The Eclipse TPTP Java Profiler
Ganymede Edition

Chris Elford (chris.l.elford@intel.com)
Content cheerfully taken from presentations by many others
(Most from EclipseCon2007/EclipseCon2008)

June 24, 2008
Portland, OR
Eclipse* TPTP Overview

- Eclipse* Tools Project, Hyades, Dec 2002
- Promoted to Eclipse* project, 2005

- Open-source platform for Automated Software Quality (ASQ) tools including reference implementations for testing, tracing and monitoring software systems

- Addresses the entire test and performance life cycle, from early testing to production application monitoring, including test recording, editing and execution, monitoring, tracing and profiling, and log analysis capabilities

- Integration with tools used in the other processes of a software lifecycle under Eclipse* environment

- Reduce the cost and complexity of implementing effective automated software quality control processes

- Share data through an OMG-defined trace, log, statistical and test model implemented via the Eclipse* Modeling Framework (EMF)

- Active participants:
  IBM, Intel(R), OC Systems

- Former participants:
  Compuware, SAP, Scapa Technologies
The “New” Java* Profiler (JVMTI) vs. JVMPI

§ “New” Java* Profiler went GA in Europa

§ Java* 5.0 introduced new standards for profiling
  § Prior to Java* 5.0 (Java* 1.4-) the standards and the interfaces to support profiling (JVMPI) were experimental
    § Java* J2SE 6 eliminated JVMPI interface
  § The new standard (JVMTI): an innovative solution to profiling; enables precise control over application parts to profile
  § Eclipse* TPTP “new” Java profiler is based on this standard (GA since Europa)

§ In Eclipse* TPTP old profiler (JVMPI) still exists
  § Only for backward compatibility with Java 1.4
  § Because of JVMPI/JVMTI interface differences there are some differences in behavior between old and new profiler
  § Eclipse* TPTP will eventually remove old Java* profiler
Eclipse* TPTP Profiler Updates for Ganymede

- Improve overall usability via assorted fixes
  - Using feedback from users and consumers
- Reduce usability diffs between JVMPI and JVMTI...
  - e.g., SSL for secure communications
  - e.g., Reduce overuse of dynamic attach/detach

Enhancements
- e.g., Better multithread analysis
  - Improved visualization
  - Contended lock analysis
  - Track more thread states (e.g., join points)
- e.g., Efficient binary transfer format
- e.g., Better Java* 6 support
Ganymede Update – Pointing out a few fixes

§ Dynamic attach vs pause/resume

Workload
- Infinite hello world

§ Start from outside the workbench
§ Attach, Detach, Reattach, Pause, Resume

```bash
set TPTP_AC_HOME=C:/utils/eclipse34/eclipse/dropins/eclipse/plugins/org.eclipse.tptp.\platform.ac.win_ia32_4.4.1.v200806171132/agent_controller
set JAVA_PROFILER_HOME=C:/utils/eclipse34/eclipse/dropins/eclipse/plugins/org.eclipse.tptp.\platform.jvmti.runtime_4.4.100.v200806180919/agent_files/win_ia32
set PATH=%JAVA_PROFILER_HOME%;%TPTP_AC_HOME%\bin;%PATH%
set AGENT=-agentlib:JPIBootLoader=JPIAgent:server=enabled;CGProf
java %AGENT% hello
```

see [http://wiki.eclipse.org/TPTP-standalone-script-TPTP4.4.0.3](http://wiki.eclipse.org/TPTP-standalone-script-TPTP4.4.0.3)
Ganymede Update – Thread Contention Analysis

- Better multithread analysis
  - Improved visualization
  - Contended lock analysis
  - Track more thread states (e.g., join points)

Producer
- Processing time
- Write packet to buffer
- Wait for buffer-empty

Consumer
- Wait for buffer full
- Read packet from buffer
- Processing time

- Lets configure and run a thread profile session
- Lets quickly see what's new in the views

TPTP Resources (profiling and beyond)

§ Learn and Try

§ Webs and Wikis
  § http://eclipse.org/TPTP
  § http://wiki.eclipse.org/TPTP

§ Downloads and Updates
  § http://www.eclipse.org/tptp/home/downloads

§ News and Mail
  § http://www.eclipse.org/tptp/home/project_info/general/mailnews.php

§ Use and Participate
  § http://wiki.eclipse.org/TPTP_User_Experiences_Profiling
  § http://www.eclipse.org/tptp/home/project_info/general