-declaration.

2 Motivation and Scope

While headers allow one to relatively easily use namespaces without wrapping files in namespace blocks, modules have no such equivalent. Using headers, one can wrapping the header files in namespace blocks and prepending the namespace onto the functions in the implementation file, although this is against good style.

Through this change, this paper creates a simple, compact way to use namespaces with modules, as opposed to wrapping implementation files inside namespace blocks.

(There is not currently a reference implementation, although one would be trivial to create, as a non-standard extension to the core language is a tough sell for compilers.)

3 Wording

Relative to N4910.

3.1 #[module]

Change the grammar before paragraph 1:

```
  module-declaration:
    export-keyword(opt) module-keyword module-name module-partition(opt)
-    attribute-specifier-seq(opt);
+    namespace-name(opt) attribute-specifier-seq(opt);
  module-name:
    module-name-qualifier(opt) identifier
```

(The re-adjusting of numbers after the introduction of a new paragraph and note was not included for diff conciseness, but is implied) Insert new paragraph after 3:

```
+ A module-declaration with a namespace-name has behaviour equal to a module containing
+ a namespace-definition with a namespace-body enclosing all exported functions.

+ [Note 2: Only the primary module interface unit of any given module
+ can contain a namespace-name. -- end note]
```