## C++ Mapping for Identifiers

A Slice identifier maps to an identical C++ identifier. For example, the Slice identifier <code>Clock</code> becomes the C++ identifier <code>Clock</code>. There is one exception to this rule: if a Slice identifier is the same as a C++ keyword, the corresponding C++ identifier is prefixed with <code>\_cpp\_</code>. For example, the Slice identifier while is mapped as <code>\_cpp\_while</code>.

A single Slice identifier often results in several C++ identifiers. For example, for a Slice interface named Foo, the generated C++ code uses the identifiers Foo and FooPrx (among others). If the interface has the name while, the generated identifiers are \_cpp\_while and whilePrx (not \_cp p\_whilePrx), that is, the prefix is applied only to those generated identifiers that actually require it.



You should try to avoid such identifiers as much as possible.

## See Also

- Lexical Rules
- C++ Mapping for Modules
- 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