

# Verifier Extension

## Mission

Harmony verifier extension is used to recalculate stack maps of a class after it is instrumented by TPTP.

## TPTP Java Profiler Development Environment

The environment is used to debug the extension. For example, it allows writing a class file with a modified stack table down.

### Requirements

1. MS Windows, Visual C++ 6.0.
2. Java JDK 1.5, 1.6.
3. Environment variable: JAVA\_HOME=<full path to JDK 1.5 directory>.
4. JDK header files (jni.h, jvmti.h, and jvmpi.h) are expected to be available under %JAVA\_HOME%\include.

### Getting the Verifier Extension

```
cd <Harmony root>
svn co http://svn.apache.org/repos/asf/harmony/enhanced/trunk
cd working_vm
svn checkout build.xml vm/vmcore/src/verifier-3363/ vm/vmcore/include/ make/ vm/include/
```

### Compiling the Verifier Extension

1. Run a command prompt. Include VC6 and ant directories into PATH, INCLUDE, and LIB environment variables.
2. From <Harmony root>\working\_vm invoke ant clean-verifier-ext verifier-ext.
3. Copy the resulting library <Harmony root>\working\_vm\build\windows\_x86\_msvc\_{debug|release}\deploy\lib\verifier-ext.lib to <TPTP root>\org.apache.harmony\_vmcore\_verifier\lib\windows\\_{debug|release}\IA-32\verifier.lib.

### Getting TPTP Sources

```
cd <TPTP root>
cvs -d :pserver:anonymous@dev.eclipse.org:/cvsroot/tptp checkout \
platform/org.eclipse.hyades.probekit platform/org.eclipse.tptp.platform.jvmti.runtime
```

### Compiling Martini

1. Open <TPTP root>\org.eclipse.tptp.platform.jvmti.runtime\src-native\build\tptp\_martini.dsw.
2. Select and build the BuildMartini32 project.
3. Open <TPTP root>\org.eclipse.tptp.platform.jvmti.runtime\src-native\build\tptp\_profiler.dsw.
4. Select and build BuildCGProf32 project.
5. Select and build BuildJPIAgent32 project.
6. All profiler binaries are now located (depending on whether you did release or debug build) in org.eclipse.tptp.platform.jvmti.runtime\src-native\bin\windows\\_{debug|release}\IA-32.

### Configuring Profiler Environment

1. Open a command window and configure Java 6 (Sun or BEA) as the active JVM (no need to set JAVA\_HOME).
2. Add the directory containing profiler binaries to the system path and environment:

```
set JAVA_PROFILER_HOME=<TPTP root>\org.eclipse.tptp.platform.jvmti.runtime\src-native\bin\windows\
{debug|release}\IA-32
set PATH=%JAVA_PROFILER_HOME%;%PATH%
```

### Launching a Profiler

1. [under construction] Extract the attached ProxyClasses.zip into <TPTP root>. This should create the following directory structure: org\ eclipse\tptp\martini, and place 3 class files there.
2. Launch the profiler on the Java2D workload (attached):

```
java -XX:-FailOverToOldVerifier -cp <TPTP root>\bin -agentlib:JPIBootLoader=JPIAgent:server=standalone,  
stackmap=true;CGProf java2d.Java2Demo
```

(Thanks to Asaf, Vasily)