Q: What version of the C++ compiler is used in NEXTSTEP Release 3.1?

A: The Release 3.1 of the Objective C/C++ compiler is based on version 2.2.2 of the GNU compiler. The GNU compiler has been extended to recognize Objective C constructs within C++ source files. Note that the language g++ accepts is very similar in semantics and syntax, but not identical to that defined by AT&T.   Both products are tracking the ongoing ANSI X3J16 standardization work and are converging on the C++ draft standard, due in late 1994.

Q: What development tools are provided with your C++ compiler?   Can I use GDB?

A: Yes, GDB, nm, and gprof can be used with C++ source code.   With Release 3.1,   if your code is compiled with the   **-gg** switch, GDB will be able to show   C++ methods with more readable names. The user can then insert breakpoints using these names.   The **-gg** switch is an abbreviation for   the **-ggdb** flag to load the gdb debug symbols.

Q: Do you provide any C++ libraries?

A: libg++ for NEXTSTEP Developer is included in Release 3.2, on the CD-ROM.   NeXT does not distribute nor support any previous versions of libg++ for any previous releases of NEXTSTEP Developer.

Q: How do I integrate C++ and Objective C code?

A: Release 3.1 provides on-line documentation on how to write mixed code (Objective C/C++). Here is

the list of the on-line reference material:

- Chapter 5 of *Object-Oriented Programming and Objective C* (/NextLibrary/Documentation/NextDev/Concepts/ObjectiveC/5_Programming) discusses how to use C++ with Objective C.
- Chapter 11 of *Development Tools and Techniques* (/NextLibrary/Documentation/NextDev/DevTools/11_Compiler) discusses C++ compiler options.
- The C++ release notes for Release 3.1 (/NextLibrary/Documentation/NextDev/ReleaseNotes/C++.rtf).

QA889

Valid for 3.1