C++ Mapping for Dictionaries

Here is the definition of our `EmployeeMap` once more:

Slice

```
dictionary<long, Employee> EmployeeMap;
```

The following code is generated for this definition:

C++

```
typedef std::map<Ice::Long, Employee> EmployeeMap;
```

Again, there are no surprises here: a Slice dictionary simply maps to an STL `map`. As a result, you can use the dictionary like any other STL `map`, for example:

C++

```
EmployeeMap em;
Employee e;

e.number = 42;
e.firstName = "Stan";
e.lastName = "Lippman";
em[e.number] = e;

e.number = 77;
e.firstName = "Herb";
e.lastName = "Sutter";
em[e.number] = e;
```

All the usual STL iterators and algorithms work with this map just as well as with any other STL container.

See Also

- Dictionaries
- C++ Mapping for Enumerations
- C++ Mapping for Structures
- C++ Mapping for Sequences

Copyright © 2017, ZeroC, Inc.