
Chapter 12. Special Containers

The C++ standard library provides not only the containers for the STL framework but also some containers that fit some special needs and provide simple, almost self-explanatory, interfaces. You can group these containers into either the so-called *container adapters*, which adapt standard STL containers to fit special needs, or a bitset, which is a containers for bits or Boolean values.

There are three standard container adapters: stacks, queues, and priority queues. In priority queues, the elements are sorted automatically according to a sorting criterion. Thus, the “next” element of a priority queue is the element with the “highest” value.

A bitset is a bitfield with an arbitrary but fixed number of bits. Note that the C++ standard library also provides a special container with a variable size for Boolean values: `vector<bool>`. It is described in Section 7.3.6, page 281.

Recent Changes with C++11

C++98 specified almost all features of the container adapters. Here is a list of the most important features added with C++11:

- Container adapters now provide type definitions for `reference` and `const_reference` (see Section 12.4.1, page 645).
- Container adapters now support move semantics and rvalue references:
  - `push()` provides move semantics now (see Section 12.1.2, page 634, and Section 12.4.4, page 647).
  - Initial containers can be moved now (see Section 12.4.2, page 646).
- Container adapters provide the `emplace()` feature, which internally creates a new element initialized by the passed arguments (see Section 12.1.2, page 634, and Section 12.4.4, page 647).
- Container adapters now provide `swap()` (see Section 12.4.4, page 649).
- Constructor adapters now allow you to pass a specific allocator to their constructors (see Section 12.4.2, page 646).