Document No: X3J16/97-0081
WG21/N1119
Date: September 29, 1997
Reply to: Roland Hartinger
Roland.Hartinger@mch.sni.de

C Library Names and namespace std

(1) Introduction

The German DIN C++ working group is not satisfied with the partial solution of the issue in the current WP, described in X3J16/97-0067 (17.3.3.1.3 External Linkage, 17.3.3.1.4 Types).

The main reason is that implementations must do rather tricky stuff inside the <c*> header files especially if they don't have control of the original C headers. Most of the problems are well worked out in the document X3J16 /97-0049 and are therefore not repeated here. Those difficulties in implementation can reduce the quality of implementations and pull the burden to the users side which is not acceptable.

This proposal is based on the idea that <c*> headers give the users access to ISO C names in a standard conforming way via namespace std. It also supports access to non ISO C names on global namespace in a compatible manner for those implementations which have to support old stuff.

(2) Proposal

(I) Allow implementations to bring ISO C names explicitly in namespace std by an using declaration which in parallel exist in the global namespace, for example:

```
// cstdio
#include <C/stdio.h>    // C Library header
namespace std {
    // ...
    using ::errno;
    // ...
    using ::size_t;
    // ...
    using ::printf;
    // ...
    // and all other ISO C function/objects/types from <C/stdio.h>
}
```

So, a user can write his C++ programs in a Standard conforming way, like:

```
#include <cstdio>

int main() {
    std::printf("Hello");
}
```

(II) For the purpose of being compatible to existing programs which use UNIX or XOPEN interfaces, let the users also succeed if they trying to use non ISO C declarations from global scope,
for example:

```
#include <cstdio>
    // ...
```

```
    printf("abc");  // Ok, ISO C taken from global scope
    // ...
    popen("xyz", "abc");  // Ok, non ISO C function taken from global scope
}
```

(3) Changes in the Working Paper

17.3.1.2 Headers [lib.headers]

Change last sentence of paragraph 4 from

> **4 ... In the C++ Standard Library, however, the declarations and definitions (except for names which are defined as macros in C) are within namespace scope (3.3.5) of the namespace std.**

to

> **4 ... In the C++ Standard Library, however, the declarations and definitions (except for names which are defined as macros in C) are within namespace scope (3.3.5) of the namespace std and the global scope.**