Chapter 71. Boost.Swap

If you use many Boost libraries and also use `std::swap()` to swap data, consider using `boost::swap()` as an alternative. `boost::swap()` is provided by Boost.Swap and is defined in `boost/swap.hpp`.

Example 71.1. Using `boost::swap()`

```
#include <boost/swap.hpp>
#include <boost/array.hpp>
#include <iostream>

int main()
{
  char c1 = 'a';
  char c2 = 'b';

  boost::swap(c1, c2);

  std::cout << c1 << c2 << '\n';

  boost::array<int, 1> a1{{1}};
  boost::array<int, 1> a2{{2}};

  boost::swap(a1, a2);

  std::cout << a1[0] << a2[0] << '\n';
}
```

`boost::swap()` does nothing different from `std::swap()`. However, because many Boost libraries offer specializations for swapping data that are defined in the namespace `boost`, `boost::swap()` can take advantage of them. In Example 71.1, `boost::swap()` accesses `std::swap()` to swap the values of the two `char` variables and uses the optimized function `boost::swap()` from Boost.Array to swap data in the arrays.

Example 71.1 writes `ba` and `21` to the standard output stream.