OpenVMS Technical Journal V12

Calling OpenVMS native routines from Java

Tim E. Sneddon

Calling OpenVMS native routines from Java 1
Overview 2
The Java application 2
Setting up the run-time library 2
Building the Java layer 3
Wrapping it all up in a Java jacket 5
Conclusion 7
For more information 9

© Copyright 2009 Hewlett-Packard Development Company, L.P

Overview

Developing a Java application that calls native routines can be tedious. Along with the native routines and the Java application, there is also the need to develop a Java Native Interface (JNI) layer. This article presents a collection of tools and libraries that remove the need to develop the JNI layer—allowing Java to call native routines and manipulate data structures directly.

The "legacy" application

For this tutorial, a simple native application written in PL/I has been selected as the target of an "upgrade". The application is an address/phone book program, called PHONE[1] (not to be confused with the OpenVMS utility of the same name). It uses the SMG$ API to draw the screen and an RMS index file to store each phone book record. Figure 1 shows the record entry screen.

Figure 1. PHONE data entry screen

The intention is to be able to access the phone book data file from Java so a new GUI can be built. By using Java, the options for future development are expanded to allow development of either a Swing-based (GUI) interface or a web-based front end (using Tomcat). However, the most important requirement is to allow the original application to continue being used with little or no change.

Setting up the run-time library

In order for Java to call a native routine it must be in a shareable image. The first step is to identify the modules that are necessary for accessing the data file. In this case, it is simple: the DATABASE.PLI module contains all routines related to file access. From this module, the information about its entry points are as follows:

OPEN_PHONEBOOK — This routine opens the phone book datafile.
CLOSE_PHONEBOOK — This routine closes the phone book datafile

[1] For those with a PL/I compiler this program can be found in the examples directory for the Kednos VAX and Alpha PL/I compiler.