Why do I get the error message "Native call not found" and how do I fix the problem?

The error message "Native call not found" means that the COBOL program has called a C language (native) function that has the same name as the native library (.dll or .so), but the function does not exist.

For example, the first statement in the following code loads a native library named foo.dll on Windows or libfoo.so on Linux/UNIX if a native library with that name exists in the Windows search path (e.g. PATH) or in the Linux/UNIX library search path (e.g. LD_LIBRARY_PATH). The second statement attempts to call a function named foo() in that native library:

```
      CALL "foo".              | load foo.dll or libfoo.so      CALL "foo" using param.   | call foo()
```

If there is no function named foo() then the isCOBOL framework will report:

Native call not found

Note that if there is no native library in the search path with that name (e.g. foo.dll or libfoo.so), then the isCOBOL framework will assume that FOO is the name of a COBOL program and will attempt to call it. If the COBOL program is not found then the isCOBOL framework will report:

CALL not found: FOO (java.lang.ClassNotFoundException: FOO)

Also note that the statement:

```
      CALL "foo" using param.
```

loads a native library named foo.dll or libfoo.so if one exists in the library search path AND attempts to call a function named foo() as well.

So if the framework succeeds in loading a native library with that name, but cannot find a function named foo() then it will report:

Native call not found

If you get the error "Native call not found" and you do not mean to call a native library (i.e. you are trying to call a COBOL program) then see if there is a DLL or shared object library with the same name as the COBOL program in the library search path. For example, on Windows the following statement produces the "Native call not found" error because the JRE includes a native library named splashscreen.dll on Windows:

```
        CALL "splashscreen" using param.
```

To fix this problem either rename your COBOL subprogram or make sure there is no DLL in the PATH with the name splashscreen.dll.

For example, you could rename your COBOL subprogram to "mysplashscreen".

Or you could remove the JDK bin (and/or JRE bin) folder from PATH and then execute the program using the absolute path to the Java executable. For example,

```
set PATH="C:\Program Files\Java\jdk1.6.0_21\bin\java.exe" MAIN
```

Or you could rename splashscreen.dll to splashscreen.dll.backup