

C++ Mapping for Modules

A Slice [module](#) maps to a C++ namespace. The mapping preserves the nesting of the Slice definitions. For example:

Slice

```
module M1 {  
    module M2 {  
        // ...  
    };  
    // ...  
};  
  
// ...  
  
module M1 {    // Reopen M1  
    // ...  
};
```

This definition maps to the corresponding C++ definition:

C++

```
namespace M1 {  
    namespace M2 {  
        // ...  
    }  
    // ...  
}  
  
// ...  
  
namespace M1 { // Reopen M1  
    // ...  
}
```

If a Slice module is reopened, the corresponding C++ namespace is reopened as well.

See Also

- [Modules](#)
- [C++ Mapping for Identifiers](#)
- [C++ Mapping for Built-In Types](#)
- [C++ Mapping for Enumerations](#)
- [C++ Mapping for Structures](#)
- [C++ Mapping for Sequences](#)
- [C++ Mapping for Dictionaries](#)
- [C++ Mapping for Constants](#)
- [C++ Mapping for Exceptions](#)