The Grinder, a Java Load Testing Framework

Table of contents

1 What is The Grinder?....2

1.1 Key features....2

1.2 Dynamic Scripting....2

1.3 History....3

2 Authors....3

3 Credits....3

1 What is The Grinder?

The Grinder is a Java^{TM} load testing framework that makes it easy to run a distributed test using many load injector machines. It is freely available under a BSD-style open-source license (license.html) .

The latest news, downloads, and mailing list archives can be found on SourceForge.net (https://www.sourceforge.net/projects/grinder) .

1.1 Key features

- **Generic Approach** Load test anything that has a Java API. This includes common cases such as HTTP web servers, SOAP and REST web services, and application servers (CORBA, RMI, JMS, EJBs), as well as custom protocols.
- **Flexible Scripting** Test scripts are written in the powerful Jython (http://www.jython.org/) and Clojure (http://clojure.org/) languages.
- **Distributed Framework** A graphical console allows multiple load injectors to be monitored and controlled, and provides centralised script editing and distribution.
- **Mature HTTP Support** Automatic management of client connections and cookies. SSL. Proxy aware. Connection throttling. Sophisticated record and replay of the interaction between a browser and a web site.

See the longer features list (g3/features.html) for further details.

1.2 Dynamic Scripting

Test scripts are written using a dynamic scripting language, and specify the tests to run. The default script language is Jython (http://www.jython.org/) , a Java implementation of the popular Python language.

The script languages provide the following capabilities:

Test any Java code
The Grinder 3 allows any code (Java, Jython, or Clojure) code to be encapsulated as a test. Java libraries available for an enormous variety of systems and protocols, and they can all be exercised using The Grinder.

Dynamic test scripting
The Grinder 2 worker processes execute tests sequentially in a fixed order, and there is limited support in some of the The Grinder 2 plug-ins for checking test results. The Grinder 3 allows arbitrary branching and looping and makes test results directly available to the test script, allowing different test paths to be taken depending on the outcome of each test.

The Grinder 2 HTTP plug-in's string bean (g2/http-plugin.html#string-bean) feature provides simple support for requests that contain dynamic data. The Grinder 3 can use the full power of Jython or Clojure to create dynamic requests of arbitrary complexity.

The powerful scripting removes the need to write custom plug-ins that extend The Grinder engine. Although plug-ins are no longer responsible for performing tests, they can still be useful to manage objects that the tests use. For example, the standard HTTP plug-in manages a pool of connections for each worker thread, and provides an `HTTPRequest` object that makes use of these connections.

Kind of dry, huh? If you never seen any Python, take a look at the Script Gallery (g3/script-gallery.html) in the user manual where you can sample the power of The Grinder 3.

1.3 History

The Grinder was originally developed for the book *Professional Java 2 Enterprise Edition with BEA WebLogic Server* by Paco Gómez and Peter Zadrozny. Philip Aston took ownership of the code, reworked it to create The Grinder 2, and shortly after began work on The Grinder 3. The Grinder 3 provides many new features, the most significant of which is dynamic test scripting. Philip continues to enhance and maintain The Grinder.

In 2003, Peter, Philip and Ted Osborne published the book J2EE Performance Testing (links.html#book) which makes extensive use of The Grinder 2.

Support for Clojure (http://clojure.org/) as an alternative script language was introduced in 3.6.

2 Authors

Over the years, many individuals (./mvn-site/team-list.html) have contributed features, bug fixes, and translations to The Grinder.

3 Credits

I thank Paco Gómez and Peter Zadrozny for the key ideas embodied in the original version of The Grinder.

I am grateful to SourceForge, Inc. (http://www.sourceforge.com/) for The Grinder's home on the Internet.

I thank Atlassian (http://www.atlassian.com/) for the free Clover (http://www.atlassian.com/clover) and FishEye (http://fisheye3.cenqua.com/browse/grinder/) licenses, and to Headway Software (http://www.headwaysoftware.com/) for the free Structure 101 (http://www.headwaysoftware.com/products/structure101) license.

This site is built with Apache Forrest (http://forrest.apache.org/) , and uses SyntaxHighlighter (http://alexgorbatchev.com/SyntaxHighlighter/) .

Philip Aston