Java* Managed Run Time Environment: Frequently Asked Questions

Topics:

Compatibility

1. Supported platforms and system requirements

What Java Virtual Machines (JVMs) does the VTune™ Performance Analyzer 7.1 support? Are there any restrictions?

- On IA-32 systems running a Microsoft* operating system:
  - BEA* WebLogic* JRockit* 8.0/8.1
  - IBM* JDK* 1.3.1
  - Sun* JDK* 1.4.1/1.4.2
  - Microsoft* JView*
- On Itanium®-based systems running a Microsoft* operating system:
  - IBM* JDK 1.3.1
  - BEA* WebLogic* JRockit* 8.0/8.1
- On IA-32 Linux* systems:
  - BEA* WebLogic* JRockit* 7.0SP2/8.0/8.1SP1

- IBM* JDK* 1.3.1/1.4.0
- Sun* JDK* 1.4.1/1.4.2

NOTES: Profiling Java programs invoked via a script is not supported on IA-32 Linux systems.

It appears that IBM* JDK* 1.3.1 only works on SuSe* 9.0 with the environment variable *LD_ASSUME_KERNEL* set to 2.2.5 — with and without profiling. However, check with JVM vendors on which platform each JVM version is supported.

Please read the next item in this FAQ for specific issues with the versions of Linux* that introduced Native POSIX* Thread Library support.

2. Java profiling on Linux* versions enabled with the Native POSIX Thread Library*

How can I profile Java programs on an NPTL enabled Linux* operating system?

Java profiling using the VTune analyzer has been validated to function on Red Hat* 9 (with the latest patches available from Red Hat), Red Hat* Fedora core 1, and Red Hat* Enterprise Linux* 3.0 with the following restrictions:

- Red Hat* 9 (with updated errata packages) and Red Hat* Fedora core 1
  - BEA*/JRockit* — Set the environment variable *LD_ASSUME_KERNEL* to **2.4.1**.
  - IBM* JDK* — Set the environment variable *LD_ASSUME_KERNEL* to **2.2.5**.
  - Sun* JDK* — No restrictions.
- Red Hat* Enterprise Linux* 3.0
  - BEA*/JRockit* — Set the environment variable *LD_ASSUME_KERNEL* to **2.4.1**.
  - IBM* JDK* — This does not seem to execute successfully ... even without profiling.
  - Sun* JDK* — No restrictions.

NOTE: Please check with the JVM vendor on support for specific Linux* operating systems. This section summarizes findings from profiling applications using the VTune analyzer and may not necessarily agree with recommendations made by JVM vendors for your version of Linux*.

Usage

3. Viewing call graph of Java program without the native DLLs

How do I get a call graph of my Java program only, without all the native DLLs that execute under the profiled process?

Select the **Java profiling** option in the call graph wizard. Specify all the relevant information about your application in the various wizard steps. The last step enables you to choose between call graph with Java classes only, and a complete call graph: Java and native calls. The default is **Java classes only**. Select **Java classes only**.

4. Viewing call graph of Java program including the native DLLs

How do I get a call graph of my Java program including all the native DLLs that execute under the profiled process?

Select the **Java profiling** option in the call graph wizard. Specify all the relevant information about your application in the various wizard steps. The last step enables you to choose between call graph with Java classes only, and a complete call graph that includes both Java and native calls. The default is **Java* classes only**. Select **All Java* classes and all native modules** to get both Java and native code call graph data. This feature is not supported when profiling on a remote Linux* system.

5. Viewing the Java command line with configuration information

How can I see the Java command line that the VTune™ Performance Analyzer invokes after I enter all the configuration information via the wizard?

After entering the application details in the call graph or sampling wizard, you can view the command line that the analyzer has configuration based on your information. You also have the option to edit the command line. This occurs within the wizard, prior to completing the creation of the activity.

6. Specifying a JAR file

How do I specify a JAR file for the Java application?

The call graph and sampling wizards enable you to specify a JAR file as the application to execute. The JAR file must contain a Main function.

7. Specifying a Java package

How do I specify a Java package?

The call graph and sampling wizards enable you to specify a package name as the application to execute.

8. Specifying the *CLASSPATH*

How do I specify the CLASSPATH that is associated with my Java application?

The call graph and sampling wizards enable you to specify the *CLASSPATH* for your application.

9. Launching outside of the VTune analyzer

Can I profile a Java application that I launch outside of the VTune analyzer?

Yes. Please refer to the VTune analyzer online Help for more details.

10. Java applet executed within the Internet Explorer

How can I profile a Java applet that executes within the Internet Explorer?

Currently, the VTune analyzer does not support applet profiling within the Internet Explorer. You can use the *appletviewer* as the launched JVM to profile applets.

11. Invoking your application via a script

How can I profile a Java program invoked via a script?

The call graph and sampling wizards enable you to specify a script that invokes your application. You should add the **–Xrunjavaperf** option on IA-32 systems and the **–Xrunjavaperf64** option on Itanium* systems to the **jvm** command line. This feature is not supported when profiling on a remote Linux* system.

Tips

12. Missing functions in sampling or call graph views

Why don't I see specific functions in Sampling or Call graph views — I'm sure these functions consume a lot of CPU cycles in my application?

Some Java JIT compilers inline functions into their calling functions. As a result, the CPU consumed by the callees is associated with the callers, and there is no explicit mention of the callees.

13. Samples associated with only last line of functions

When I drill down to the Java source file to see sampling data, all samples are associated with the last line of the functions.

This issue has been reported with the IBM* JDK* 1.3 on Microsoft* Windows* platforms. Try using another JDK for viewing this type of information.

14. Cannot terminate application

At the end of a sampling Activity, a message pops up saying that the application can't be terminated. What should I do?

There are two solutions:

- Execute the following command once:

```
vtl global-options kill-app-without-prompt=yes
```

- Click **End Now** and **OK** to terminate the program.

15. Java application fails to execute during remote profiling

Why does my Java application fail to execute when doing remote profiling and setting the application CLASSPATH via the wizard?

Any *CLASSPATH* environment variable component required for your remote application to execute should be added manually. Using the wizard, you can add it when specifying the *CLASSPATH* or by editing the complete invocation line displayed in the command line (one of the steps of the Java wizard).

16. Application fails to execute when invoked from command line

Why does my Java application fail to execute when invoked from the VTune analyzer command line?

You might have set the **–cp** JVM command-line option with an incomplete list of *CLASSPATH* arguments.

17. Error message regarding unknown pthread library

When profiling with BEA JRockit, I get the following error message:

```
ERROR: The pthread library is unknown. Are you running a supported
Linux distribution? (Segment-register gs appears to be unused.)
```

This error message may appear only during remote data collection on a machine running some version of Linux*. To resolve this problem, please add **/lib/i686** to the beginning of the

LD_LIBRARY_PATH environment variable of the command shell where you invoke vtserver, or set it to **/lib/i686** if it does not exist.

18. Profiling 32-bit JVMs on Itanium®-based systems

When profiling on Itanium®-based systems running a Microsoft operating system, the profiling session fails immediately claiming incompatible file architecture.

This error message may appear due to configuring the VTune analyzer to launch an IA-32 JVM on an Itanium®-based system. The VTune analyzer does not support this type of profiling.

19. Semaphore removal failure on Linux* RDC

What should I do when the following message shows up on the vtserver's command window during Linux* Remote data Collection?

```
Failed to delete semaphore XXXXXXX: Operation not permitted
```

(Where XXXXXXX is the semaphore ID)

Try restarting **`vtserver`** to automatically remove the semaphore.

To remove the semaphore manually:

1. Log on as root.
2. Verify that the semaphore exists by executing the command:

   ```
   ipcs
   ```

   The semaphore should appear under the Semaphore Arrays list. It can be recognized by the *XXXXXXXX* id).
3. To remove it, execute the command:

   ```
   ipcrm sem XXXXXXXX
   ```

   where *XXXXXXX* is the semaphore ID.
4. Verify that it was removed by re-executing the **ipcs** command again.