The Windows sockets (winsock) API is mostly compatible with the Berkeley sockets (BSD sockets) API. This document addresses the major differences and describes how to write applications that are portable between winsock and BSD sockets. All winsock code has been tested under Visual Studio 2010 and MinGW. All BSD socket code has been tested under Linux and Cygwin.

**Header Files**

The winsock API uses a different set of header files than the BSD socket API. Most winsock routines are declared in `winsock2.h` or `ws2tcpip.h`. Portable code can use the C/C++ preprocessor to test for predefined macros and include the appropriate header files:

```c
#if defined(unix)       // Cygwin and Linux
    #include <sys/socket.h>
    #include <netdb.h>
    #include <netinet/in.h>
    #include <netinet/ip.h>
    #include <arpa/inet.h>
#elif defined(_WIN32)   // Visual Studio and MinGW
    #include <winsock2.h>
    #include <ws2tcpip.h>
#endif
```

**Startup/Cleanup Code**

The winsock API requires that an initialization routine be called before any of the winsock functions can be used. BSD sockets do not require initialization. We can test for a preprocessor macro to ensure that the startup code is called when programming in an environment that uses the winsock API:

```c
#if defined(_WIN32)       // Visual Studio and MinGW
    int iResult;
    WSADATA wsaData;
    // Initialize Winsock
    iResult = WSAStartup(MAKEWORD(2,2), &wsaData);
    if (iResult != 0) {
        std::cerr << "WSAStartup failed: " << iResult << "\n";
    }
#endif
```

Similarly, a winsock cleanup routine should be called after an application has finished using all socket routines:

```c
#if defined(_WIN32)
    WSACleanup();
#endif
```

**Linking**

Linking to the BSD socket routines is done automatically under Linux and Cygwin. To link to the winsock routines under Visual Studio or MinGW you must link to the `ws2_32` library. This is not done automatically. Under the project Properties window add `ws2_32.lib` to the Additional Dependencies under Configuration Properties→Linker→Input area.