

C++ Standard Template Library by Ranges

Norbert Pataki

Dept. of Programming Languages and Compilers,
Fac. of Informatics, Eötvös Loránd University, Budapest
e-mail: patakino@elte.hu

Abstract

C++ Standard Template Library (STL) was developed by *generic programming* approach. In this way some containers defined as class templates and many algorithms can be implemented as function templates. Furthermore, algorithms are implemented in a container-independent way, so one can use them with different containers. Iterators bridge the gap between containers and algorithms. In this approach ranges are passed as two independent parameters. Nevertheless, STL was designed to be extensible.

Iterators introduced many different problems. First, hard to write hand-crafted iterators. Some people argue for the range types which can be a superior abstraction. Ranges can be aggregated in a better way. They have better checking abilities.

In this paper we present an overview about the differences between ranges and iterators. We examine a set of algorithms with ranges. Creating new range types is also considered. Few libraries have already implemented range types, and they are compared.

Keywords: STL, iterators, ranges

MSC: 68N19 Other programming techniques

Norbert Pataki

Pázmány Péter sétány 1/c., H-1117 Budapest, Hungary