

3.3 Release Notes: libg++

This file contains release notes for the initial release with NEXTSTEP of the GNU C++ class library (libg++ version 2.3).

Example

A simple test program would be:

```
#include <stream.h>

main()
{
    cout << "Hello World!\n";
}
```

Compile this program with the following command:

```
cc++ test.cc -o tst -lg++
```

Running **tst** would output:

```
Hello World!
```

Using **genclass**

The **genclass** script generates **libg++** class definitions for user consumption. Typically, to generate a linked list of strings, one would use:

```
genclass String ref defs
```

to generate **String.defs.h**, and

```
genclass String ref SLList
```

to generate **String.SLList.h** and **String.SLList.cc**, which provide the definition and implementation of **StringSLList** class.

Notes Specific to Release 3.2

The compiler has been updated to look for included header files in **/NextDeveloper/Headers/g++** when the compiler is invoked on a C++ file. This eliminates the hassle of using the **-I** option every time a C++ source file is compiled. The original GNU source has been compiled and a subset of the header files have been massaged for NeXT compatibility. Those specific header files, located in **/NextDeveloper/Headers/g++**, are:

fcntl.h	regex.h	strings.h
libc.h	stdio.h	time.h
math.h	stdlib.g	unistd.h
memory.h	string.h	values.h

sys/socket.h sys/time.h sys/times.h

Note that the last three headers, located in the g++ include directory, cover the functionality of the ^astandard^o headers included by the same name; that is, **#include€<sys/socket.h>** in a C++ file would use the file in **/NextDeveloper/Headers/g++** rather than the standard UNIX file.

The stream manipulators declared in the file **/NextDeveloper/Headers/g++/iomanip.h** are not supported due to the lack of support for templates in **gcc** in the current version of NEXTSTEP.