Eclipse Memory Analyzer Graduation, Move and 1.0 Release Review

Review Date: 28 April 2010

Community Channel:

•mailto:mat-dev@eclipse.org

•http://www.eclipse.org/forums/index.php?t=thread&frm_id=62&

Author: Krum Tsvetkov (project lead)

Introduction

- Memory Analyzer is a tool for heap dump analysis, which helps in analyzing memory leaks and high memory consumption of Java applications. It works fine with multi-million objects heap dumps and can quickly point to the suspicious objects and who is retaining them in the heap.
- Memory Analyzer is sub-project under the Technology TLP since 2008
  - http://www.eclipse.org/mat
- It was part of the Galileo simultaneous release and is preparing to be part of Helios

Communities

- Contributors and committers
  - 5 committers – 4 from SAP, 1 from IBM
  - Small number of non-committer contributions done via Bugzilla
- Adopters
  - Integrated into SAP NetWeaver CE
  - Integrated into IBM Support Assistant
  - Used as RAP example at several conferences
  - Some additional query and parser plugins are being written as shown by questions on the forum
- Users
  - The standalone RCP application has about 1000 downloads per week over the last year (over 70,000 since the beginning)
  - Part of Galileo update site (downloads not counted)
  - Forums (429 messages in 134 topics) and Bugzilla used as communication channel

Communities – Public Presence

- Eclipse Webinar (May 2008)
- Presentations at Conferences
  - Eclipse Con (2008)
  - Eclipse Summit Europe (2008, 2009)
  - JavaOne (2008, 2009)
  - JAX (2008, 2009)
  - Eclipse Demo Camp Walldorf (May 2009)
  - TheServerSide Java Symposium (October 2009)
  - More …
- Blog http://dev.eclipse.org/blogs/memoryanalyzer/

Open Source Operation

- After the initial contribution by SAP (2008) the development is ongoing in the Memory Analyzer incubator project under the Technology project
- With the joining of Andrew Johnson (IBM) the team has gone to a more open development process
  - features/changes discussed in Bugzilla (product MAT)
  - announcements and discussion on the mat-dev mailing list
  - user interaction mainly via the Memory Analyzer Eclipse forum
- The project maintains a wiki:
  - http://wiki.eclipse.org/index.php/MemoryAnalyzer

IP Issues

- All plugins contain appropriate license files
- All committers have completed Eclipse Committer Agreements and have been approved by the PMC
- All non-committer contributions are properly marked in Bugzilla
- Approved IP Log available at: http://www.eclipse.org/mat/1.0/Approved_IP_Log.pdf, Bug 299253

Move (page 1/2)

- After graduation the Memory Analyzer project will move from Technology to Tools top-level project
- Repositories
  - /cvsroot/org.eclipse/www/mat – no change needed
  - /svnroot/technology/org.eclipse.mat – should be move/renamed to /svnroot/tools/org.eclipse.mat
- CQs
  - 3794 | DTFJ: Subject to distribution under this license and no other, any version |International License Agreement for Non-Warranted Programs |works with
  - 3854 | vm.jar from IBM VMs Version: IBM Java 6 | International License Agreement for Non-Warranted Programs | works with
- Committers
  - No new committers are expected. All current committers should be moved
- Web Pages
  - Eclipse projects list page has to be updated http://eclipse.org/projects/listofprojects.php

Move (page 2/2)

- Bugzilla Products
  - Product: MAT - no change needed
- Newsgroups
  - No change needed
- Mailing Lists
  - mat-dev{at}eclipse.org - No change needed
- Builds
  - The project will need to update its build jobs on the Hudson at build.eclipse.org

Features

- Report memory leak suspects
- Report memory waste – redundant Strings, empty collections
- Calculate retained sizes
- Find who is keeping objects alive
- Query heap with an SQL-like language
- Works with multi GB heap dumps
- Supports various dump formats, e.g. HPROF binary dumps, IBM PHD and system dumps (via IBM's Diagnostic Tooling Framework for Java)
- Thread stack Information + Java locals

+ Trigger heap dumps from within the tool

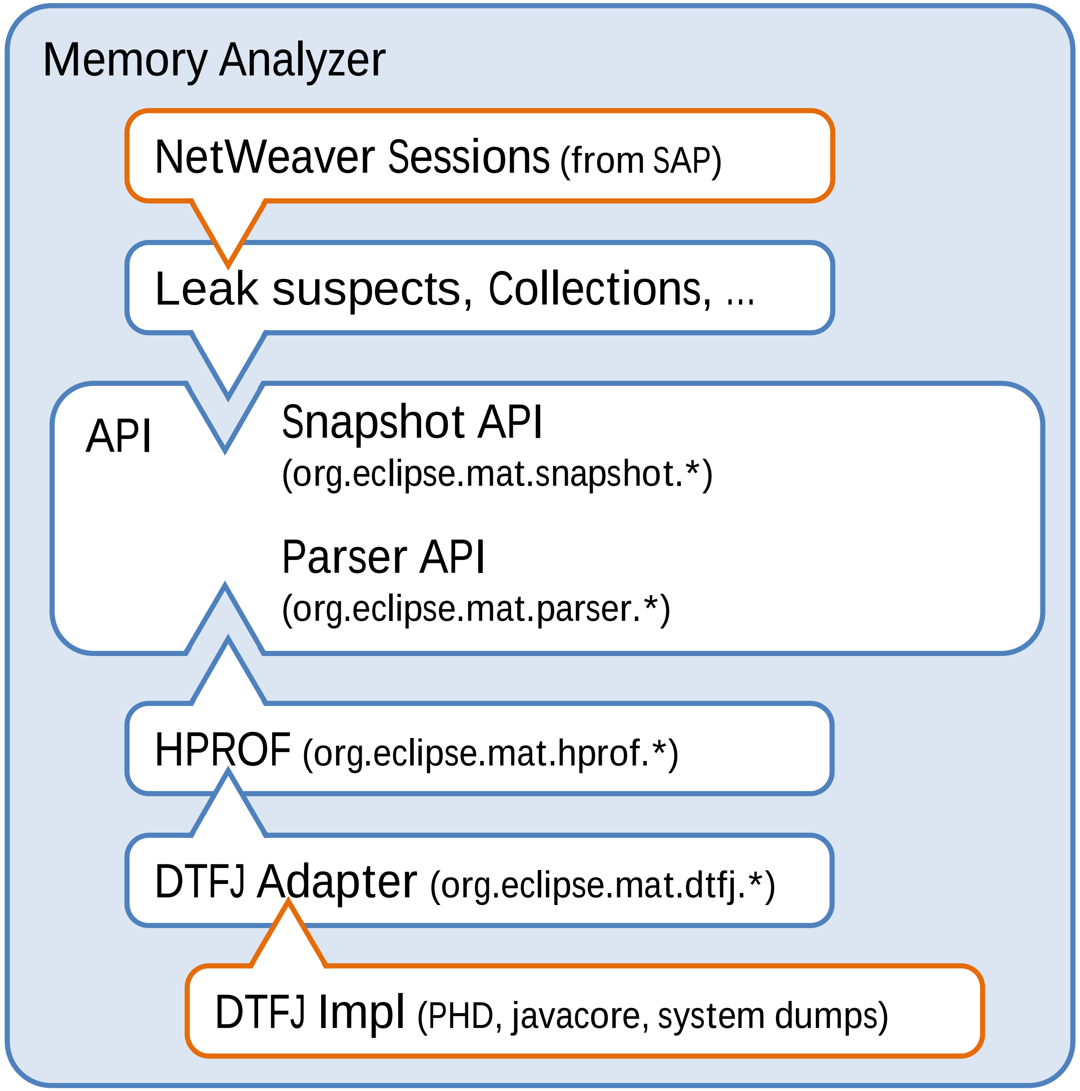
+ Compare any two or more table-formatted results

Non-Code Aspects

- Documentation is generated using DITA
  - provided via the help center
- Online documentation via
  - WIKI http://wiki.eclipse.org/index.php/MemoryAnalyzer
  - Webinar http://live.eclipse.org/node/520
  - Blog http://dev.eclipse.org/blogs/memoryanalyzer
- Cheat sheets

Summary: a wealth of material is available, but often brief and not easily accessible to non-domain experts.

APIs


Memory Analyzer
NetWeaver Sessions (from SAP)
Leak suspects, Collections, ...
API
Snapshot API
(org.eclipse.mat.snapshot.*)
Parser API
(org.eclipse.mat.parser.*)
HPROF (org.eclipse.mat.hprof.*)
DTFJ Adapter (org.eclipse.mat.dtfj.*)
DTFJ Impl (PHD, javacore, system dumps)

The Memory Analyzer provides two major interfaces:

a)The Snapshot API provides access to the logical object graph inside the heap. It enables inspections that analyze collections, identify leak suspects etc.

b)The Parser API makes reading the raw heap dump format pluggable.

APIs conform with Eclipse Quality Standards.

▢	MAT @ Eclipse.Org
▢	(known) 3rd Party Extensions

Architectural Issues

Summary: Architecture is settled and performs well on multi-GB heap dumps

Tool Usability

- The Memory Analyzer tool is very helpful for troubleshooting of OutOfMemoryErrors. It can be also used proactively to analyze and reduce memory consumption.
- The tool provides rich and responsive UI.
- The sheer number of heap inspections can be overwhelming for a novice user.

End-Of-Life

- This is the first release after graduation. No feature is removed
- Changes to the API are documented in bug 299371: MAT API changes for 1.0 https://bugs.eclipse.org/bugs/show_bug.cgi?id=299371

Bugzilla

- Messages statistics
  - Total 152
    - Open 40
    - Closed 112
- Bugzilla used for discussions on new features / modifications
- No major defects open for 1.0

Standards

MAT requires

- Execution Environment J2SE-1.5
- Eclipse Platform 3.4 or higher
- BIRT Chart Runtime 2.2.0 or higher

UI Usability

- Follow User Interface Guidelines
- Multi-language support
  - UI Strings are externalized via Eclipse NLS
  - Memory Analyzer is part of Babel
- Accessibility was not systematically tested
  - Several fixes for better keyboard support (bug 300655)

Schedule

- Release 1.0.0 (May 2010)
  - Join the Helios release
  - Trigger heap dumps from MAT
  - Comparison of multiple table results
- Release 1.0.1 (TBD)
  - Bugfixes for 1.0.0
- Release 1.1.0 (TBD)
  - Improve comparison features
  - Research analysis techniques based on comparison
  - Provide code samples for MAT queries

Project Plan

Available at

http://www.eclipse.org/projects/project-plan.php?projectid=technology.mat

Future Themes

- Research / provide comparison based analysis
- Detect more anti-patterns (e.g. leaks in WeakHashMap where values reference the keys)
- Closer integration into the IDE
- Research possible integration of memory analysis within system tests