

Objective-C Mapping for Built-In Types

The Slice built-in types are mapped to Objective-C types as shown below.

Slice	Objective-C
bool	BOOL
byte	ICEByte
short	ICEShort
int	ICEInt
long	ICELong
float	ICEFloat
double	ICEDouble
string	NSString or NSMutableString

Slice `bool` maps to Objective-C `BOOL`. The remaining integral and floating-point types map to Objective-C type definitions instead of native types. This allows the Ice run time to provide a definition as appropriate for each target architecture. (For example, `ICELong` might be defined as `long` on one architecture and as `long long` on another.)

Note that `ICEByte` is a typedef for `unsigned char`. This guarantees that byte values are always in the range 0..255, and it ensures that right-shifting an `ICEByte` does not cause sign-extension.

Whether a Slice string maps to `NSString` or `NSMutableString` depends on the context. `NSMutableString` is used in some cases for operation parameters; otherwise, if a string is a data member of a Slice structure, class, or exception, it maps to `NSString`. (We will discuss these differences in more detail as we cover the mapping of the relevant Slice language features.)

See Also

- [Objective-C Mapping for Modules](#)
- [Objective-C Mapping for Identifiers](#)
- [Objective-C Mapping for Enumerations](#)
- [Objective-C Mapping for Structures](#)
- [Objective-C Mapping for Sequences](#)
- [Objective-C Mapping for Dictionaries](#)
- [Objective-C Mapping for Constants](#)
- [Objective-C Mapping for Exceptions](#)
- [Objective-C Mapping for Interfaces](#)