

Features of The Grinder 3

Table of contents

1 Capabilities of The Grinder.....	2
2 Open Source.....	2
3 Standards.....	2
4 The Grinder Architecture.....	3
5 Console.....	3
6 Statistics, Reports, Charts.....	3
7 Script.....	4
8 The Grinder Plug-ins.....	4
9 HTTP Plug-in.....	4
10 TCP Proxy.....	5
11 Documentation.....	5
12 Support.....	5

Thanks to **Edwin DeSouza** for his help in compiling this feature list.
Last updated: 5 January 2008

1. Capabilities of The Grinder

Load Testing	Load Testing determines if an application can support a specified load (for example, 500 concurrent users) with specified response times. Load Testing is used to create benchmarks.
Capacity Testing	Capacity Testing determines the maximum load that an application can sustain before system failure.
Functional Testing	Functional Testing proves the correct behaviour of an application.
Stress Testing	Stress Testing is load testing over an extended period of time. Stress Testing determines if an application can meet specified goals for stability and reliability, under a specified load, for a specified time period.

2. Open Source

BSD style license	The Grinder is distributed under a BSD style license.
Dependencies	The Grinder depends on a number of other open source products including <ul style="list-style-type: none"> • Jython (http://www.jython.org/) • HTTPClient (http://www.innovation.ch/java/HTTPClient/) • JEdit Syntax (http://syntax.jedit.org/) • Apache XMLBeans (http://xmlbeans.apache.org/) • PicoContainer (http://picocontainer.codehaus.org/)

3. Standards

100% Pure Java	The Grinder works on any hardware platform and any operating system that supports J2SE 1.4 and above.
Web Browsers	The Grinder can simulate web browsers and other devices that use HTTP, and HTTPS.
Web Services	The Grinder can be used to test Web Service interfaces using protocols such as SOAP and XML-RPC.
Database	The Grinder can be used to test databases using JDBC.
Middleware	The Grinder can be used to test RPC and MOM

	based systems using protocols such as IIOP, RMI/IIOP, RMI/JRMP, and JMS.
Other Internet protocols	The Grinder can be used to test systems that utilise other protocols such as POP3, SMTP, FTP, and LDAP.

4. The Grinder Architecture

Goal	Minimize system resource requirements while maximizing the number of test contexts ("virtual users").
Multi-threaded, multi-process	Each test context runs in its own thread. The threads can be split over many processes depending on the requirements of the test and the capabilities of the load injection machine.
Distributed	The Grinder makes it easy to coordinate and monitor the activity of processes across a network of many load injection machines from a central console.
Scalable	The Grinder typically can support several hundred HTTP test contexts per load injection machine. (The number varies depending on the type of test client). More load injection machines can be added to generate bigger loads.

5. Console

Graphical Interface	100% Java Swing user interface.
Process coordination	Worker processes can be started, stopped and reset from one central console.
Process monitoring	Dynamic display of current worker processes and threads.
Internationalised and Localised	English, French, Spanish, and German translations are supplied. Users can add their own translations.
Script editing	Central editing and management of test scripts.

6. Statistics, Reports, Charts

Test monitoring	Pre-defined charts for response time, test throughput. Display the number of invocations, test result (pass/fail), average, minimum and maximum values for response time and tests per second for each test.
Data collation	Collates data from worker processes. Data can be saved for import into a spreadsheet or other analysis tool.

Instrument anything	The Grinder records statistics about the number of times each test has been called and the response times achieved. Any part of the test script can be marked as a test.
Statistics engine	Scripts can declare their own statistics and report against them. The values will appear in the console and the data logs. Composite statistics can be specified as expressions involving other statistics.

7. Script

Record real users	Scripts can be created by recording actions of a real user using the TCP Proxy. The script can then be customised by hand.
Powerful scripting in Python	Simple to use but powerful, fully object-oriented scripting.
Multiple scenarios	Arbitrary looping and branching allows the simulation of multiple scenarios. Simple scenarios can be composed into more complex scenarios. For example, you might allocate 10% of test contexts to a login scenario, 70% to searching, 10% to browsing, and 10% to buying; or you might have different workloads for specific times of a day.
Access to any Java API	Jython allows any Java-based API to be used directly from the test script.
Parameterization of input data	Input data (e.g. URL parameters, form fields) can be dynamically generated. The source of the data can be anything including flat files, random generation, a database, or previously captured output.
Content Verification	Scripts have full access to test results. In the future, The Grinder will include support for enhanced parsing of common results such as HTML pages.

8. The Grinder Plug-ins

HTTP	The Grinder has special support for HTTP that automatically handles cookie and connection management for test contexts.
Custom	Users can write their own plug-ins to a documented interface; although this is rarely necessary due to the powerful scripting facilities.

9. HTTP Plug-in

HTTP 1.0, HTTP 1.1	Support for both HTTP 1.0 and HTTP 1.1 is
--------------------	---

	provided.
HTTPS	The Grinder supports HTTP over SSL.
Cookies	Full support for Cookies is provided.
Multi-part forms	The Grinder supports multi-part forms.
Connection throttling	Low bandwidth client connections can be simulated.

10. TCP Proxy

TCP proxy	A TCP proxy utility is supplied that can be used to intercept system interaction at the protocol level. It is useful for recording scripts and as a debugging tool.
HTTP Proxy	The TCP proxy can be configured as an HTTP/HTTPS proxy for easy integration with web browsers.
SSL Support	The TCP proxy can simulate SSL sessions.
Filter-based architecture	The TCP proxy has a pluggable filter architecture. Users can write their own filters.

11. Documentation

User Guide	http://grinder.sourceforge.net/g3/whats-new.html (../g3/whats-new.html)
FAQs	http://grinder.sourceforge.net/faq.html (../faq.html)
Tutorial	http://grinder.sourceforge.net/g3/tutorial-perks.html (../g3/tutorial-perks.html)
Script Gallery	http://grinder.sourceforge.net/g3/script-gallery.html (../g3/script-gallery.html)
Articles	http://grinder.sourceforge.net/links.html (../links.html)
Commercial books	<i>Professional Java 2 Enterprise Edition with BEA WebLogic Server</i> J2EE Performance Testing (../links.html#book)

12. Support

Mailing Lists	grinder-use@lists.sourceforge.net (mailto:grinder-use@lists.sourceforge.net) grinder-development@lists.sourceforge.net (mailto:grinder-development@lists.sourceforge.net) grinder-announce@lists.sourceforge.net (mailto:grinder-announce@lists.sourceforge.net)
---------------	---

