STRESS, LOAD, VOLUME, PERFORMANCE, BENCHMARK AND BASE LINE TESTING TOOL EVALUATION AND COMPARISON

Cordell Vail

MakingSuccessWork@yahoo.com
www.vcaa.com

NOTE: The information contained in this document or on the handout CD at the seminars is for use only by the participants who attend one of our seminars. Distribution of this information to anyone other than those attending one of the seminars is not authorized by the authors. It is for educational purposes of the seminar attendees only.

It is our intention that by putting this information here from the vendor web pages and from other testers evaluations, that you will have a tool that will let you do your own evaluation of the different tool features. Hopefully this will let you find tools that will best meet your needs.

The authors are not recommending any of these tools as one being better than another. All the information has been taken from reviews we have found on the Internet, from their own web pages, or from correspondence with the vendor. They have been grouped here according to our presentation outline. For some of the tools, we were not able to determine the cost or type (ie open source) from their web page or correspondence with the vendor.

Users are cautioned that they will be installing any downloaded tools at their own risk.

TABLE OF CONTENTS

Table Of Contents Key:
EVALUATION SECTION **Page**
VENDOR (and foreign country if known)
Tool Name [type and price if known]

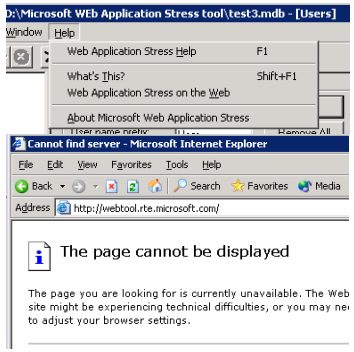
TOOLS WORTH LOOKING AT (the listing by price does not imply preference)

~~~~~~

VENDOR:		***MICROSOFT***		
TOOL NAME:		**Web App Stress Tool (WAS) [free]**		
Web Page		http://www.microsoft.com/downloads/details.aspx?familyid=e2c0585a-062a-439e-a67d-75a89aa36495&displaylang=en		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Overview The Microsoft WAS web stress tool is designed to realistically simulate multiple browsers requesting pages from a web site. You can use this tool to gather performance and stability information about your web application. This tool simulates a large number of requests with a relatively small number of client machines. The goal is to create an environment that is as close to production as possible so that you can find and eliminate problems in the web application prior to deployment. System Requirements **Supported Operating Systems:** Windows 2000, Windows NT Required operating system: Windows 2000, Windows NT 4.0 Service Pack 4 with Internet Explorer 4.0 or newer. Microsoft Data Access Components 2.1 are required, but will be installed if not present.	FREE for unlimited users	INSTALL Very user friendly and easy to do. It did not put an ICON on the desk top. I had to go find it. The following was in the README file: *Web Application Stress Tool only supports stress against one server or DNS name at a time.* *To use authentication, you must add users with valid user names and passwords.* *To record, you will need to modify your browser settings if using a browser other than, or prior to, IE 5.0. See online help for more information.*
~~~~~~

		Performance Testing with the Web Application Stress Tool Published: January 1, 2001 By Aaron Ching , Pedro Silva , and Allen Wagner, Microsoft Developer Network **Duwamish Online** **Summary:** This article discusses the importance of performance testing in the successful deployment of a Web application, focusing on Microsoft Web Application Stress (WAS), the tool used to test the performance of the Duwamish Online application. Introduction Performance testing is an essential element in successfully deploying a Web application. It's important to understand how the application and your Web server farm would behave as more and more users visit the Web site. In order to simulate that type of usage for a Web application, you would either need to coordinate with hundreds or even thousands of real users to access your Web site within a designated period of time—or work with a testing tool that can reproduce such user loads. Many Web performance testing tools are available to help. Basically, these tools allow you to use a minimal number of client computers to simulate a large number of virtual users, concurrently requesting predefined pages or *URL*s (Uniform Resource Locators) of the Web site. Each of these virtual users emulates the exact communication protocols between a real Web browser and the Web server. In this article, we are going to focus on one of these tools: the freely available Microsoft® Web Application Stress (WAS) tool. WAS can be downloaded free of charge from http://www.microsoft.com/technet/archive/itsolutions/intranet/downloads/webstres.mspx. It is also available from the Microsoft Windows® 2000 Resource Kit CD (WAS version 288). Benefits of WAS First, we will discuss some of the benefits of using WAS to test the performance of your application.		*Webtool respects the "user-agent: stress-agent" directive in the ROBOTS.TXT file on a web server and will not start a stress test against a server or domain that contains this file with this directive. This file should be located in the root directory of your web site* *In general, you should NOT install the WebTool service on the web server you are testing since this may interfere with your results.* *The Web Application Stress Tool object model can be used to start, stop, and configure a test run. However, this feature cannot be used to modify a running test in this version of the tool.* *Web Application Stress Tool is year 2000 compliant.* *IMPORTANT: Start your test with a small number of Concurrent Connections and then work your way up to the desired stress level slowly, watching the processor utilization on the clients and web server. If the processor utilization is consistently above 90% on one or more machines, the test is probably invalid. If this occurs, decrease the number of concurrent connections or add more client machines to handle the desired stress load.* *On-line help contains a discussion of performance testing techniques.* *Although there is no hard limit on the number of script*

		It's Free As already noted, unlike some other Web performance testing tools, Microsoft's Web Application Stress tool can be downloaded free of charge from the WAS Web site. Although there is no official support for the tool, you will be able to find useful information on this site. The best resource available is a knowledge base that provides in-depth answers to the most frequently asked questions regarding performance testing. You can also send your questions to the peer support e-mail alias webtool@microsoft.com. It's Simple WAS allows you to easily generate a test script in several different ways: You can record a script by walking through a site with a browser, by importing the URLs from Web server log files, or by selecting files in a Web content directory. And, of course, you can also manually type in the URLs to create a new test script. Unlike some other tools, you can run a test script using any number of client computers, all controlled by one centralized master client. At the beginning of each test, the master client performs the following tasks transparently: • Communicates with all the other clients • Distributes the test data to all clients • Initiates the test in all clients simultaneously • Collects the test results and reports from all clients This feature is particularly helpful in performance testing a big Web farm that might require the use of many client computers in order to maximize its throughput. It's Highly Usable WAS is designed to simulate Web browser requests for any standard Web server compliant with the Hypertext Transport Protocol (HTTP) 1.0 or 1.1 standards, regardless of the platform on which the server is run. In addition to its easy-to-use test generation features, WAS has many other useful features, including:		*items you can have in one script, the most practical limit is 3000 script items. Beyond that you will find that it takes an inordinately long time to start and stop tests.* *Happy Testing,* *The Web Application Stress Tool Team* ~~~~~~~~~~~~~~~~~~~~~~~~ This version covers the most needed features for stress testing three tier personalized Active Server Page web sites running on NT server. The help is quite good. But the application is not as intuitive as one would hope to see it be. When you run your script it does not allow you to watch the test run and the results are very “TEXT” oriented rather than charts and graphs. The manual coding did seem that hard to learn record and play back but seems very restrictive in what it allows you to do. If you write the code yourself that looks very complex (which is what I have heard bad about this tool from other testers)

		Allows the creation of user accounts for sites that require authentication Allows the storage of cookies and Active Server Pages (ASP) session information for each user Supports randomized or sequential data set for use in specifying name-value pairs Supports bandwidth throttling and randomized delay ("think time") to simulate more realistic scenarios Supports the Secure Sockets Layer (SSL) protocol Allows the grouping of URLs and specification of hit percentage for each group Provides an object model that can be manipulated through Microsoft Visual Basic® Scripting Edition (VBScript) or a custom program to start, stop, and configure test scripts programmatically Limitations of WAS Advantages aside, WAS does have a few limitations. The current known bugs and issues are listed on the WAS Web site. Here are some features that are not currently supported in WAS: The ability to modify the parameters of a URL, based on results returned from previous requests The ability to run or emulate client-side logic The ability to specify a fixed number of test cycles for the duration of the test The ability to run tests simultaneously against multiple Web servers with different IP addresses or domain names **Note:** You can use multiple master clients to test against multiple Web servers simultaneously. However, you'll need to consolidate the data from multiple WAS databases if you want to correlate all test results as a whole. The ability to support redirection of pages to another server with a different IP address or domain name The ability to record SSL pages directly from a Web browser **Note:** WAS already supports testing against SSL pages, but not recording them. You are required to manually turn on SSL support for each designated URL after script recording is complete.		On a Microsoft web page I would not expect to get a page not found error in the help files but I did  Just using this tool for a couple of hours it does not compare to some of the others that you pay for, especially the reporting. But it certainly is not as bad as all the press that I have heard about it for that past couple of years. I good programmer likely could make good use of this free tool. Certainly the price is right.

Although there are workarounds for some of these limitations, you might not be able to take full advantage of the benefits of WAS if your application depends on one or more of these capabilities.

Currently, the development effort of WAS has been focused on implementing the next-generation Web application performance testing tool—Microsoft Application Center Test (ACT). This new tool, a beta version of which was recently released as part of the Visual Studio.NET beta, is designed to replace WAS with an improved UI and additional features.

Installing WAS

The Web Application Stress tool requires Microsoft Windows NT® 4.0 Service Pack 4 or later, including the Windows 2000 platforms. It also requires Internet Explorer 4.0 or later, with Internet Explorer 5.0 being preferable.

To install the Web Application Stress tool, download the latest version of the setup.exe program. Follow the installation wizard instructions. Copy and run setup.exe on all of your test computers.

Note: All procedures presented in this article are based on WAS version 293.

Creating Test Scripts

Although you can create a test script manually, WAS can assist by recording browser activities, importing a Web server log file, or evaluating the contents of a Web content directory. In this article, we'll focus on creating a test script by recording the Web browser activities. This method is chosen over the others for several reasons, including:

- This method of recording browser activities captures all user interactions with a high level of accuracy. Any URL references, application parameters, and HTTP header information sent from the browser to the Web server will be automatically recorded in the new test script.
- The method of importing Web server log files is best used for sites that have already been in production stage with real user traffic. However, a new Web site might not have enough of such real user data yet. Furthermore, it might be necessary to incorporate a large data set of the log file in order to achieve a better representation of the actual user activities. This will require the creation of a large test script that demands

greater system resources on the client computers.
The method of selecting a Web content directory is best used for testing sites with mostly static HTML files. This method allows quick creation of a test script based on existing Web pages on the server. However, this method does not capture any application parameters generated by most application files, such as Common Gateway Interface (CGI) programs or Active Server Pages (ASP).

You will only need to create and store your test scripts on the master client computer. The scripts will be automatically distributed to other clients when tests are initiated from the master client.

Preparing the Client Computer

If you are using WAS over a proxy server on your corporate network and requesting pages from clients outside of your corporate network, and your company is using Microsoft Proxy Server, follow these steps to set up your clients:

1. From the **Start** menu, point to **Settings** \ **Control Panel**. Double-click the **Administrative Tools** icon, and then the **Services** icon.
2. Double-click the WebTool service to open the **Properties** dialog box.
3. Click the **Log On As** tab, and then click the **This account** option button to add your network user name and password. Use the domain\user name format.
4. Stop and restart the WebTool service.
5. Next, install the Microsoft Windows Proxy client 2.0 software. This is also known as the Winsock Proxy client, which can be found on the Microsoft Proxy Server CD. (For more information on how to install and configure the software, please refer to the documentation included on the CD.)
6. Repeat steps 1-5 for each test client you intend to use over the proxy server.

If your company uses other proxy servers, locate and install the proxy client software that comes with that particular proxy server.

Preparing the Browser

Before starting to record a script, you need to prepare your Web browser by

		clearing the browser cache. Otherwise, WAS might not be able to record the required browser activities because the browser might be retrieving Web pages from its own cache instead of requesting them from the Web server. **To turn off caching in Internet Explorer** 1. On the **Tools** menu, click **Internet Options**. 2. Click the **General** tab, and then click the **Delete Files...** button. You will not need to change any of the proxy settings if using Internet Explorer 5.0 or later, which includes new features that allow WAS to modify the settings programmatically. However, for Internet Explorer 4.0 or earlier, WAS uses a built-in proxy server to record browser activities. **To specify the proxy settings as required by WAS** On the **Tools** menu, click **Internet Options**. From the **Connections** tab, modify the proxy settings so that the proxy server is pointing to **Localhost** and the port used is **8000**. Clear the **Bypass proxy server for local addresses** check box. Recording the Script After your browser and clients are ready for recording, do the following: 1. When you first start the WAS program, you will see the **Create new script** dialog box (Figure 1), asking you to specify a method of creating a new test script.		

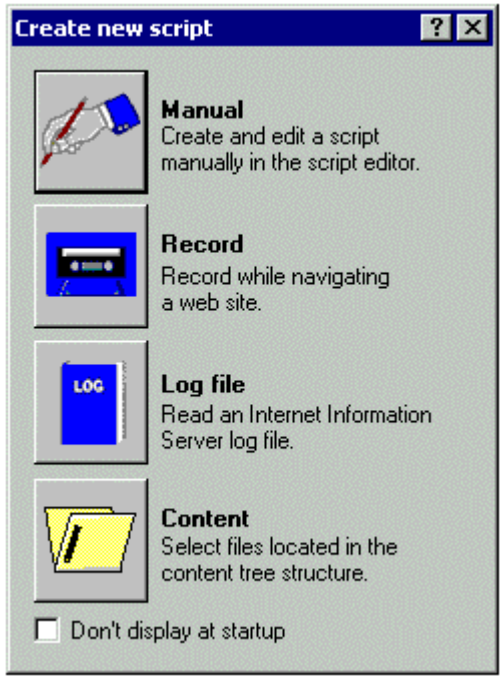


Figure 1: Creating the script

2. Click the **Record** button. If you have previously selected the **Don't display at startup** check box, the **Create new script** dialog box will not be displayed. Instead, from the **Script** menu, select **Record** and then **Create**.
3. In the **Browser Recorder** — **Step 1 of 2** dialog box, you will be asked to specify some recorder settings. In our case, clear all check boxes and click **Next** to continue.
4. In the **Browser Recorder** — **Step 2 of 2** dialog box, click **Finish**. A new Internet Explorer window will be created for recording browser activity and the WAS program window will be placed in recording mode.
5. In the **Address** bar of the new Internet Explorer window, type in the Web address of your target site. In the WAS program window, you will see the HTTP header information in real time as you navigate through the site.

6. When finished with your site navigation, switch back to the WAS program window—which is still in recording mode—and click the **Stop Recording** button. This will terminate the recording and create a new test script, as shown in Figure 2.

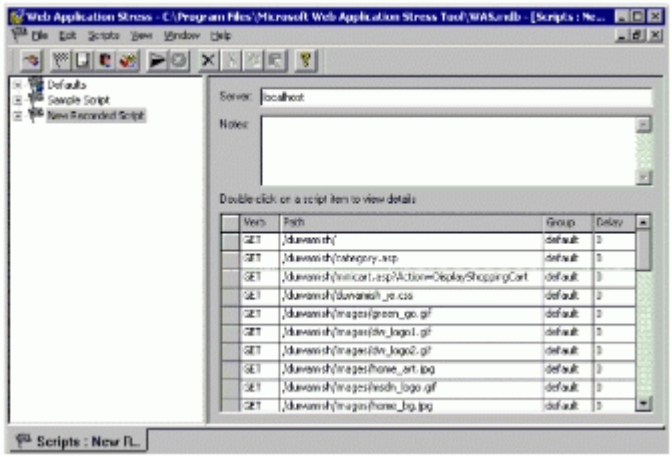


Figure 2: The WAS program window after recording is finished
See full-sized image.

On the bottom portion of the right pane, you will see a table listing all script items.

For Web sites that require secured connections, WAS does support testing against Secure Sockets Layer (SSL)-enabled pages. However, it does not allow SSL recording. To work around this limitation, you can turn off SSL on the Web server, record the script, and then re-enable SSL on the Web.

Configuring Test Scripts

The newly created test script is not ready to be used for testing just yet. One or more of the following configuration tasks must still be completed:

- Adjust script items and their properties.
- Adjust test script settings.
- Set up page groups and hit percentages.
- Set up user accounts.

		• Set up clients. • Set up performance counters. Adjusting Script Items There are a few considerations in modifying script items of a test script, which we will discuss next. **Removing undesirable script items** Reduce the noise factor in your test by removing duplicate items, or those with invalid URLs. When tuning a particular function, remove all script items that refer to images, style sheets, and other auxiliary static files. **Specifying think time for script items** The last column of the script items table is titled "Delay." This column allows you to specify a fixed delay time (also called *think time*) prior to the execution of the script item. There is not a single standard of determining what is the right think time to use for performance testing. While some might use think time of zero seconds in their testing, others might consider using think time of 30 seconds. The choice depends mainly on the content of the site and the purpose of the test. For example, a Web site with lengthy page content might use longer think time than a simple site with short pages because users are expected to take more time to read through the content. On the other hand, think time of zero seconds might be used if your objective is to quickly determine the maximum throughput of a Web server with a minimal number of client computers. With no think time, every thread of WAS works at full speed in generating load against the Web server. **Setting up a list of values for script items** Instead of using the same form value for each request, WAS allows you to assign a list of values to a name-value pair of a script item. This is an important feature for building realistic test scenarios where users are not expected to call the same page with the same set of form values over and over again.		

For example, one of the items in our test scripts calls an ASP page to display the detail information of a product item. Instead of calling the page with the same product ID every time, we configure WAS to randomly pick a different value from a list of predefined product IDs.

To set up a list of values for a script item

1. In the script items table of the WAS program window, double-click the square button (in the first column of the table) of a specific script item to open the details screen of the item.
2. From the **Querystring** tab (also known as Querystring Editor, shown in Figure 3), select the **Format data to CGI standard** check box. The associated name-value pairs will then appear in the table below the check box.

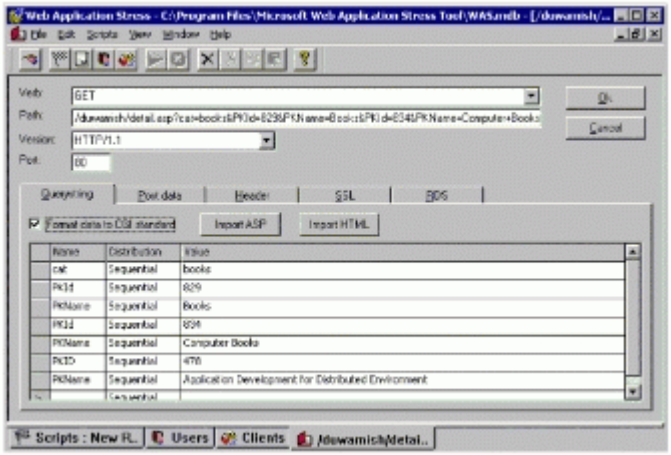


Figure 3: Querystring Editor screen
See full-sized image.

3. Click the value field of the specific name-value pair. A new **…** button will appear.
4. Click the **…** button to open the **Field Values** dialog box.
5. Enter the list of values in the **Field values** dialog box, with each value on a separate line. You may also cut and paste the list from a spreadsheet or a data file.
6. In the Querystring Editor, click the **Distribution** column of the same name-value pair in the table. Select **Random** from the drop-down menu.

Setting up SSL for script items

To enable SSL for a particular script item, do the following:

1. In the script items table of the WAS program window, double-click the square button (in the first column of the table) of a specific script item to open the details screen of the item.
2. From the **SSL** tab, select the **Use SSL** check box. (Note that the port number should change from 80 to 443 when you enable SSL for the item.)

Adjusting Script Settings

You will need to modify some settings of your test script in order to satisfactorily run performance tests. By double-clicking the script name in the left pane to expand the information tree of the script, you will find an item labelled **Settings**, where you can specify many of these settings for your test script. Clicking it will open the Settings view in the right pane, as shown in Figure 4.

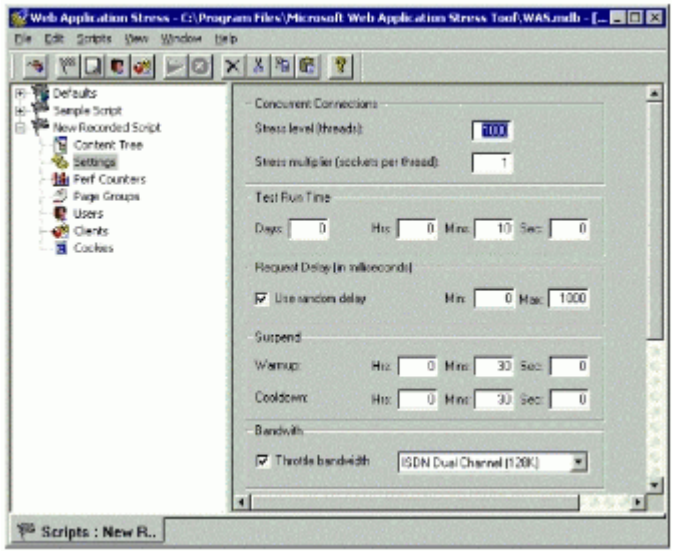


Figure 4: Settings view screen
See full-sized image.

Specifying the target Web server

By default, the target Web server is set to be "localhost," which should be replaced with the IP address or the domain name of the target Web server.

To change this setting

1. Click the name of the test script in the left pane.
2. Enter the IP address or the domain name of the target Web server into the **Server** field, located at the top of the right pane.

Note: Enter the cluster IP address in this field if you want to run tests against a Web server farm with Network Load Balancing, as is the case with Duwamish Online.

Specifying concurrent connections

Under the **Concurrent Connections** section of the Settings view, you can specify the value of **Stress level (threads)** and **Stress multiplier (sockets per thread)** to control the load/stress level applying to the target Web server. Stress level is the total number of Windows NT threads that are created across all of the clients. Each thread can create multiple sockets and each socket is a concurrent request.

The following formula explains this relationship:

Total Concurrent Requests = Stress level (threads) x Stress multiplier

(sockets per thread) = Total Number Sockets

In our performance lab, we use various values of Stress level (threads) for our performance tests. For example, we have used values of 100, 200, 300, 400, 500, 750, 1000, 1500, and 2000 in consecutive test runs for studying how well our server farm responds to increasing load levels.

You should adjust these numbers based on your preliminary test results. In

		general, you want to collect more data points at lower thread levels where system throughput is expected to increase linearly with an increasing number of threads. On the other hand, you can save time and effort by running fewer tests at high thread levels, especially when system throughput has peaked and leveled out. Note that our first test is set to be run at 1000 threads. The purpose of this is to run enough requests to build up the data cache used in our application. Because the performance of applications with and without cache can differ dramatically, this helps to maintain a consistent system environment for load testing. **Specifying test run time** Under the **Test Run Time** section of the Settings view, you can specify the total run time by days, hours, minutes, and seconds. Depending on the expected latency (or response time) of your script items, it's recommended that you run your test script for at least a few minutes in order to generate a large enough number of requests to avoid skewed test results. The higher the latency in your application, the longer the test should be run to sample a large set of data. You can run frequent but short tests to monitor and investigate any performance problems of your site. In addition, you will want to run much longer tests (for example, as long as 30 days) to see if performance of the site degrades over time, especially when it's under a medium or high level of load. In Duwamish Online, most of our performance tests are run for 7 to 10 minutes, which is enough time to stabilize the test results. **Specifying random delay** Under the **Request Delay** section of the Settings view, you can choose to add random delay time (or think time) to each script item prior to execution. If the **Use random delay** check box is selected, each WAS thread will idle for a random period of time (between the **Min** and the **Max** values) in addition to the fixed think time specified for each script item. The following formula explains the delay calculation: Total delay for each item = Random delay + Fixed delay for each item		

		This random delay feature is particularly useful when fixed delays are specified for the script items. Without the use of random delay, all threads are likely to send their requests to the Web server at about the same time, and then wait for about the same fixed delay time before the next requests are sent. Random delay helps to even out the peaks and valleys when applying load against the Web server, thus representing a more accurate testing scenario for the desired load level. **Specifying suspend times** Under the **Suspend** section of the Settings view, you can specify the **warmup** and **cooldown** times in hours, minutes, and seconds. Warmup time is the length of the initial test run during which no performance data would be collected or calculated into the test result. Similarly, cooldown time specifies the ending period of the test run with no data collection. Warmup and cooldown times are used to minimize distortion of test results. Usually, in the initial stage of a new test run, more system resources are consumed by certain one-time activities, such as component or application cache initialization. Warmup time helps to stabilize the system environment before any test data is collected. On the other hand, cooldown time helps to avoid skewed data at the end of a test run, when additional system resources are used to stop the test and start collecting data from all clients. Furthermore, socket connections might be terminated prematurely, contributing to a larger percentage of socket errors. In Duwamish Online, we use 30 to 60 seconds for both warmup and cooldown times in most of our performance tests. **Specifying bandwidth throttling** Under the **Bandwidth** section of the Settings view, WAS allows you to simulate various network bandwidth ranging from 14.4 Kbps modem connections to T1 (1.5 Mbps) Local Area Network (LAN) connections. The greatest benefit of this feature is the ability to sustain a higher number of concurrent connections on the target Web server. This is a common phenomenon that many Web sites (with customers using slower modem connections) are experiencing. **To enable bandwidth throttling**		

1. From the **Bandwidth** section of the Settings view, select the **Throttle bandwidth** check box.
2. From the drop-down menu, select a bandwidth that represents the connection throughput of most users.

In Duwamish Online, we have tried different settings for bandwidth throttling. Initially, we throttled the users to 56 Kbps connections, in an effort to understand how our application performs under conditions many Web sites experience. We also tried throttling the users to ISDN Dual Channel (128 Kbps) to simulate broadband trends of the future, where most of our target users will access our site via faster connections. Finally, we tested our site with no bandwidth throttling. Interestingly, we found that this setting generates the same load conditions as when using 128 Kbps connections.

No matter what your setting for bandwidth throttling, keep it consistent among all test runs to which you want to compare test results.

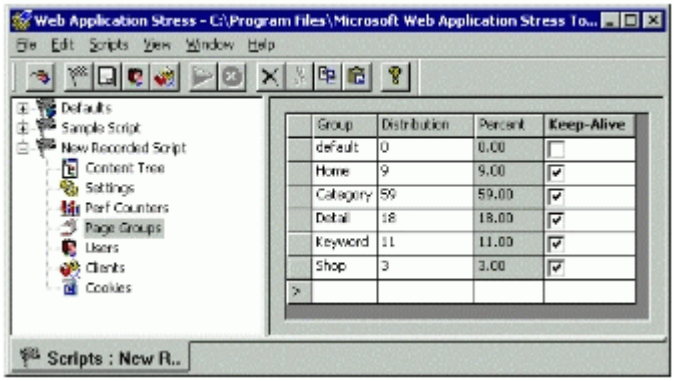
Specifying other settings

Under other sections of the Settings view, we have kept the default values, with the exception of the HTTP redirection. We have purposely disabled this feature by clearing the **Follow HTTP redirects** check box. This is necessary in cases where the redirected URLs have already been recorded in the script during its creation process. You don't want to run those URLs twice in your test run.

Setting up Page Groups

In WAS, you can organize sets of script items into the so-called page groups. This feature allows you to group all page elements (including the HTML files, graphic files, style sheets, etc.) or multiple related pages into one logical unit. You can specify a different hit percentage for each page group, thus controlling which page or related pages would be requested more or less frequently. If you have usage scenarios for your site—like catalog browsing or shopping carts—page groups allow you to run these in the percentage that you expect for your site.

To set up page groups

		1. Expand the information tree under the name of the script in the left pane. 2. Click the **Page Groups** node to open the corresponding view in the right pane. You will find the "default" page group already created with 100-percent distribution in the group table. All script items are by default assigned to this group initially. 3. In the blank row of the group table, type in a new group name (for example, "Home" for the home page) in the **Group** column and a number in the **Distribution** column. The distribution number will be used to calculate the hit percentage of the page group, as shown in the **Percent** column. Repeat this step to add more page groups to the table. 4. Click the script name in the left pane to switch to the view with the script items. 5. In the **Group** column of the script items table, select one of the page groups from the drop-down menu. Repeat this step for each script item in the table. All related pages should be assigned with the same page group.  **Figure 5: Example of page groups definition** See full-sized image. 6. Click the script name in the left pane to switch to the view with the script items.		

7. In the Group column of the script items table, select one of the page groups from the drop-down menu, as shown in Figure 6.

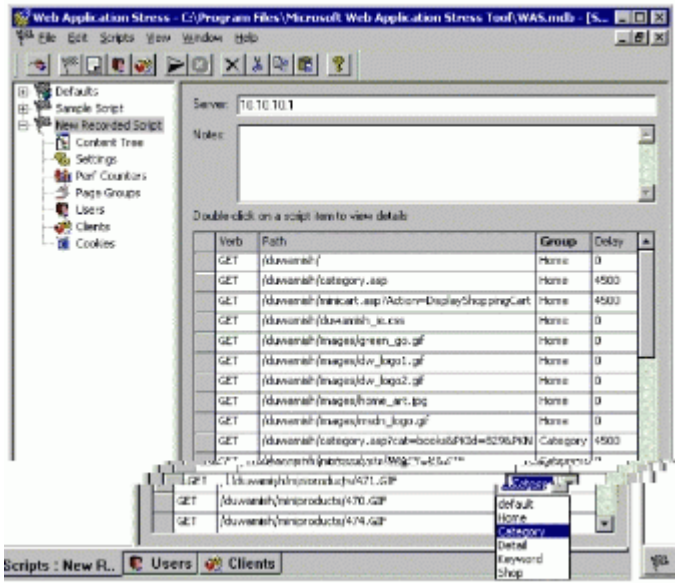


Figure 6: Script Items view screen showing group selection
See full-sized image.

8. Repeat steps 6 and 7 to select a page group for each script item. All related items (such as ASP pages, style sheets, and graphics files) should be assigned with the same page group.

Another way to create and assign page groups to script items is to specify the group during script recording. To use this method, switch back to the WAS program window (see Figure 2) from the browser before jumping to a new

assigned with the new group name.

Setting up Users

For testing Web sites that require personalization and authentication, WAS provides a feature called **Users**, which can be used to store multiple records for user name, password, and cookie information.

When a test starts, all users are divided among the threads given by the stress-level setting. As requests are made, each thread uses the user name, password, and cookie from a new user from the pool allocated to that thread. If WAS has been configured for fewer users than threads, some threads will not have users—any authenticated pages will fail, and any interaction with cookies will be disabled. Therefore, whenever testing personalized Web sites, it is very important to have more users than threads.

There is no imposed limit to the number of users you can create in WAS. However, because each user requires more memory and resources across clients, your tests might take longer to be started and stopped if a huge user list is used.

To create new users

1. Expand the information tree under the name of the script in the left pane.
2. Click the **Users** node to open the corresponding view in the right pane.
3. Double-click the **Default** user group to open the Users view.

   Note that 200 user records have already been created by default. You can simply click on each cell in the user table to modify the **user** and **password** fields according to the logon accounts of your application, if any.

You can also create a new set of users by doing the following:

1. Click the **Remove All** button to clear all existing records.
2. In the **Number of new users** field, enter the number of users you would like to create.

		3. In the **User name prefix** field, type a value that you would like to precede each user's number, for example "User." 4. In the **Password** field, type a password. This same password will be assigned to all users. 5. Finally, click the **Create** button. The user table will be filled with the number of users record as specified. If you would like to use a custom user name and password list, you can import them from a preformatted text file. See the section "Importing user names and passwords" in the WAS Help file for more information on this feature. Setting up Client Computers WAS allows you to use multiple client computers to load test your Web site. When a test starts, WAS will automatically communicate with all the defined clients, transfer all test information (including the script items, page groups, and users definitions) to them, start and stop the test in them, and then collect the test results from them. Use one of the computers as your master client. This should be the client that you have been using to record and configure your test scripts. **To set up clients for your test** 1. Expand the information tree under the name of the script in the left pane. 2. Click the **Clients** node to open the corresponding view in the right pane. 3. Double-click the **Default** client group to open the Clients view. A client record for the localhost (the master client on which you are working) is already created in the client table by default. 4. To add another client into the client table, enter the IP address or the domain name of the additional client into the **Machine name** field. 5. Click the **Add** button and the new client will be added to the table with status of **Connected**. 6. Repeat steps 5 and 6 until all client computers are added to the table. When adding new clients, try to add computers with roughly equivalent		

slower than the other clients can produce more socket errors than not having that computer at all. It appears that WAS does not check the computers' capabilities when distributing its test load. An older, slower client computer will be expected to contribute as much to load generation as a newer, faster one.

We also found that we had better luck setting up a dedicated machine to serve as a master client, but not contribute to load generation. In this configuration, we generated fewer socket errors and tests stopped more quickly.

To set this up, remove the master client's computer name from the list of clients. If you have a slower machine available that you decided not to use for load generation, it could act as your master client without affecting the outcome of the tests. Be aware, however, that the master client still does all the report generation and distribution of test scripts. Having a slower master client will mean your tests are slower to start and that reports might take more time to generate.

Setting up Performance Counters

WAS can integrate with the Windows NT Performance Monitor interface to simplify test data collection. You can store your favorite Performance Monitor counters with each script and WAS will collect their data with the other information it collects.

To add Performance Monitor counters into your script

1. Expand the information tree under the name of the script in the left pane.
2. Click the **Perf Counters** node to open the corresponding view in the right pane.
3. In the **Collection Interval** field, enter the number for collection interval. This is the time, in seconds, between samplings.
4. Click the **Add Counter** button.
5. From the **Add counter to report** dialog box, select the computers, objects, and counters you are interested in collecting by clicking the **Add** button after each selection.

For a list of the common performance counters, please refer to the "Common performance monitor counters" section in the WAS Help file.

If you encounter any problems using this feature, please refer to the WAS

knowledge base.

Running Test Scripts

Once you have set up and configured the test scripts, you are now ready to run the scripts on your client computers.

To start a test run on the master client computer

1. Click the target script name to highlight it.
2. From the **Scripts** menu, select **Run**.

You may also click the **Play** button on the toolbar to run the script.

Examining the Test Report

After the test has finished, you should first examine the test report for any socket or HTTP errors.

To check for these errors from the report

1. From the **View** menu, select **Reports** to open the corresponding view in the right pane, as shown in Figure 7.

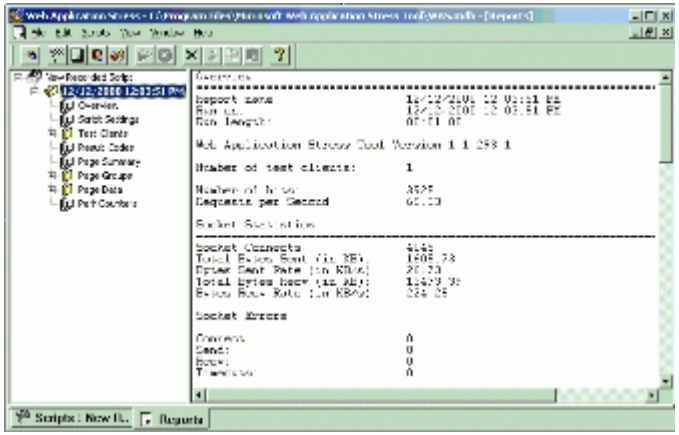


Figure 7: Reports view screen
See full-sized image.

2. In the left pane, double-click the script name to expand the test report tree, if needed.
3. Click the report name specified by the test run time, if needed. You will see the report summary in the right pane.

From the report summary, check under the **Socket Errors** section for any socket-related errors (with values other than zero). Here are the explanations for each type of socket error:

- **Connect**—The number of times any client failed to connect to the Web server. If you have a high value on this error, check on any potential network problems between the clients and the Web server. Ping the IP address of the Web server from each of the clients or telnet to port 80 of the Web server to verify that you have proper response from the server.
- **Send**—The number of times any client failed to send data to the Web server. If you have a high value on this error, check that the Web server is running and functioning properly. Open a browser from the client and manually click through the Web site to verify that the site is functioning as expected.
- **Recv**—The number of times any client failed to receive data from the Web server. If you have a high value on this error, follow the same procedures as for the **Send** error. Also, check if the errors diminish when you decrease the load level.
- **Timeouts**—The number of threads that timed out, and were subsequently closed. If you have a high value on this error, open a browser from the client and manually click through the Web site to verify whether your application is too slow even with just one user. Do a complete load test against the site at a different load level and study the latency characteristic of your application.

4. If the socket errors are very low or zero, scroll down the Summary Report view and find the **Result Codes** section.
5. Check if all result codes are 200, which indicates that all requests have been successfully returned by the Web server. If you find any result codes that are greater than or equal to 400, continue with the following steps to check which script items (URLs) have produced those HTTP errors.
6. Expand the information tree under the name of the script in the left pane.
7. Double-click the **Page Data** node to expand the listing of all script items.
8. Click each script item to view the page data report in the right pane.

		. 9. Check the **Result Codes** section on the page data report of each script item and verify whether any of them have generated the HTTP errors. For a list of all common result codes, please read the "HTTP result codes" section in the WAS Help file. Running the Scripts After preparing the test scripts as just described, you are now ready to run the test and collect data. You may follow the steps described earlier to run each test manually. However, this can be a time-consuming process. WAS exposes an object model that allows you to create your own Microsoft Visual Basic Scripting Edition (VBScript) script to control and configure test runs. Go to http://www.microsoft.com/technet/archive/itsolutions/intranet/downloads/webstres.mspx, where you will find some VBScript samples that might help you run your performance tests more efficiently. For example, we used a modified version of script7.txt to automate our load test process. While the test is being run, you should monitor and record various performance-related system counters, including counters to track system throughput, latency (response time), and resource utilization. Conclusion: Best Practices **Client computers.** Monitor the system utilization of each client closely. If the CPU or memory usage is above 80 percent, the clients are probably overloaded and you should consider using more client computers for your tests. Stressing the client computers will cause unreliable results and produce socket errors on connection to the Web servers. **Set multiplier to stress the servers.** Estimate the maximum number of concurrent user requests required to push your Web server farm to 100-percent utilization in a pretest. When the server farm needs to be stressed out without a sufficient number of test client computers, a higher multiplier might be needed. For example, if you found that with 4,000 WAST threads, all with a multiplier of 1, you still could		

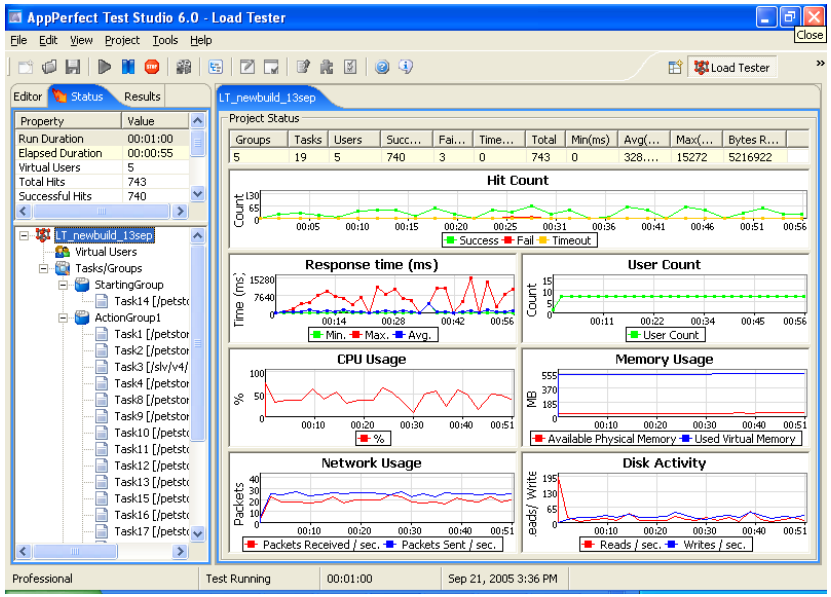
		not stress out your server, you could use the multiplier to increase the stress. However, using a multiplier greater than 1 will not accurately reflect the correct TTLB of the Web applications pages. If possible, add more client computers to your test environment rather than rely on the user multiplier. **Use SessionTrace.** Use SessionTrace to record detailed communication between WAS and the Web server(s). When defining a new WAS script, it is important to find out if all URLs used in the script are functioning as expected and the Web server is returning the desired response. If not, there is a possibility that you might obtain improved performance results while the Web server is simply returning error response. You should set SessionTrace to 1 with type REG_DWORD. SessionTrace can be turned on in Registry \HKEY_LOCAL_MACHINE \SOFTWARE \Microsoft \WAS. Finally, remember to turn SessionTrace off (0) after validation of the new script. Otherwise, the disk space will quickly become full. **Monitor Web server log files.** Be prepared to relocate or purge your Web server log files. They grow very fast with lots of performance testing, especially for long-running tests. You can also use the log files to help troubleshoot application errors reported by WAS. **Limit the number of script items and users.** Avoid creating more than 1,000 script items or users unless there is a specific reason that you require more of these objects. Although their allowable numbers are only limited by the amount of memory on the client computer, you might find that it takes too long for a test to initialize when using a large number of script items or users. **Follow HTTP Redirects option.** Do not use this option if the script has already recorded the redirected URLs. If you use this option, the redirected pages will be counted twice. **%USERNAME% and %PASSWORD%.** The WAS Help file refers to these entries as a way to fill in forms using the USERNAME and PASSWORD values specified for each WAS user. In our testing, using %USERNAME% and %PASSWORD% dramatically increased the amount of time required to shut down a script distributed across multiple clients. On advice from other internal WAS users, we specified USER and PASSWORD name-value pairs in the *QueryString* portion of each WAS test. By setting the access method to sequential, for both USER and PASSWORD, we guaranteed that the passwords always corresponded to the appropriate user.		

		Despite some of these limitations, WAS is a great tool for simulating users on your site before you go live. Using a performance testing tool is extremely important to a successful application launch. These tools allow you to understand your application's performance characteristics, so you will know exactly how the application will perform when placed under high user load. The fewer surprises that you have while operating your Web site, the better your Web site's reliability.		

VENDOR:		***APPPERFECT***		
TOOL NAME:		**LoadTester [Free]**		
Web Page		**http://www.appperfect.com/products/devsuite/lt.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Overview AppPerfect Load Tester is a highly scalable load simulation software that helps you test your application's behavior under conditions of heavy usage. It is used primarily for load testing and capacity planning. Scenarios such as large number of concurrent users, large number of URLs, sustained usage over hours/days, etc. can be tested using the AppPerfect Load Tester. AppPerfect Load Tester is easy to use with no programming required. Its provides a unique view of the target machine with integrated loading testing parameters such as response time and hit count along with monitoring of system resources such as CPU, disk, network and memory.	Free	

Screenshot

Current Status view



More screenshots:

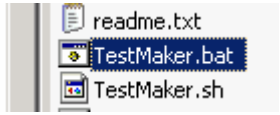
1. Test Editor
2. Test Results Summary
3. AppPerfect Analytics - Database Schema
4. Running inside Eclipse as a plug-in

Key Features

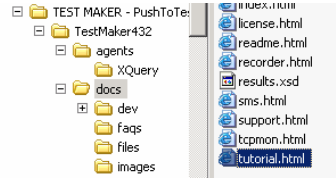
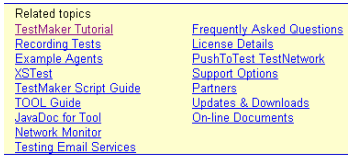
- Scalable architecture to simulate thousands of virtual users
- Easy to use browser-based recording of test scripts
- Support for simulating exact browser behavior for more realistic testing (MSIE and Mozilla/Firefox simulation)
- Support for cookies, basic authentication, URL re-directing

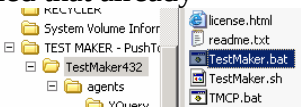
		and SSL • Support for grouping and parameterization of URLs, including dynamic parameterization (source is database, text file, etc) • Support for extensive validation based on return code, returned html code, etc. • Comprehensive reports including custom report designer and export of reports to PDF, Excel and HTML • Full IDE integration for Eclipse, NetBeans, IBM WebSphere Studio, IntelliJ, JDeveloper and JBuilder • Integrated monitoring of target machine's system resources ## Supported Products • JDK : JDK 1.3.1, JDK 1.4.1, JDK 1.4.2, JDK 1.5 • OS : Windows 2000/XP/2003, Linux x86, Solaris 8,9, Mac OS X • App Server : BEA WebLogic 7.x/8.x/9.x; IBM WebSphere 4.x/5.x; Apache Tomcat 4.x/5.x; JBoss 3.x • IDE : Eclipse 2.1.x/3.x; NetBeans 4.0/4.1; IBM WSAD 5.1, IBM RAD 6.0; IntelliJ 3.x/4.x/5.x; JBuilder 9.x/10.x ## Price and Availability AppPerfect Load Tester is a component of the AppPerfect Test Studio. AppPerfect Load Tester is not sold separately. The AppPerfect Test Studio is currently available. AppPerfect Test Studio is shipped in two different editions: Standard and Professional. The AppPerfect Test Studio - Standard Edition is free for use, it has no license fees. The AppPerfect Test Studio - Professional Edition is licensed for US $495 per user. Click **here** for details on feature differences between the two versions. Visit **AppPerfect Sales** for purchasing information and options. One license allows a single user to install and use the software on a single machine. ## Download		

		Latest information about AppPerfect Load Tester is available at http://www.AppPerfect.com/products/devsuite/lt.html. A free 15-day evaluation version of the AppPerfect Test Studio - Professional Edition can be downloaded from **here**. At the end of 15 days, the software		

VENDOR:		***PUSHTOTEST***		
TOOL NAME:		**Performance Kit [Free]**		
Web Page		**http://www.pushtotest.com/Downloads/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		PushToTest offers TestMaker, a free open-source utility and framework, that software developers, QA technicians, and IT managers use to check Web-enabled applications for functionality, scalability, and performance. Documentation and articles on TestMaker are available at docs.pushtotest.com. Free open source utility maintained by PushToTest.com and Frank Cohen, for performance, scalability, and functional testing of Web application. A framework and utility to build and run intelligent test agents that implement user behaviors and drive the system as users would. Features an XML-based scripting language and library of test objects to create test agents. Includes capability to check and monitor email systems using SMTP, POP3, IMAP protocols. Java-based tool - runs on any platform. Description: **TestMaker from PushToTest.com delivers a rich environment for building and running intelligent test agents that test Web-enabled applications for scalability, functionality, and performance. It comes with**	FREE OPEN SOURCE for unlimited users $10,000 for Enterprise edition	This is amazing that they are letting us use a $10,000 tool for free even in a scaled down version. **DOWN LOAN** Very simple to down load **INSTALLATION:** Not so simple. Can't find the Icon to start the application. It was not on the desk top. I tried the .BAT file 

		a friendly graphical user environment, an object-oriented scripting language (Jython) to build intelligent test agents, an extensible library of protocol handlers (HTTP, HTTPS, SOAP, XML-RPC, SMTP, POP3, IMAP), a new agent wizard featuring an Agent Recorder to write scripts for you, a library of fully-functional sample test agents, and shell scripts to run test agents from the command line and from unit test utilities. Requirement: **Java 1.4 or higher virtual machine on Windows, Linux, Solaris, and Macintosh.** The Performance Kit A comprehensive overview of web service performance factors. Includes a complete test environment to check scalability and performance of applications running on IBM WebSphere, BEA WebLogic Server, BEA WebLogic Workshop, and the SunONE Application Server. Enterprise information manager choices for Web Service infrastructure are critical to deploying well performing, highly reliable and scalable systems for running Web Services. We receive calls everyday from software developers, QA analysts and IT managers wanting to know: What are the most important considerations to build reliable Web Services? What equipment do I need? What is the best way to deploy Web Services? What is the best software platform to build my Web service? The Web Services Performance Kit helps information managers understand the differences between the Sun Java System Application Server (formerly Sun One,) BEA WebLogic Server and IBM WebSphere Application Server. The Performance Kit features detailed step-by-step instructions to configure a Test Web Service on each server, plus a unique Web Service test utility called TestMaker to show the actual performance of each server. Inside The Performance Kit you will find these things: Easy-to-read instructions describing how to build and test Web Services on the Sun Java System Application Server (formerly Sun One,) BEA WebLogic Server and IBM WebSphere Server. Test Web Service (TWS) - a free open-source Web Service that simulates a database-driven Web service. Performance_Agent - a free intelligent test agent that will determine TWS scalability and performance. These metrics are key to building the right datacenter and having happy customers. TestMaker - a free open-source utility to run Performance_Agent and to create agents for your own Web service testing. The Performance Kit is free.		A DOS black box came up with this information: (enlarge to 200% to read)  **README:** Information on "Java Testing and Design: From Unit Testing To Automated Web Tests" by Frank Cohen from Prentice Hall Publishing. In many ways this 489-page book is the missing manual for TestMaker. Java Testing and Design is filled with our favorite tips, techniques and example code. Details are at: http://thebook.pushtotest.com We recommend you learn about TestMaker by reading the documentation found in the doc directory. Additional assistance is available from the Help drop-down menu. Try running the sample test agents that come with TestMaker in the agents directory. The latest information, software, updates and agents are available on the http://www.PushToTest.com web site. See the doc/faq.html for common installation problems and answers. Lastly, send an email to users@lists.pushtotest.com if you need help from the TestMaker community. TestMaker requires Java 1.4 or higher. - The TestMaker Team '''

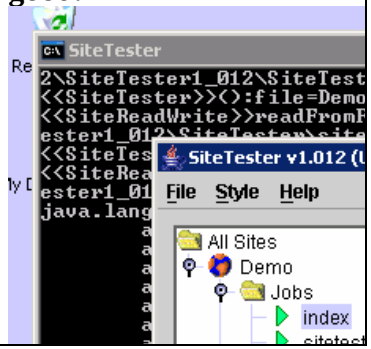
				Surely it cant be this hard to just launch the application??? After much searching I found a tutorial  I wonder if one of the FAQ's is HOW DO YOU LAUNCH THE APPLICATION  To run TestMaker: • Windows users run the testmaker_home /TestMaker.bat file. This file executes a simple shell script that configures the

				local classpath and launches the TestMaker application. I tried that already  And got the black DOS window saying I have to install JAVA which is already installed on my server. **NEXT STEP:** Call the system engineers and let them use it. OR.. pay for the $10,000 edition that has a user manual.

VENDOR:		***PILOT SOFTWARE LTD (TURKEY)***		
TOOL NAME:		**SiteTester [$29.00]**		
Web Page		**http://www.pilotltd.com/eng/sitetester/**		
Web	**Functional**	**Feature pro/cons**	**Aprox**	**Our Evaluation / Rating**

Testing	Testing		Pricing	
		SiteTester is a load-testing utility designed to test web servers and web applications. *SiteTester* simulates concurrent access of various users from different hosts to a web/application server. Each virtual user follows a predefined procedure to access the server. By increasing the number of virtual users you can test the capacity of your servers and determine the extra hardware and bandwidth needs, if any. Many web applications fail to operate on simultaneous uses, despite every effort to test and debug the application on the development environment. This is mainly due to the nature of multi-threaded programming, which makes it difficult to simulate the actual process. Application codes, which are not multi-thread-safe, which leak memory, which may produce deadlocks, have great affect on the performance and stability of the web sites. *SiteTester* allows web application developers to test their codes amongst many users, and take precaution before the actual deployment occurs. Load test tool from Pilot Software Ltd. Allows definition of requests, jobs, procedures and tests, HTTP1.0/1.1 compatible requests, POST/GET methods, cookies, running in multi-threaded or single-threaded mode, generates various reports in HTML format, keeps and reads XML formatted files for test definitions and test logs. Requires JDK1.2 or higher. An easy-to-use tool to test web servers and web applications against stress. Are you looking for a tool that let you simulate exactly, what hundreds of users doing on your servers? Are your web application codes really tested for full load, before you actually deploy them? Try *SiteTester*, the newest tool to test server performance easily... *SiteTester1.0* is a JDK1.2 compatible Java application and now is available for the Windows and UNIX environments. *SiteTester* is a load-testing utility designed to test web servers and web applications. *SiteTester* simulates concurrent access of various users from different hosts to a web/application server. Each virtual user follows a predefined procedure to access the server. By increasing the number of virtual users you can test the capacity of your servers and determine the extra hardware and bandwidth needs, if any. Many web applications fail to operate on simultaneous uses, despite every effort to test and debug the application on the development environment. This is mainly due to the nature of multi-threaded programming, which makes it	$29.00	OUR EVALUATION: **1. LOCATION ADDRESS:** Pilot Yazilim Buklum sk. 20/20 Kavaklidere(06660)/Ankara Turkey. **Download is simple. However the install is not for sissies. It does have step by step instructions (if you can understand them). They are pretty technical. You may need a Systems Engineer to help you do the install.** **2. INSTALLATION** Installing the *SiteTester* utility is a simple process of extracting the zip archive to some directory on your machine, with execute and write rights. *SiteTester* is a Java application, compliant with the JDK1.2 or later, so a java virtual machine (JVM) is required to run this software. If not already installed, you can download and install the

		difficult to simulate the actual process. Application codes, which are not multi-thread-safe, which leak memory, which may produce deadlocks, have great affect on the performance and stability of the web sites. *SiteTester* allows web application developers to test their codes amongst many users, and take precaution before the actual deployment occurs. **Benefits of using *SiteTester*** Test how many concurrent users can your server handle. Learn the average load time of each page, when the server is busy. Test if your CGI, Perl, JSP, ASP, PHP or other web application codes do support multi-threading. Test if your application codes produce dead-locks. Check whether the application codes degrade performance when called hundreds of times. Determine the hardware and baudrate needs for the number of online users requirements. Determine and tune the DBMS performance, if any. **Features** Allows definition of requests, jobs, procedures and tests. Allows HTTP1.0/1.1 compatible requests. Allows POST/GET methods. Allows user-defined pauses between jobs, for each virtual user. Stores and handles cookies.		JVM from Sun Microsystems. **3. RUNNING SITETESTER** Before running SiteTester you have to edit the *SiteTester.bat* file (or *SiteTester.sh* on UNIX). Point the JAVA_HOME parameter to the root of the JVM (e.g. C:\jdk1.3.1_01\jre). In order to launch the application on windows, double-click *sitetester.bat*. On UNIX, execute "*sh ./SiteTester.sh*" command from the shell. The application frame should open when the scripts(s) succeed to run. ~~~~~~~~~~~~~~~~~~ **It took some study to learn how to do all of that. It is not just intuitive for a novice where you can just do it. We had to do the JAVA down load. In order to make the install we had to ask our systems engineer to help. He**

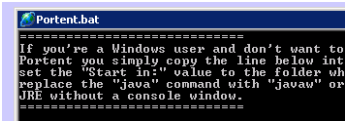
		Allows applying the same test(s) on different occasions. Displays test progress visually. Allows running each job, in multi-threaded or single-threaded mode. Allows repeating the procedure any number of times, for each virtual user. Generates various reports in HTML format. May gather and analyze data, collected on different hosts. Keeps and reads XML formatted files for test definitions and test logs. **Specifications** OS: Windows95/98/NT/2K/XP, UNIX Variants. Java Runtime: JDK1.2 or higher. RAM: 64MB or more.		**finally figured out the reason it was not working is that the path to the JVM had to be the 8 character DOS file name format so in the .bat file he changed the path from JAVA_HOME=C:C:\Program Files\Java\j2re1.4.2_05 to JAVA_HOME=C:\Progra~1\Java\j2re1.4.2_05 Then the install worked. It has a good demo program to help you learn how to use it.** **The DOS window stays up while running the program. That is not good.** 

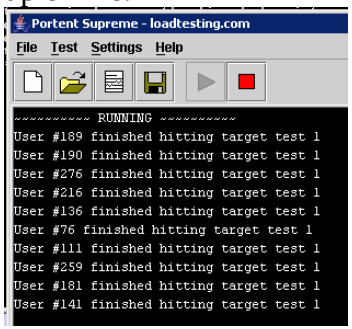
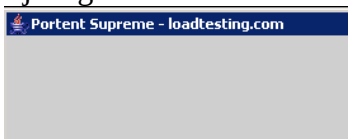
VENDOR:		***SOFTLOGICA LLC (Russia)***		
TOOL NAME:		**WAPT [$299]**		
Web Page		**www.loadtestingtool.com**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
YES	NO	Web load and stress testing tool from SoftLogica LLC. Handles dynamic content and HTTPS/SSL; easy to use; support for redirects and all types of proxies; clear reports and graphs. **Accurate Real User Simulation** WAPT provides accurate simulation of web server load created by real users. Each virtual user participating in the test is independent from other ones. It can have its own cookies, data and all other parameters. Randomization of delays between page hits and user connection speed emulation let you create realistic workload against tested server. WAPT provides proxy header substitution to emulate activity of multiple users coming from different computers. **Dynamic Test Data Generation** WAPT has excellent abilities of run-time test data generation. The values of request parameters and URIs can be calculated in a number of ways. They can even be determined by server response to the previous request. **Recording and Playback of HTTPS/SSL Pages** WAPT supports all security features related to HTTPS/SSL. You can both record and play back requests to secure content. Testing of sites protected with basic authorization is also supported. **Flexible Test Volume & Load Definition** You can specify how to change the number of virtual users and other parameters during the test. You can set WAPT to perform multiple iterations of	$299 for unlimited users	We tested this tool extensively. We wrote 60,000 lines of code and ran the tests for 2 weeks. It is an excellent tool and worked very well against our application. The vendor is in Novosibirsk, Russia  but we were able to call them on the phone (6 pm here in Seattle is 8 a.m. there in Russia) so we were able to coordinate with them by phone.

		test sequence and multiple test runs. You can specify the duration of test or the number of hits to perform. You can also limit the load level to a fixed number of hits per second. **Clear Reports and Graphs** Test results are represented by descriptive graphs and reports to give you all the detailed information about web site performance in specified load conditions. Text reports are saved in Microsoft Excel compatible format. **Full Log of Virtual Users Activities** WAPT creates complete log of virtual users activities for each test, including all requests and server responses. You can set WAPT to log only error responses. **Intuitive user-friendly interface** You can get familiar with basic WAPT features and start using it in a few minutes after the installation. Test creation is very simple and based on recording your own actions while you browse a web site in the Microsoft Internet Explorer. When you finish recording you can edit the recorded sequence to fully adjust it to the needs of your test. New scenario wizard assists to adjust the key parameters according to your current test objectives. **Other Features** Support of all types of proxy servers: HTTP(S), SOCKS4(5). *Scheduled run* option allows to start testing at specified time. Ability to add custom strings to *HTTP headers*. Ability to test multiple web servers simultaneously. Support of redirects, including redirects to another server.		They speak very good English. They also hooked us up on ICQ5 chat room where we were able to chat live with them. **ICQ Number:** 206-101-760 **Name:** Gras Alex **NickName:** gav Using that tool, we sent them literally 100's of suggestions for improvement and they actually listened and changed the software. We were very impressed with their responsiveness to issues. Then they came out with a new version that incorporated all our suggestions and much more. Version 4.0 is much improved over the original Version 3.0 that we started testing with. We were more than pleased with this tool and it did everything that we hoped it to do. The amazing thing is that it is only $300. When I asked them about that, they said that they wanted it to cost about the same as a

				common Microsoft application like purchasing a copy of Excel or Word. They certainly have done that. It is worth 10 times that. You can have unlimited virtual users. There is no annual maintenance fee. The customer service is 210%. The application is very easy to install. It is very easy to learn. The coding is not difficult to learn yet if you want to get very technical there is that capability as well to do your own programming. They sent us a free upgrade when we went from 3.0 to 4.0 The charts and graphs for the test results were very easy to read for our managers. This is the tool that we decided to go with after all our evaluations and we were not disappointed.

VENDOR:	***LOADTESTING.COM***
TOOL NAME:	**Portent Supreme [$279.95]**

Web Page		www.loadtesting.com		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Loadtesting.com's low-priced web load testing tool. Has minimal hardware requirements. Page validation via matching string in page. Written in Java; multi-platform. A new version will be coming out this summer that will have many more new features and more competitive with other tools and will still be under $500 User-Centric Load Testing Software While Portent Classic is good for creating large loads of HTTP GETs on a web server, Portnet Supreme includes features that allow for testing user-based scenarios when load testing. Portent Supreme is a great upgrade from Portent Classic for those that wish to have more realistic user simulation when performing load tests of their web site. With Portent Supreme QA and testing engineers have all of the tools necessary to create test plans that reflect real user experiences and simulate those scenarios on their web site during the development and testing process. Improved Support of the HTTP protocol To more accurately simulate real user experiences while load testing, Portent Supreme handles most of the browser and web surfing experiences a user would face when encountering your web site. The features we've added to increase load testing of real user experiences are: Increased support of HTTP methods such as **GET, POST, HEAD, PUT, DELETE and TRACE** Ability to define standard **HTTP headers** and custom headers to your HTTP target requests **Full CGI support** across all HTTP methods **BASIC authentication** for simulating user authentication against a domain	$279.95 for unlimited	The download was not hard to understand but the install is not intuitive at all. Very difficult to understand. No step by step instructions. RUNNING This black DOS screen stays up all the time. That is irritating. It says you can turn it off but the instructions are quite technical for a normal user  Portent.bat ========== If you're a Windows user and don't want to Portent you simply copy the line below int set the "Start in:" value to the folder wh replace the "java" command with "javaw" or JRE without a console window. ========== Here is what it says in the black box (to give you an example of what I mean about it being very technical: If you're a Windows user and don't want to see the DOS window when you run Portent you simply copy the line below into a Windows shortcut. Make sure you set the "Start in:" value to the folder where Portent is installed. Also, replace the "java" command with "javaw" or "javaw.exe" which will start the JRE without a console window.

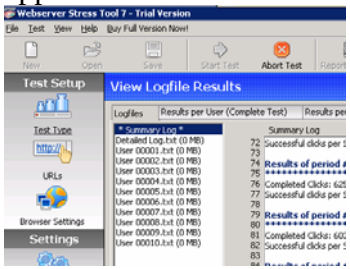
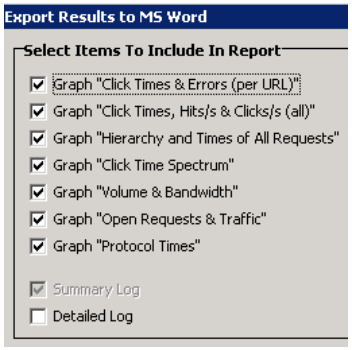
		with a given login/password combination **Cookie handling** and supression since most sites today use cookies to enhance the user's experience **SSL encryption**, commonly used for e-commerce web sites Download of **attached or embedded media** such as images, Flash, and other content that is part of the resultant HTML document returned by the web server Dynamic handling of **HTTP redirects** (status 3xx), commonly used for banner ads, affiliate programs and other web-based marketing features Resultant **HTML can be validated** and the status of the validation is reported. **NEW Proxy Server support** - HTTP connections can now be established through proxy servers, supporting authentication and proxy exclusions for local network URLs. Enhanced Test Plan Creation, Management and Execution Test plans are created with a GUI editor inside of Portent Supreme. Each test plan can be saved to a file and restored for execution at a later time in both GUI and console mode. Test plans are a group of targets that represent the "visits" to your site that you would like the virtual users to visit. You may specify the order in which they are visited (sequential, reverse, random) and think time between hits (see below). Test plans execution can be defined to end after a certain amount of time, a certain number of hits, a number of times the test plan is iterated through, or		For me that language is too technical for a common user. Tool seemed quite easy to learn at first and the instructions in help are quite intuitive. But when I tried to run my first test the system locked up on me.  I just got a blank screen  And had to reboot the server to get out of the application and start over. Don't hold this against the tool itself. I was trying to learn to run it with out reading any instructions. Over all it looks like a very powerful tool if you

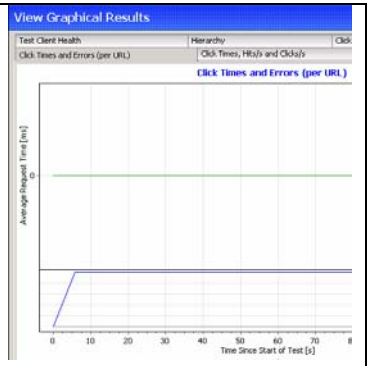
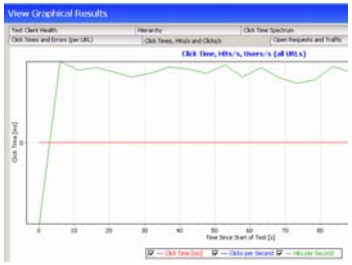
		indefinitely until the user stops the test. Page Transitions and "Think Time" When a user visits a site they usually experience some "think time" before going on to the next page. This is usually due to the fact that the user needs time to read the page, or maybe they had to step away from the computer for a moment. These delays between page visits can be simulated with the creation of "transitions" when creating a test plan. The transition can be defined to be a fixed amount of time. Or, a random delay can be created by specifying a minimum and a maximum delay to pause between hits. User-Defined Error Checking When hits are generated and the results are downloaded you can report errors if the results do not contain a certain matching string. This will enable you to check the context of the hits on the site, provided an added level of functional testing to the test plan. Reporting Features Test results can be exported to a "|" delimited file. For each hit generated the data preserved include the amount of time it took to connect to the site, to download the results, number of bytes downloaded, protocol and user-defined errors encountered, and other relevant data. This data is easily imported into a spreadsheet program for analysis. Portent Supreme also includes a "Results" tab for a GUI display of results by user and by target. This is a useful analysis tool when considering where you system bottlenecks may reside. Incredible cost-effectiveness, ROI While most load testing tools START at US$10,000.00 just for basic load testing tools, you can see Portent Supreme is a very cost-effective way to load test your site. The ability to ensure that your site can handle thousands and millions of visitors is priceless information to your e-business requirements. With Portent Supreme, you get robust capabilities at a very affordable price.		have the technical skills to learn to use it. And certainly the price is right to make it worth the time to learn. In talking with the Salesman from the vendor they said a new version would be out in October but I have not hard from them again (writing this October 5th 2005). They hopefully will take our suggestions and make them enhancements in the new version. The price of the new version will still be under $500 according to the salesman. Worth taking a look at.

		The Most Open, Cross-platform Solution on the Market Using Java™ "write-once, run anywhere" technology, Portent Supreme is the only load testing tool that can be run on a dozen operating systems, all with the same software license. This allows you to leverage as much hardware that you have available no matter what platforms the run on. For a detailed list of features and comparison to other Portent products please see our comparison page. portent supreme View an on-line demo of Portent Supreme Download a free demo of Portent Supreme **Purchase Portent Supreme software US$279.95 BuyNow**		

VENDOR:		***PAESSLER (Germany)***		
TOOL NAME:		**Webserver Stress Tool 7 [$624.95]**		
Web Page		**www.paessler.com**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Web stress test tool from Paessler GmbH handles proxies, passwords, user agents, cookies and ASP-session IDs. Shareware. For Windows. Standard, Professional, and Enterprise versions. Web Server Performance-, Load-, and Stress-Test	$624.95 for one user license for 10,000 virtual users	We ordered this one in July 2005 and it has not come yet (October 2005). I just wanted to wait and see how long before they

		What is it? Webserver Stress Tool is a powerful HTTP-client/server test application designed to pinpoint critical performance issues in your web site or web server that may prevent optimal experience for your site's visitors. What does it do? By simulating the HTTP requests generated by hundreds or even thousands of simultaneous users, you can test your web server performance under normal and excessive loads to ensure that critical information and services are available at speeds your end-users expect. Detailed test logs and several easy to read graphs make analyzing results a snap. Webserver Stress Tool for Windows (98/ME/2000/XP/2003) can benchmark almost any HTTP server (e.g. static pages, JSPs/ASPs, or CGIs) for performance, load, and stress-tests. How much load can Webserver Stress Tool generate? We have successfully tested Webserver Stress Tool 7 with more than ~500 MBit/s network load, more than 1.000.000 pageviews per hour and up to 10.000 simultaneous users See our Sample Performance Tests section for more detailed test reports. The actual load you can achieve is highly dependent on your network infrastructure, your server/client hardware, the file sizes, and your web application. Why should you use Webserver Stress Tool? Using Webserver Stress Tool when developing and running Websites is important for your web infrastructure: **Maximize Uptime:** Webserver Stress Tool ensures that critical issues in your		would contact us if we did not get the software. We sent them a purchase order and still have not heard back from them. We very much want to evaluate the full version of this tool. The description sounds like it is well worth the $625 but we have not heard back from them. I should not have to call them to see why they did not fill our order. In our presentation next year we will have evaluated this tool for you. We tested the demo version which is limited to 10 virtual users and it looks like it is the powerful tool it is advertised to be. Easy to learn and the help files are very useful. Was easy to install. We did not use the customer service yet but from the web page it looks like it is very well staffed. They even have a newsletter to keep users informed.

		website are resolved before they bring down your web resources **Maximize Performance:** Webserver Stress Tool ensures that your websites and applications are given the server resources they need when they need it to guarantee a high quality user experience **Maximize ROI:** Webserver Stress Tool ensures that you are getting everything out of the investment in your webserver technology through consistent and in-depth testing and analysis. **Maximize Value:** Webserver Stress Tool is the most cost-effective solution in the market for simulating performance, load, and stress tests for your web server. **Minimize Efforts:** Webserver Stress Tool installs in 5 minutes and works with all web servers Key Features Basic Functionality Webserver Stress Tool simulates large numbers of users accessing a website via HTTP/HTTPS. The software can simulate up to 10.000 users that independently click their way through a set of URLs. Simple URL patterns are supported as well as complex URL patterns (via a Script file). How much load can Webserver Stress Tool generate? We have successfully tested Webserver Stress Tool with up to 500 MBit/s network load up to 1.000.000 pageviews per hour and up to 10.000 simultaneous users See our Sample Performance Tests section for more detailed test reports. The actual load you can achieve is highly dependent on your network		http://www.paessler.com/support/newsletter/ The reporting is very clear with good graphs and statistics. Lets you watch the progress as the test run. Was easy to program the test. Worked well with our web based application.   The volume of test reports were impressive. Easy for management to understand.

		infrastructure, your server/client hardware, the file sizes and your web application. Here's how it works Webserver Stress Tool simulates independent users stepping through a set of URLs or through URLs that you can specify using a VB script. Based on parameters you specify, the application not only requests the HTML of a URL but also frames, images, flash files, etc. emulating the same behaviour a web browser would show when accessing the website. Each user is simulated by a separate thread with his own session information (i.e., cookies for each simulated user are stored separately) and "surfs" the URLs independently from the other users – just like in real-world Web usage. URLs can be parameterized for each user and the sequence of URLs can be varied. POST and GET requests are supported as well as BASIC HTTP Authentication and several other settings. With the new scripting functionality you can even create highly complex URL patterns for large scale web applications. Supported Test Types Webserver Stress Tool supports a number of different testing types. For example **Performance Tests**—this test queries single URLs of a webserver or web application to identify and discover elements that may be responsible for slower than expected performance. This test provides a unique opportunity to optimize server settings or application configurations by testing various implementations of single web pages/script to identify the fastest code or settings. **Load Tests**—this tests your entire website at the normal (expected) load. For load testing you simply enter the URLs, the number of users, and the time between clicks of your website traffic. This is a "real world" test. **Stress Tests**—these are simulations of "brute force" attacks that apply excessive load to your webserver. This type of "brute force" situation can be caused by a massive spike in user activity (i.e., a new advertising campaign).		  This tool is well worth the money asked in our opinion.

		This is a great test to find the traffic threshold for your webserver. **Ramp Tests**—this test uses escalating numbers of users over a given time frame to determine the maximum number of users the webserver can accommodate before producing error messages. **Various other tests**—working with Webserver Stress Tool simply gives you more insight about your website, e.g. to determine that web pages can be requested simultaneously without problems like database deadlocks, semaphores, etc. Reported readings Webserver Stress Tool provides a number of robust and flexible reports and logging features to help you gather valuable data about your webserver and website performance. Reports include: Click Time—the time it took for a user to complete a pageview (i.e., loading a page and all associated images/objects). This is the time between clicking a link and end of the http last request – which would be the moment a user would see the complete page. Time for DNS—the time it takes for a client to resolve a URL's domain name using the client system's current DNS server. Time to connect—time to set up a connection to the server. Time to first byte (TFB)—time between initiating a request and receiving the first byte of data from the server. Request Time (TLB, Time to last Byte)—time for a single HTTP request (i.e. HTML page, image, frameset, etc.). Server Bandwidth—the bandwidth achieved by the webserver. User Bandwidth—the average bandwidth that was available for each user. Sent Requests—the number of requests sent to the server during a period. Received Requests—the number of answers received from the server during a		

		period. Open Requests—the number of open requests for a given moment. Error rates—the number of failed requests per time period, per user, or per URL Reporting and Logging This stress and load test tool provides graphs and data in a number of different formats including Several easy to use graphs (visit the sample tests page for sample graphs) Summary Text Log Detailed Text Log User Text Log (one for each user) Machine readable CSV files for the request log and the raw graph data Other Features Webserver Stress Tool also offers a number of other robust, flexible features to provide you the most comprehensive webserver stress and load test tool on the market. These features include Built-in report generator—reports can be generated as HTML files and MS WORD documents Includes a URL recorder (complete web browser) to select the URL(s) you want to test Works on any HTTP or HTTP(s) URL Can test any script—CGI, ASP, PHP, etc.		

		Can also be used to test requests of larger download files (e.g. ZIP) Works with any webserver (no part of the software has to be installed on the server) Includes support for proxies (not for HTTPS) with optional proxy authentication basic user authentication (username/password) user agent string any HTTP request header lines individual cookie handling for each simulated user (e.g. ASP-Session-IDs) redirected requests and " http-meta-refresh " redirections several IPs for the client machine (up to 5000 IPs) data rate throttling (e.g. to simulate users accessing the server via a slow modem line) timeouts (e.g. to simulate surfers that click away after 20 seconds without answer of the server) When testing more than one URL several URL selection methods can be selected to simulate different user behavior Tests can run until a specified number of clicks is reached for each user until a specified time has passed Tests can be started at a specified time		

		Optional link checker can check all URLs for validity Optionally all received HTML pages can be written to files Optionally all resulting files can be archived in a ZIP file for each test for later reference		

VENDOR:		***SIS –Florida Tech***		
TOOL NAME:		**Holodeck [$1,500]**		
Web Page		**http://www.sisecure.com/holodeck/what_is.shtml**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Holodeck is NOT a load testing tool. It does NOT simulate users. Holodeck is a fault injection tool. It simulates the effect of depriving a system of resources such as memory, etc. Fault injection is more accurate and repeatable than load testing. What Is Holodeck (and why was it developed?) Versatile and Powerful Reliability, Debugging and Fault-Injection Testing Tool. Holodeck is the first commercially available fault-simulation tool and was developed by leading researchers in the application quality field. It is ideal for testing any Windows Application, Windows Service or Web Service, regardless of what language it is written it. Holodeck is a dynamic analysis tool and requires no source code, allowing you to authentically analyze the interaction of the application with its environment. It is a filemon, regmon, netmon, processmon, libmon, and apimon all in one	$1,500	You have to understand that this is not a normal LOAD testing type tool. This is a "FAULT INJECTION" testing methodology. Of all the tools we looked at this is the top of the line. It costs a little more than the others but you more than get what you pay for. We would highly recommend this tool for system engineers who want to

		user interface, so you can scope by process or process and thread. This allows you to understand your application better than you could otherwise because it can very hard to see all the dependencies and interactions an application has. Simply viewing or testing source code does not accomplish this objective or give you realistic insight. Holodeck can be used in multiple ways, including: In conjunction with other Automated Test Frameworks to analyze and debug error handlers Running as a standalone to inject faults in an application Executed from the command line using scripts and batch files to automate the testing of applications Plugging the QA holes. Holodeck was designed to plug the gaps in existing testing methods -- the software QA practices that let tens of thousands of bugs currently slip through every day. It utilizes fault-injection techniques to introduce the application to simulated situations and scenarios that arise from the result of "broken" environments, such as bad memory conditions, corrupt files, or other "attacks" to exercise error-handling routines. Holodeck is an ideal functional and reliability test tool for testers and GUI developers who need superior control for testing Java®, Microsoft® Visual Studio, .NET®, and Web-based applications. Its powerful environment simulation and fully programmatic design offers multiple extensibility points for deeper testing - giving you the ability to test the limits of your application. Holodeck can also be used as an investigative tool to perform ad-hoc, stress, and code-coverage testing. You can customize the faults, based on how your application is designed. The more you use Holodeck, the more you'll know about your application and be better equipped to test. Additionally, you'll get answers to key questions such as the following: How can I replicate my customers' environment in my test lab? What files does my application touch/need when opening? operating? ...what happens if I corrupt those files or deny my application access to them? What specific exception/error caused my application to crash?		simulate testing where you do things like reducing the band width, memory or disk space until the system fails. We also purchased the $300 training package but as of this time we have not had time to take the training. We will have a more complete report on this tool and the training in our presentation at the PNSQC conference next year. We also recommend James A. Whittaker's two books, "*How To Break Soft Ware*" and "*How To Break Software Security*". You get a scaled down version of Holodeck on a CD with the book where you can try it out. www.howtobreaksoftware.com

		What happens if I starve my application of resources or deprive it of dependencies? Will it crash, reveal sensitive data, etc.? How can I predict errors that will only occur under real-world conditions? Will my application expose security defects when running under heavy load? Is my application ready for deployment in hostile conditions? Holodeck can be used for the following: (more on how to use Holodeck and tests you can run) Investigative testing Stress testing Code coverage testing Ad-hoc/Exploratory testing Application analysis and debugging Environment Simulation Security Testing Fault-injection Testing Network and File Corruption Tool Features & Benefits Holodeck's powerful environment simulation and extensibility allow you to assault your application and create hundreds of scenarios that you otherwise would not have the means, time or capability to perform. **Automated Point-and-Click** Fault Scenarios (more.) Holodeck ships with		

		many "out-of-the-box" fault scenarios that are realistic and match real world, hostile conditions. These pre-configured faults will fail certain API function calls to simulate Operating System faults such as File in Use, Insufficient Memory, or Network Down. File Corruption. Files are corrupted at the time of use and Holodeck redirects the application to the new corrupted file. This allows you to create a different corrupted file every time the application accesses the file. :: more>> Resource Faults. Fail access to a specific process, library, file, folder, registry key, registry value or COM object and easily target single resource dependencies with realistic fault simulations. Systematically remove each dependency in turn and observe the results Network Faults. Corrupt files and network packets using powerful regular expressions System Faults. Limit the amount of disk, memory or network bandwidth that your application has access to Custom Faults. Modify existing faults or create your own custom faults. Automatic Test and Fault Generation. Analyzes your application and generates tests for you. Improves stress testing results and increases code coverage numbers Application Function Call Logging. Holodeck intercepts and logs all API calls regardless of whether it was called directly by the application or not. So, you can see if APIs called other APIs and you can record the behavior that led to a fault for easier debugging and reproduction. This makes it very easy to find out what was happening precisely at the time an error occurred - and is an extremely valuable asset for your test or development team, who typically spend countless hours trying to reproduce defects reported by customers or other testers. :: more>> Record/Replay. Allows developers to easily reproduce and fix bugs found by Holodeck **Operations intercepts**. Allows fault injection between application dll's as well as between application and system. Holodeck will intercept hundreds of Windows API's, including Registry, File, Network, Dangerous, COM and		

		Memory operations. You can wrap multiple applications simultaneously, and automatically wrap any new processes that are spawned during your test. You can also track the disk, memory and network bandwidth consumption of your application :: more>> Works with other automated test frameworks. Holodeck can be used in conjunction with your already existing Automated Test Frameworks to increase the probability of finding bugs. It can be used for ad-hoc testing by manipulating tests, faults, limits in the UI while running your test framework :: more>> Integrated debugger. The debugger catches all application crashes and exceptions and creates a minidump file that can later be loaded into Visual Studio for debugging, closing the loop between tester and developer. The developer can instantly see the line of code on which the crash occurred as well as the entire state of the machine at the time of failure. You can also compare logs from successive builds of your product to look for unwanted changes. :: more>> Using limits to quickly inject system faults. Holodeck will allow you to view how much of the disk, memory or network resources the AUT is using and restrict those resources to simulate different test scenarios without having to physically change the memory or disk availability on the testing computer. These limits include Disk Limits, Memory Limits, and Network Limits. Use these limits to simulate different hardware configurations without having to change your system hardware. API Scheduled Tests and Test Wizard. You can create scheduled tests in just a few clicks and you can also program the test to fire only when the conditions you define are met, such as: Fire xx% of time; Fire based on call stack matching; Pause when a test fires so you can set other tests, faults etc at that exact moment **Complete control over the application.** You can remove, stop, or restart any application in the current project. This allows you to recover after the application crashes, or restart the application if it gets into an unpredictable or frozen state. :: more>> Works with both Windows and .NET Applications. Holodeck supports reference parameters, private methods, and system reflection. Automation-Ready. XML scripts to drive the Holodeck UI. Holodeck exposes everything you can do in the UI on a .NET interface and generates code that		

		allows you to easily instrument the application you are testing Fully Supported Windows Services. Launch or attach to any service regardless of whether it is a standalone service or part of a svchost process. :: more>> **Custom Test Project Creation.** Using the Custom Test Project, Holodeck can generate test code for .net and win32 APIs which allow you to create your own logic for intercepted functions. It is extremely easy to use - most code is generated for you and you select which APIs you want to intercept. :: more >> **Network Packet Logging.** You can see all the network packet traffic from the application and drill down into each packet to find out exactly what's being sent and received. If you've created a network corruption fault, corrupted bytes are highlighted in the Network Message Details Pane. :: more>> Code Coverage Test Generation. Holodeck varies when the test will be applied to more thoroughly test the application. You can analyze your application and automatically apply resource and corruption faults and tests that are most likely to cause failure. Holodeck also includes automatic test generation for stress testing and testing for code coverage of your application. :: more>> Fully-Extensible. Holodeck is fully programmatic and offers multiple extensibility points, allowing you to write your own imposter functions that intercept system API calls, which is extremely useful if you are testing an in-house library. You can also use Holodeck in conjunction with other development environments and other scripting languages, allowing extensibility of all the HD features to any DLL the user wants to test. It also includes: HeatApi. C++ DLL. Provides low-level fault injection functionality (Attach to app, Intercept function, Detach from app, Etc.) HoloScript. Managed C++ DLL. Allows high-level control of Holodeck features such as Set Limits, Inject faults, Create scheduled tests, Examine log entries, Attach to delegates (Thread created, Process created, Log created) Custom Intercept Generation Extensive Help Documentation. Revised and extended Help Documentation which are task-based rather than feature-based.		

MAJOR TOOLS (in the 93% of Market Share)

~~~~~~

VENDOR:		***MERCURY INTERACTIVE***		
TOOL NAME:		**LoadRunner [Over $5,000] 54%**		
Web Page		www.mercury.com/us/products/performance-center/loadrunner/		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Mercury's load/stress testing tool for web and other applications; supports a wide variety of application environments, platforms, and databases. Large suite of network/app/server monitors to enable performance measurement of each tier/server/component and tracing of bottlenecks. Integrates with other Mercury testing and monitoring products. Mercury LoadRunner Are your IT systems expected to support thousands of users across multiple application environments with a complicated mix of homegrown and third-party components? Does your organization have to deliver 99.9 percent uptime and availability in the face of both unpredictable user loads and a never-ending stream of product patches and upgrades? Does your IT organization adequately stress test applications for system, user, and network scalability before deployment to ensure no performance surprises? Mercury LoadRunner™ is the industry-standard performance and load testing product for predicting system behavior and performance. Using limited	Over $5,000 for 100 virtual users	54% of market share
~~~~~~

		hardware resources, LoadRunner emulates hundreds or thousands of concurrent users to put the application through the rigors of real-life user loads. Your IT group can stress an application from end-to-end and measure the response times of key business processes. Simultaneously, LoadRunner collects system and component-level performance information through a comprehensive array of system monitors and diagnostics modules. These metrics are combined into a sophisticated analysis module that allows teams to drill down to isolate bottlenecks within the architecture. LoadRunner supports a wide range of enterprise environments, including Web Services, J2EE, and .NET. It is the only performance scalability testing product customized and certified to work with ERP/CRM applications from PeopleSoft, Oracle, SAP, and Siebel. With LoadRunner, you can: Obtain an accurate picture of end-to-end system performance. Verify that new or upgraded applications meet specified performance requirements. Identify and eliminate performance bottlenecks during the development lifecycle. **Features and Benefits** Uncovers bottlenecks in complex IT infrastructures to minimize production surprises: Real-time performance monitors help quickly identify and resolve bottlenecks before deployment Ensures that production performance requirements will be met at go-live: LoadRunner uses sophisticated correlation, analysis and reporting capabilities to help improve application performance. Supports the most common enterprise environments:		

		Including ERP/CRM, Web, J2EE, .NET, wireless, and streaming media. Leverages investments in other solutions: LoadRunner fully integrates with Mercury's suite of testing and monitoring products to deliver the full value of Mercury's BTO offerings. **Papers** Learn more about Mercury LoadRunner and automated load testing by reading the latest white papers, application notes and solution sheets. ¦LoadRunner Data Sheet (399 KB) PDF **White Papers** ¦Mercury LoadRunner Best Practices Guide for SAP R/3 Early detection is the key to avoiding common problems such as poor online transactions process response time or inadequate batch throughput in SAP applications. LoadRunner can help you avoid pitfalls that bring down your applications. ¦Siebel 7 Integration with Mercury LoadRunner (182 KB) PDF Mercury LoadRunner is a highly scalable load testing solution that allows Siebel 7 customers to predict system behavior and optimize performance. This document describes the integration between LoadRunner version 7.5 and Siebel 7. ¦Effective Load Testing and Performance Management for Citrix MetaFrame-based Applications (655 KB) PDF With the growing popularity of server-based computing for running business-critical applications, customers require more precise measurement tools to monitor system performance and optimize server load. Mercury and Citrix have jointly developed a solution that accurately simulates application responsiveness at the user level for both test and production environments.		

		Web Performance Management Solutions for Streaming Media (141 KB) PDF An increasing number of Internet and intranet sites are using streaming media—with mixed results. Mercury's solutions allow you to verify that streaming media applications perform as expected before and after deployment, so you can benefit the most from this technology. Get Ready for Production with Mercury LoadRunner Tuning Module This paper examines the new trends in performance testing and describes how the new Mercury Tuning Module enables you and other performance testing customers to do a more effective job of finding and resolving bottlenecks. Further, the paper examines how you can best take performance testing results and use them in production to optimize overall system performance. Load Testing to Predict Web Performance You don't need guesswork to predict how well your Web applications will perform after deployment. You just need an automated load testing tool. Find out why load testing is the only way to accurately test and optimize the end-to-end performance of your Web site before going live. And gain insight into the criteria that define an effective load testing solution—namely, accuracy and scalability. Then walk step-by-step through the entire load testing process with LoadRunner, the industry-standard load testing tool. Web Performance Management Solutions for Wireless Applications Wireless technology will have a major impact on your e-business strategy. Let Mercury help you prepare for the opportunities and challenges of the wireless market with solutions that can help test, identify, isolate and fix performance bottlenecks in your Web infrastructure. The Complete Testing Solution for SAP's R/3 (418 KB) PDF SAP R/3 helps organizations streamline their business processes, reduce application development time and provide a high-level of reliability. Ironically, these achievements present several challenges for enterprise testing tools. Gain insight into the ways Mercury WinRunner, LoadRunner and Mercury TestDirector work to tackle these challenges.		

		WAN Emulation Overview Whitepaper (695 KB) PDF This is a great overview of how networks affect application performance and how WAN emulation can help customers pinpoint network-related issues. Though not discussed directly, the WAN emulation capability described is that provided in LoadRunner. **Analyst Reports** Annual Load Test Market Summary and Analysis (379 KB) PDF "Mercury remains the dominant market leader and the one vendor that all others aim to compete against." Once again, Mercury emerges the clear leader in the load testing market space, with 54% of the worldwide market in 2000. For distributed environments based on load testing product alone, Mercury fared even better, capturing 58%. The Load Test Revolution (460 KB) PDF Throughout 1999 (and even stronger into 2000), the market for load testing tools and related services in distributed environments has continued to gain traction and increase market momentum, reaching $214 million in 1999, an increase of 55% over 1998. For 1999, Mercury is the clear market leader across the board, retaining and defending their long-standing leadership position in this market space. Mercury retained 40% overall market share for tools and related services, growing load testing revenue from $56.9 million in 1998 to $85.1 million in 1999. In terms of pure load testing revenue in the web space Mercury has 49% market share. **Solution Sheets** The Complete Testing Solution for Oracle Applications (146 KB) PDF Mercury combines the strengths of three industry-proven tools—Mercury TestDirector, WinRunner and LoadRunner—to provide a complete solution for Oracle Applications. Learn how these tools can test the functionality, scalability and performance of your Oracle system, as well as manage the testing process to ensure application readiness/		

		The Complete Testing Solution for SAP's R/3 (418 KB) PDF Mercury's complete solution for testing SAP R/3 consists of Mercury TestDirector, Mercury WinRunner and Mercury LoadRunner. Learn how this solution can accelerate your R/3 implementation or upgrade and give you the confidence that your system will work reliably throughout its lifecycle. Enterprise Testing and Performance Management for Java Applications (812 KB) PDF Mercury offers broad support for the most prevalent tools, environments and technologies in the Java space. Learn how our experts have integrated Java support into our Application Delivery and Application Management solutions.. **Detailed Reports** Real-time Monitoring during the Load Test Learn how you can accelerate the load testing process. This white paper describes the most comprehensive set of performance testing monitors available today within Mercury LoadRunner. **Articles** Learn how leading hardware, database, enterprise resource planning (ERP) vendors and ISVs use Mercury LoadRunner in their benchmarking and competency centers to produce and publish standard benchmark results, white papers and add-in products. Amdocs HP and Amdocs Benchmark Proves Virtually Unlimited Scalability Baan Baan ERP 5.0b Benchmark On IBM RS/6000 BMC BMC Software and Mercury Partner to Maximize ERP Application Availability and Performance BT BT's Web Application Performance Testing, - A Case Study of an On-line Learning Application Cisco Cisco Quality of Service Solutions Protect Oracle Application Performance		

		Clarify and Microsoft Front Office Scalability and Performance with Clarify and Microsoft SQL Server Compaq Scalability of Oracle 9i Real Application Clusters on Compaq Tru64 UNIX Dell Michael Dell demonstrates scalability using Mercury LoadRunner at the DirectConnect conference HP HP's Enterprise Server Sets New Scalability and Performance Record for PeopleSoft HP/Microsoft Microsoft Uses HP NetServers to Migrate Customers' Solutions to SQL Server 7.0 IBM IBM Siebel 7 Performance and Scalability Benchmark Intentia and IBM Intentia and AS/400 offer a high octane Java ERP application Medtronic Medtronic benchmark on AS/400 Onyx and Microsoft Onyx2000/SQL 7.0 Benchmark White Paper Oracle Scalability of Oracle 9i Real Application Clusters on Compaq Tru64 UNIX Oracle Tuning Oracle Applications R11.03 on Windows NT Oracle Oracle Applications Standard Benchmark PeopleSoft PeopleSoft CRM 8.4 Benchmark Using SQL Server 2000 on Unisys ES7000 PeopleSoft PeopleSoft 8 Enterprise Portal Using Oracle 8i on Compaq Alpha Servers PeopleSoft PeopleSoft Enterprise Portal Million User Benchmark PeopleSoft PeopleSoft Internet Architecture Benchmark on PeopleTools 8.10 with IBM DB2 and Sun Solaris Siebel Siebel's PSPP (Platform Sizing and Performance Program) Siebel Siebel 7 Benchmark of 30,000 Users on IBM DB2 Siebel The Impact of NetCache on Siebel 7 Sun Microsystems Cognos Series 7 Scalability Benchmarks on Solaris 8		

		(Go to their web page to see these links)		

VENDOR:		***IBM RATIONAL***		
TOOL NAME:		**Performance Tester [Over $5,000] 11%**		
Web Page		www-306.ibm.com/software/awdtools/tester/performance/index.html		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Rational Performance Tester Overview IBM Rational Performance Tester is a load and performance testing solution for teams concerned about the scalability of their Web-based applications. Combining ease-of-use with deep analysis, Rational Performance Tester simplifies the test creation, load generation and data collection processes to ensure the ability of their applications to scale to thousands of concurrent users.	Over $5,000 for 100 virtual	11% of market share Web page very confusing and hard to use. Nothing like the old Rational web page before they merged with IBM

		Multi-user performance testing tool for any team needing to validate web application scalability before deployment. Windows and Linux-based user interface. Windows and Linux supported as distributed controller agents. Delivers both high-level and detailed views of tests with a rich, tree-based test editor. Provides flexible modeling and emulation of diverse user populations. Enables large multi-user tests with minimal hardware resources. Delivers automatic identification of and support for dynamic server responses. Provides flexible test customization with custom Java code. Enables immediate recognition of performance problems with real-time reporting. Speeds problem resolution by identifying the root cause of poor performance. For further product details please view the Rational Performance Tester data sheet.		

VENDOR:		***COMPUWARE***		
TOOL NAME:		**QALoad [Over $5,000] 9%**		
Web Page		**http://www.compuware.com/products/qacenter/performance.htm**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Compuware's QALoad for load/stress testing of web, database, and char-based systems. Integration with other Compuware tools provides an in-depth view by monitoring its operating system, database and network components, as well as the application itself. Works with a variety of databases,	Over $5,000 for 100 virtual	9% of market share

		middleware, ERP. ***QA*Load** Testing for application performance under heavy load Today's client/server systems are expected to perform reliably under loads ranging from hundreds to thousands of simultaneous users. Organizations need to perform repeatable load testing and determine the ultimate performance and potential limits of a system. *QA*Load helps you achieve loads that mimic realistic business usage as well as validate that the system can meet acceptable service levels. The top five questions to ask yourself about load testing Do you experience problems with performance in production? What is the cost of downtime, including monetary, man hours, and intangibles such as customer satisfaction, reputation, and opportunity cost? Does your application scale with an increase of users? Do you have a method for obtaining real performance metrics? How do you repeat/reproduce a performance problem? Take a closer look at performance testing with *QA*Center Performance Edition. QALoad Product Detail Advantages Scalable Testing		

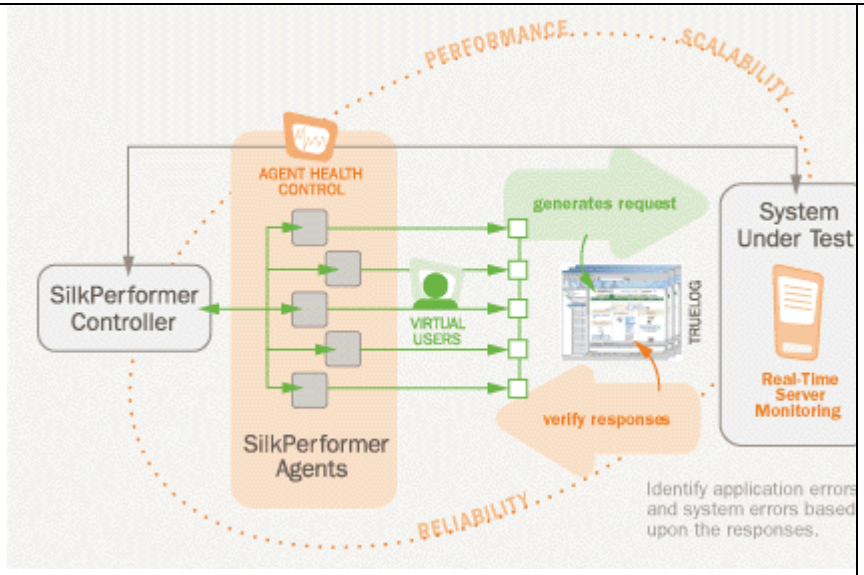
		Streamlined Test Script Development Comprehensive Analysis Integrated View of System Resources Supported Technologies Scalable Testing Using *QA*Load, you can emulate the load generated by hundreds or thousands of users on your application—without requiring the involvement of the end users or their equipment. You can easily repeat load tests with varied system configurations to achieve optimum performance. From the Conductor module in *QA*Load, you set up a load testing scenario to control the conditions for the test, create the virtual users you need to simulate the load, initiate and monitor test and report the results. A Player module simulates the roles of users performing multiple functions using testing scripts that represent your application Streamlined Test Script Development *QA*Load's Script Development Workbench provides customized modules that facilitate the development of testing scripts. These modules, called EasyScripts, allow testers to record the traffic an application generates and convert it into a testing script. The resulting scripts directly reflect the actual traffic generated by the application and measure the time taken to perform these transactions to ensure that the system under test will perform to specifications under load. Comprehensive Analysis During each *QA*Load test session, performance statistics can be viewed online. *QA*Load enables testers to insert checkpoints into scripts to identify and review specific areas of system performance for a test. *QA*Load displays test results in a variety of reports, graphs and charts. The data from a testing session can also be automatically exported to office		

		automation packages or custom analysis programs for further review. Integrated View of System Resources Compuware's ServerVantage provides an in-depth view of your application by monitoring its operating system, database and network components, as well as the application itself. *QA*Load's integration with ServerVantage provides monitoring for resource utilization problems while you generate a realistic load on your system. Application response time and resource utilization data from a load test can be accessed from the *QA*Load Conductor, allowing you to quickly identify and resolve application bottlenecks. Supported Technologies **Database:** ADO, DB2 UDB, Oracle, Sybase, SQL Server, ODBC **Distributed:** CORBA, TUXEDO, Winsock, Java, Citrix ICA **ERP:** Oracle Applications, SAP R/3, PeopleSoft, Siebel **E-commerce:** HTTP, SSL, SOAP, XML, Streaming Media **Legacy:** VT100-520		

VENDOR:		***SEGUE***		
TOOL NAME:		**SilkPerformer [Over $5,000] 7%**		
Web Page		www.segue.com/products/load-stress-performance-testing/index.asp		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Enterprise-class load-testing tool from Segue. Can simulate thousands of users working with multiple protocols and computing environments. Allows prediction of behavior of e-business environment before it is deployed, regardless of size and complexity. SilkPerformer Lite version also available	Over $5,000 for 100 virtual	7% of market share

		forup to 100 simulated users. **Segue Software** is a global leader dedicated to delivering quality optimization solutions that ensure the accuracy and performance of enterprise applications. Today Segue® solutions are successfully meeting the quality optimization challenges of more than 2,000 customers around the world, including 61% of the Fortune 100. Our results-oriented approach helps our customers optimize quality every step of the way. **Load, Stress & Performance Testing** **Optimize enterprise application performance and quality to ensure business success** Today's mission-critical enterprise applications are often very complex, involving multiple tiers, different - and often disparate - technologies and legacy application components. Additionally, application development teams are under pressure as never before to develop new applications in less time and with fewer resources. All of these factors can make applications prone to debilitating performance, scalability and reliability problems unless they are properly tested throughout the development process. Deployment of quality applications and realization of maximum uptime is critical in order to gain or maintain a competitive edge. Deploying poor quality applications can lead to a risk of project failure, projects that are over budget and behind schedule, and an over-investment in the supporting IT infrastructure to ensure responsive system performance - all factors that can jeopardize a company's business goals. To prevent these issues, enterprises require assurance that the mission-critical applications they deploy will provide the required performance, scalability and reliability characteristics in the most efficient and cost-effective manner. Deploying high quality applications that meet the quality expectations and the service-level requirements of users, customers and partners enable organizations to realize their own business goals. With automated load testing solutions from Segue®, you can find and fix the weak points in virtually any type of application before it goes live. That's because Segue's solutions support a wide range of enterprise technologies		

		and applications, scale up to simulate high loads, emulate realistic user behavior, generate repeatable and accurate test results and pinpoint the root cause of bottlenecks. Get the confidence you need to successfully deploy business applications using …. **SilkPerformer**®, for powerful, enterprise-class load testing of mission-critical applications **SilkPerformer® Lite** for, a cost-effective approach for load testing just Web applications **SilkPerformer® Component Test Edition**, for testing remote application components early and efficiently in the development cycle ...from Segue, the industry leader in automated testing technology. **SilkPerformer** **Test the limits of your enterprise applications** SilkPerformer® is the industry's most powerful - yet easiest to use - automated load and performance testing solution for optimizing the performance, scalability and reliability of mission-critical enterprise applications. With SilkPerformer, you can accurately predict the "breaking points" in your application and its underlying infrastructure before it is deployed, regardless of its size or complexity. SilkPerformer has the power to simulate thousands of simultaneous users working with multiple computing environments and interacting with various application environments such as Web, client/server, Citrix® MetaFrame®, or ERP/CRM systems - all with a single script and one or more test machines. Yet its visual approach to scripting and root-cause analysis makes it amazingly simple and efficient to use. So you can create realistic load tests easily, find and fix bottlenecks quickly, and deliver high-performance applications faster than ever.		



Benefits

Increase application deployment success:

Performance, scalability and reliability metrics that meet business goals

Acceptable service-level thresholds

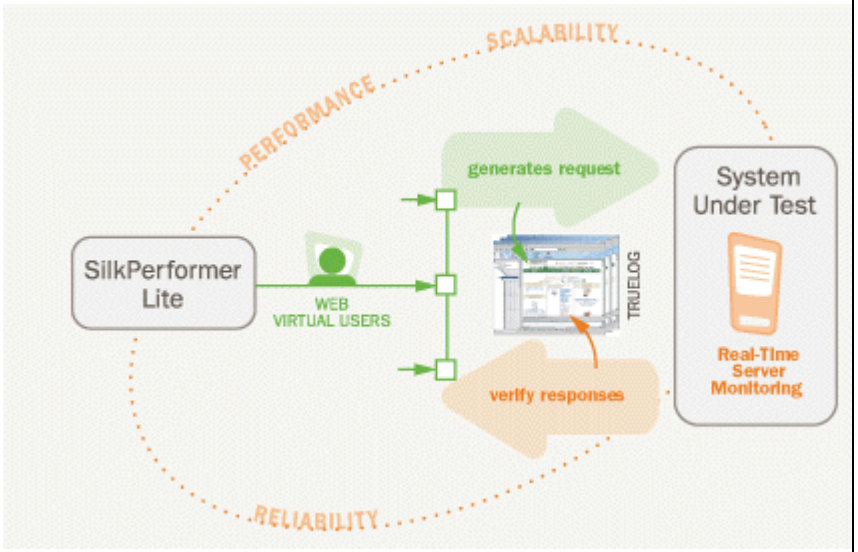
Shorten project cycle times and reduce project costs while improving quality:

Reduce defect and bottleneck repair costs

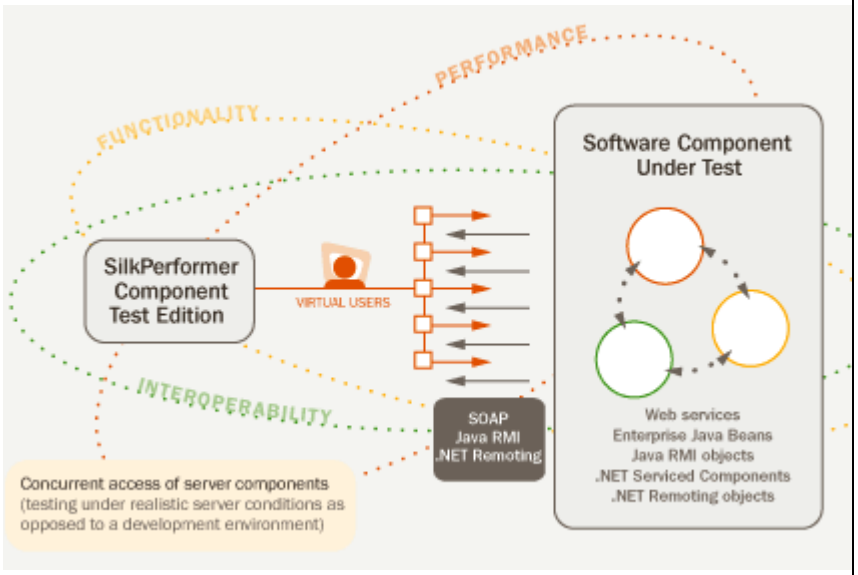
Efficient use of available QA resources (staff and equipment)

Eliminate over-investment in IT infrastructure:

Adequately sized server systems

		Optimal application performance and scalability **SilkPerformer Lite** **Optimize performance of Web applications cost effectively** SilkPerformer® Lite from Segue® is the ideal solution for small and mid-size organizations seeking an affordable, yet fully functional, solution for optimizing the performance, scalability and reliability of Web applications before they go live. SilkPerformer Lite enables you to predict the "breaking points" in your Web application, as well as in its supporting infrastructure, with the utmost accuracy. It's simple to use, too. With just a single standard computer and the intelligent visual scripting and built-in workflow offered within SilkPerformer Lite, you can quickly and easily create a load test that simulates the realistic behavior of up to 100 simultaneous users under dynamic load conditions. Visual root-cause analysis tools and management reports help you find bottlenecks easily, fix them quickly and launch high-performance Web applications on time.  		

		If your testing priorities include the testing of non-Web applications (such as client/server, Citrix®, ERP/CRM, middleware), loads of over 100 Virtual Users or analyzing heterogeneous enterprise environments, please refer to information about **SilkPerformer®**, Segue's enterprise-class load and performance testing solution. **Benefits:** Increase application deployment success: Performance, scalability and reliability metrics that meet business goals Acceptable service-level thresholds Shorten project cycle times and reduce project costs while improving quality: Reduce defect and bottleneck repair costs Efficient use of available QA resources (staff and equipment) Eliminate over-investment in IT infrastructure: Adequately sized server systems Optimal application performance and scalability Easy migration path to SilkPerformer for enterprise-class load testing **SilkPerformer Component Test Edition** **Optimize component quality and reduce costs by testing remote application components early in development** As the central building blocks of a distributed application, remote application components are key to ensuring application quality. SilkPerformer® Component Test Edition from Segue® lets you test and optimize three major quality aspects of critical remote components early in the application lifecycle - even before client applications are available:		

		Functionality Performance Interoperability Unlike unit testing tools, SilkPerformer Component Test Edition tests these components under realistic server conditions by exposing them to concurrent user access. What's more, its visual scripting technology and script import wizards are quick and easy to use. As a result, even non-programmers can detect potential problems that occur only under real-world conditions … resolve issues early in development before they become costly realities … and release high quality applications on time. ---  **Benefits:**		

		Reduce defect and bottleneck repair costs: Find problems early in the development cycle when they are least expensive to fix Shorten project cycle times while improving quality: Efficient reuse of existing test assets from development Improve collaboration and communication between QA and development Leverage your investment by migrating to SilkPerformer for enterprise-class load testing		

VENDOR:		***EMPIRIX***		
TOOL NAME:		**e-TEST suite [Over $5,000] 6%**		
Web Page		**http://www.empirix.com/default.asp?action=article&ID=419**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		A managed, hosted load testing service from Empirix with unlimited load generation capacity utilizing multiple points-of-presence. Free trial available. The Empirix e-TEST suite includes e-Manager Enterprise for test management, e-Tester for functional testing, and e-Load for scalability testing. e-Tester is a flexible, easy-to-use solution for automated functional and regression testing of your Web applications and Web Services. It provides the fastest way to create automated scripts that emulate complex Web transactions. e-Tester then allows you to use these scripts for automated functional and regression testing. The same scripts can also be used in e-Load for load and performance testing and in OneSight for post-deployment application management. **Features**	Over $5,000 for 100 virtual	6% of market share

		Integrated WebSmart technology allows e-Tester to automate the most complex Web applications and associated technologies out-of-the-box. Visual Scripting™ provides the fastest and easiest way to create test scripts through an intuitive graphical user interface, with no programming required. Advanced scripting extensibility with Visual Basic for Applications gives you the flexibility to handle a wide array of testing challenges, using a standard programming environment. Data Bank Wizard™ simplifies the creation of data-driven tests so you can leverage external data sources to drive your automated Web transactions. Default and user-defined test cases enable you to validate application functionality and ensure that the correct content is being returned to your users. OneScript allows you to use the same scripts, without modification, for functional and regression testing, load and performance testing, and application management. **Benefits** Keeps you ahead of changing applications by delivering the best capabilities for testing today's complex and dynamic Web technologies. Lets you focus on testing, not programming test scripts by providing the most intuitive scripting interface and the best out-of-box support for automated script generation. Reduces test cycle times by enabling you to quickly and efficiently run automated tests without having to go through extensive training or learn proprietary scripting techniques. Improves application quality by allowing you to effectively leverage automated functional and regression testing to increase test coverage and perform more extensive validation. **e-LOAD** Load test tool from Empirix Software; for use in conjunction with test scripts from their e-Tester functional test tool. Allows on-the-fly changes and has real-time reporting capabilities. Includes script editor with advanced debugging and maintenance capabilities. Works with a wide variety of platforms. **Products & Services: e-Load®** Load Testing for Web Applications e-TEST suite Brochure PDF » e-TEST suite Online DEMO » e-Load Free Evaluation DOWNLOAD » e-Load is a powerful solution that enables you to easily and accurately test the		

		performance and scalability of your Web applications and Web Services. Using e-Load you can simulate hundreds or thousands of concurrent users, executing real business transactions, to analyze how well your Web applications will perform under load. It also allows you to monitor the performance of your back-end application infrastructure, during your load test, to identify bottlenecks and help you tune application performance. e-Load is fully accessible via a Web browser interface, which enables testers and developers to collaborate during the application testing and tuning process. **Features** WebSmart technology allows you to automate the most complex Web applications and associated technologies out-of-the-box to create robust load test scripts. Scalable load generation allows you to simulate hundreds or thousand of users and more while minimizing the hardware required to run your test. Integrated ServerStats monitors enable you to gather critical performance metrics from your back-end server and network components to identify bottlenecks that occur under load. Intuitive Web-based user interface allows users to access e-Load via any Web browser to easily configure load tests, set up server monitors, run tests and view real-time and post-run results. Collaborative load testing allows multiple distributed users to share e-Load results during a running load test session and collaborate on application performance testing and tuning. **Benefits** Maximizes application performance by allowing you to tune your application under peak load conditions, so you can get the most out of your Web infrastructure. Improves application response times by helping you to quickly identify and address bottlenecks that limit performance under load and cause application slowdowns. Pinpoints hard to find bottlenecks by allowing you to monitor your back-end application infrastructure while running your load test, to identify performance bottlenecks at their source. Reduces testing time by enabling testers and developers to share test results in real-time, via the Web, to enable a collaborative testing approach. The Empirix e-TEST suite includes e-Manager Enterprise for test management, e-Tester for functional testing, and e-Load for scalability testing. This download includes e-Tester and e-Load. e-Manager Enterprise is available as a separate download.		

VENDOR:		***RADVIEW***		
TOOL NAME:		**WebLoad [[Over $5,000] 3%**		
Web Page		http://www.radview.com/products/WebLOAD_features.asp		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Load testing tool from Radview Software, also available as part of their TestView web testing suite. Capabilities include over 75 Performance Metrics; can view global or detailed account of transaction successes/failures on individual Virtual Client level, assisting in capturing intermittent errors; allows comparing of running test vs. past test metrics. Test scripting via visual tool or Javascript. Wizard for automating non-GUI-based services testing; DoS security testing. **WebLOAD** An ideal solution to practically test the performance of web sites-applications under user-defined system loads and view the results in real-time, ensuring that your applications meet performance expectations and maintain functional integrity under stress. User behaviors and web traffic scenarios can be realistically modeled in WebLOAD, including 'virtual' load generation and 'physical' (remote) performance metrics. Licensing/pricing is very competitive and based on the number of concurrent users (virtual clients) simultaneously hitting the site. Permanent license pricing starts 100 concurrent virtual-clients (user) load. One-year 'subscription' licensing is also available. **Other Features and Benefits** **WebFT Recording Environment** Uses WebFT to record and edit test agendas. See the WebFT Features and Benefits for more information. Cruise Control	Over $5,000 for 100 virtual users	3% of market share

		See how many users can access your application before a certain condition is met. With Cruise Control you can set WebLOAD to automatically increase the number of virtual users executing your load test until your defined threshold is met. **Load Profiler** Emulate user trends such as peak access hours, to measure your applications performance in multiple types of scenarios. **Throttle Control** Manually control the number of virtual users running your load test while it is in progress. Throttle Control allows you to increase or decrease the number of virtual clients at any point during the running of your test. **Data Drilling** Drill down into failures that occur with your Web application to isolate issues down to the object and instance level. **Web Services** Test your application's Web Services components. Easy-to-use wizards automate the process of testing non-GUI-based Web Services. **Script Sharing** Use your test scripts in both WebLOAD for load testing, and WebFT for functional testing, eliminating the need for separate functional and load test scripts. **Cacheing** Emulate a browser cache for each virtual client allowing you to accurately test the performance of your Web application in real-world situations. **Auto Run** Schedule your load tests to run automatically allowing you to make better use of your time. **DDoS** Simulate a Distributed Denial of Service attack so you can be sure your application is prepared.		

		MultiThreading Accurately simulate real-world browser traffic with multi-threaded virtual clients with WebLOAD. Some load testing tools use single threaded virtual clients. **Notification Manager** Receive instant notification when a problem is encountered. Automated alerts notify administrators via email or pager when application performance falls outside of a user-defined threshold. **Multiple Browser Support** Simulate realistic traffic to your application from a variety of browser types during your load test. WebLOAD supports IE, Netscape, and Mozilla. **Defect Tracking** Log defects directly from your WebLOAD test to your own Defect Tracking System. **Data Driven Testing** Generate data driven tests to thoroughly test your application with multiple data sets. In addition, by creating your input data directly through a WebFT wizard, you can more easily manage multiple input files for various tests. **Performance Measurements Manager** Gather performance metrics from multiple application server types and operating systems to isolate bottlenecks and issues. **Debug Environment** Add Breakpoints and Watch variables, or step through your test to debug your script.		

OTHER LEADING VENDORS

VENDOR:		*MICROSOFT*		
TOOL NAME:		**Application Center Test [$2,999]**		
Web Page		http://msdn.microsoft.com/library/default.asp?url=/library/en-us/act/htm/actml_main.asp		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Tool for stressing Web servers and analyzing performance and scalability problems with Web applications, including ASP, and the components they use. Supports several authentication schemes and SSL protocol for use in testing personalized and secure sites. The programmable dynamic tests can also be used for functional testing. Visual Studio .NET Edition. Application Center Test Purpose of Application Center Test Application Center Test is designed to stress test Web servers and analyze performance and scalability problems with Web applications, including Active Server Pages (ASP) and the components they use. Application Center Test simulates a large group of users by opening multiple connections to the server and rapidly sending HTTP requests. Application Center Test supports several different authentication schemes and the SSL protocol, making it ideal for testing personalized and secure sites. Although long-duration and high-load stress testing is Application Center	$2,999 per CPU	Not Evaluated

		Test's main purpose, the programmable dynamic tests will also be useful for functional testing. Application Center Test is compatible with all Web servers and Web applications that adhere to the HTTP protocol.		

VENDOR:		***REDGATE SOFTWARE***		
TOOL NAME:		**ANTS Load Professional Edition [$2,490]**		
Web Page		**http://www.red-gate.com/advanced_dotnet_testing_system.htm**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Advanced .NET Testing System from Red Gate Software. A load and stress testing tool focused on .NET web applications, including XML Web Services. ANTS generates multiple concurrent users via recordable Visual Basic .NET scripts and records the user experiences, at the same time performance counter information from Windows system is integrated into the results. **ANTS™ – Load Testing** ANTS Load™, and ANTS Load Enterprise Edition, analyze a web application's performance as it is put under load by multiple users. They will tell you how (or if) your web application scales as you ramp up the number of concurrent users. Read about the technical requirements of our ANTS products, or find out more about ANTS Profiler, ANTS Load, or ANTS Load Enterprise Edition.	$2,490 for 100 virtual clients	

		ANTS Load comes in three editions. All editions allow you to: **Analyse a .NET web application's performance** so you can find out how your application will behave when it is accessed by multiple concurrent users. **Record scripts** for ASP and ASP.NET websites through Internet Explorer. **Native support for XML web services** so you can test web services by calling methods on proxy objects rather than hand-code verbose XML. **Analyse response times** for web applications as the numbers of users increase. **Analyse server behaviour** by tracking CPU usage, memory usage and disk usage. **Write scripts in VB.NET** – we've embedded Visual Studio for Applications within ANTS Load so you can script in a powerful, familiar, environment. **ANTS Load™ Editions** **ANTS Load Entry Level Edition** lets you simulate up to 25 concurrent users, running on a single client computer. ANTS Load Entry Level Edition is ideal for proof-of-concept testing, testing early on in the life cycle, or load testing small- to medium-sized web applications prior to release. **ANTS Load Professional Edition** lets you simulate up to 100 concurrent users, running on a single client computer. ANTS Load Professional Edition is ideal for load testing medium-sized web applications at all times throughout the development life cycle. **ANTS Load Enterprise Edition** lets you simulate thousands, or even tens of thousands, of concurrent users, running on multiple client computers. ANTS Load Enterprise Edition is ideal for load testing larger web applications.		

VENDOR:		*OPENDEMAND*		
TOOL NAME:		**OpenLoad [$2,195]**		
Web Page		http://www.opendemand.com/openload/load_test.shtml		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Affordable and completely web-based load testing tool from OpenDemand; knowledge of scripting languages not required - web-based recorder can capture and translate any user action from any website or web application. Generate up to 1000 simultaneous users with minimum hardware. The **OpenLoad™** product suite is designed to address a specific need in the market for **simple, powerful and affordable** testing tools. OpenLoad provides the first **viable alternative** to high priced and overly complex commercial **load testing tools** and low-end **performance testing tools** that are largely unsupported and lack the robust feature set required to adequately test today's enterprise-level Web applications. **Open Load Tester V4.5** is the industry's first easy to use, browser based, performance optimization solution for **functional and load testing** of dynamic Web sites and applications. Powered by **IBM WebSphere** and DB2 Universal Database and fully integrated with IBM WebSphere Studio Application Developer, **Open Load Tester V4.5** delivers unprecedented ease of use, accuracy and scalability from a single integrated development and test environment. **Open Load Tester V4.5** substantially minimizes the time and skill set required to **stress test** and tune the performance of Web based applications and services by simplifying the process of building real-world user scenarios, verifying expected functional behavior through **automated regression testing** and **pinpointing performance bottlenecks** within Web applications and IT infrastructure from both inside and outside the firewall; offering application development and testing teams the ability to perform fast and productive optimization of dynamic Web sites and applications. **Open Load Tester V4.5** enables developers and testers to quickly optimize	$2,195 for 100 virtual users #3995 for 500 users and $9995 for 1000 uses includes maintenance.	

		the performance of **Java™ 2 Enterprise Edition (J2EE™)**, **J.D. Edwards**, **PeopleSoft**, **Visual Studio .NET** and **Macromedia ColdFusion** applications, including **Web Portals, CRM, SCM, ERP, e-Business and e-Learning** solutions. **Open Load Tester V4.5** is powered by IBM WebSphere Application Server 5.0, and fully integrated with IBM WebSphere Studio Application Developer 5.0, an industry standard IDE with a full-featured code-editing, debugging, unit testing and management environment, making it easier for developers to test and fix bugs through a single interface **The family of OpenLoad products includes:** **OpenLoad Express** - designed for small teams testing Web sites, applications and services with enterprise-level testing requirements and medium sized workloads; capable of simulating up to 500 virtual users. **OpenLoad Advanced** - designed for medium to large size organizations testing Web sites, applications and services with enterprise-level testing requirements and large workloads; capable of simulating 1,000 or more virtual users. **OpenLoad On Demand** - designed for any size organization requiring **pay-as-you-go** or subscription-based testing with term licensing options for testing inside and outside the firewall; capable of simulating any number of virtual users. This is a perfect option for System Integrators or software development firms that need to test on a project basis.		

VENDOR:	***TEVRON***
TOOL NAME:	**CitraTest [Over $5,000]**

Web Page		http://www.tevron.com/citratest.asp		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		CitraTest® is the ideal software test automation solution for conducting performance benchmarking, application monitoring, functional and latency testing for ANY Windows based application, including all Web browsers, all Web applications, all Terminal Emulation modes (3270, 5250, VT100, etc.), Citrix, Terminal Server and all legacy applications. One Solution with Support for Every Application Many Windows and Web applications contain unique on-screen control objects that are custom developed within non-standard environments. Software test automation tools that drive the target application by recognizing and interacting with its' on-screen control objects in runtime memory fail to offer universal compatibility with their applications of this nature. Because CitraTest® deploys a completely dynamic "image verification" methodology to interact with the targeted application, this powerful software test automation solution that is uniquely qualified to test Windows and Windows based Web applications created within non-standard development environments or that contain custom control objects. CitraTest® is a standalone Windows application whose image comparison and user input capabilities are included in the CitraTest function set, accessed from within a Microsoft VBA environment. CitraTest® image verification enables a consistent and seamless approach to test scripting and test execution for all applications running in the Windows environment as well as the Citrix Systems remote computing environment and Terminal Services. CitraTest® recognizes and evaluates the targeted Windows or Web application's response during test script playback by comparing the application's display to images stored in a user-created baseline. CitraTest® interacts with the target application by issuing standard keyboard and mouse events to the system under test (SUT) in accordance with the user defined test script logic. CitraTest® acts as a virtual human user in that it detects, evaluates, measures, and responds to the on-screen activity based the model represented by the test script logic. CitraTest® can automate ANY task that a	Over $5,000 for 100 virtual	Will test in the CITRIX environment

		real user can do from any PC. Load Testing for Citrix/Microsoft Terminal Services Realistic and accurate load modeling in a server-based computing environment requires a unique approach compared to web/client-server load testing. With applications deployed via Citrix MetaFrame/Citrix NFuse/Microsoft Terminal Services, no underlying controls or objects are delivered to the client. Instead, a proprietary protocol delivers only an image of the application from the server out to the client GUI. CitraTest VU is the ideal Load Modeling Solution for Citrix/Terminal Services **CitraTest VU**, the ideal load modeling solution for applications deployed via Citrix MetaFrame/Citrix NFuse/Microsoft Terminal Services, can… Generate 100% realistic load modeling scenarios with varying number of "virtual users" by using automated scripts to replicate real user activity in the actual client GUI Measure End-to-End application response metrics from the real user perspective- the actual client GUI- under different levels of virtual user activity Observe and record server performance metrics **What is the difference between CitraTest and CitraTest VU?** CitraTest is the application that enables you to create and play back the scripts. CitraTest VU simply refers to way it is licensed in the load testing methodology: you can run multiple scripts on a single "load machine" to generate the stress on your server(s)-under-test. **CitraTest** uses an advanced and intelligent "image recognition" approach to replicate transactions from the real user's perspective. **CitraTest** measures and reports End-to-End response times at the client GUI during playback by comparing the screen to "baseline response" images created during test script development. This makes it the ideal solution for automating real user activity within any server-based computing client interface. **CitraTest VU** can **intelligently** and **consistently** generate virtual user activity in the Citrix ICA,		

		Citrix NFuse, or Microsoft TS/RDP client interfaces with this unique **"image recognition"** approach. **CitraTest VU** is the only test automation solution for Citrix/Terminal Services that replicates real user activity using the actual client connection GUI, exactly as a real user would CitraTest's image recognition logic allows intelligent "synchronization" and persistence in your Load Test scripts. Other tools rely on time delays between each step and their scripts quickly fail as the environment is stressed and End-to-End response times increase. **CitraTest's** dynamic image recognition is automatically built into the script logic. ***No additional scripting is required to add image verification intelligence.*** It does ***not*** require that windows and images be displayed at the same location during each test playback. CitraTest runs on the client machine and is virtually non-invasive, adding no overhead to the client connection performance.		

VENDOR:		***HOSTEDLABS***		
TOOL NAME:		**HostedToolBox [$20 per month]**		
Web Page		**http://www.hostedtoolbox.com/features.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		hostedTOOLBOX provides a complete systems management framework and toolset for your mission-critical infrastructure. Rapidly assess your security position, load test your infrastructure, and monitor your internet applications. Our hosted, 100% web-based system can provide you third-party validation and piece of mind. **hostedTOOLBOX** enables IT professionals to manage their mission-critical applications and infrastructure in under 10 minutes. It makes it easy to: • Perform security audits at any time against all exposed systems • Run web application tests to establish performance and capacity	$20 per month for 100 users	Hosting Service Not Evaluated

		metrics • Monitor web applications, mail servers, firewalls, and other critical infrastructure • Diagnose and troubleshoot network issues from your customers point of view (outside your network) • View and analyze long term performance, availability, and security metrics Our suite of hosted, web-based applications offers you enterprise-class functionality, on-demand, without the enterprise-class cost. A few key benefits are: • Simple and easy to use, no learning curve • 100% Web-based, no installs, no software or hardware to manage • Hand-held accessibility, remote access from anywhere • No upfront cost, use as needed, pay only for what you need **hostedTOOLBOX** is perfect for organizations that want to manage their internet infrastructure quickly and efficiently. Organizations that can benefit from agile systems management are: • Consultants, IT managers, systems administrators, and lead developers. • Hosting companies or ISPs • Enterprise departments or mid-sized organizations with 500 servers or less. • Web-based companies and start-ups looking for the most time and cost effective solution Our hosted, web-based systems management suite makes it simple to manage your many servers and applications through one browser page. You define the systems you want to manage, setup the individuals or groups you want results sent to, and schedule your scanning, testing, and monitoring jobs within minutes. Everything is maintained on our servers which are spread across multiple data centers providing you with the highest levels of service		

		with no administration overhead. The data from every scan, test, or monitoring transaction, including all detected outages, is stored online for 5 years to give you a long term view of your performance and availability. We provide you with independent, third-party validation of your infrastructure SLA's enabling contract enforcement and service provider analysis. **Web Testing** - Our web application load testing product helps system administrators, testers, and developers analyze their web application under load and establish performance baselines. This helps determine breaking points, impact of code changes, and requirements achievement. It has the following features: Web-based script recorder allows you to browse through your test script. • Supports SSL, Cookies, GET, POST, and HTTP Authentication. • Every transaction is logged and uniquely tagged for correlation with web logs. • Error pages are logged and exportable. **hostedTOOLBOX** was created from our work at RSG, the consulting practice of our founders. We found that the existing testing, monitoring and scanning solutions in the marketplace were geared toward Fortune 1000 firms and were extremely cumbersome and costly. We wanted a fast, simple, and mobile solution that didn't cost a fortune, but still enabled us to deliver mature, enterprise-class projects. These insights led us build our own solution suite and we thought it made sense to make it available to other systems management professionals. **hostedLABS** is the parent company of hostedTOOLBOX and is the platform for our web-based applications and technology development. hostedLABS was officially formed in June of 2004 by Jason Rexilius and Mike Funk. We are an Illinois LLC and 100% employee owned and operated.		

VENDOR:		*AUTOMATEDQA*		
TOOL NAME:		**AQtime [$599]**		
Web Page		**http://www.automatedqa.com/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: Automated test tool from AutomatedQA Corp. includes web load testing capabilities. AQtime is the performance and memory profiler for Win32 and .NET applications. Over two dozen performance and memory profilers and productivity tools included **Debug, Optimize, and Deliver Rock Solid Applications** AQtime 4 is the next generation of AutomatedQA's award-winning performance profiling and memory debugging toolset for Microsoft, Borland, Intel, Compaq and GNU compilers. This new version combines the benefits of our flagship product AQtime (geared towards profiling of Win32 applications) and AQtime for .NET (the first performance and memory allocation profiler for the Microsoft .NET platform). Like its predecessors, AQtime 4 helps you easily isolate and eliminate all performance issues and memory/resource leaks within your code by generating comprehensive and detailed reports for your .NET and Win32 applications. AQtime is built with one key objective - to help you completely understand how your programs perform during execution. Using its integrated set of performance and debugging profilers, AQtime collects crucial performance and memory/resource allocation information at runtime and delivers it to you both in summarized and detailed forms, with all of the tools you need to begin the optimization process. From customized filters and graphical call	$599	

		hierarchies down to source code views. With AQtime's instrumentation in hand, you will know the exact cause of speed, memory usage, and application usability issues in your programs. As you optimize and improve your code, AQtime gives you all the tools to compare and merge results so that over time, an exact and accurate "picture" of your application's state takes shape. You will soon discover that AQtime is an invaluable PERFORMANCE WATCHDOG — Monitor and report on all areas that could degrade your project's performance and quality, Assess the thoroughness of your tests by using Coverage Profiling, Ensure that you make the right code improvements from day one, Help you drill down to the exact line of source code that is responsible for performance or memory allocation inefficiencies. AQtime is a TOTAL solution for professional Windows and .NET developers that want to eliminate guesswork during development and deliver rock solid programs when the project is complete. Via its unrivaled reporting and test result analysis architecture, AQtime does not *just tell* you that there are bugs and bottlenecks in your project - *it shows* you exactly which module, class, routine or line of code is causing the problem so you can go to work and quickly eliminate it! We are so confident that you will find AQtime an invaluable and essential ingredient in your current and future development efforts, that we offer a 60-day unconditional money-back guarantee. **Comprehensive Analysis of Application Performance** AQtime 4 vastly improves the process of optimizing performance and memory use. It digs deeper and finds more information than ever when analyzing your applications. Using the new Performance profiler, you will not only be alerted to the presence of performance slowdowns, you will also be able to easily find what caused those slowdowns.		

VENDOR:		***WEBPARTNER***		
TOOL NAME:		**Stress Testing [$500]**		
Web Page		**http://www.webpartner.com/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Test tool from WebPartner for stress tests, load performance testing, transaction diagnostics and website monitoring of HTTP/HTTPS web transactions and XML/SOAP/WSDL web services. Available as a downloadable software package or it can be run as an ASP Hosted solution. **WebPartner's** software includes **Stress Testing** features that quickly test the ability of your J2EE & .NET (HTML/JSP/ASP) website applications and Web Services (XML/SOAP) to keep up with an increasing number of users/requests. WebPartner Professional offers stress testing of approximately 60 simultaneous users from a single PC. You can also configure WebPartner Test & Performance Center to use as many additional load servers as you need to simulate hundreds and thousands of users/requests. Avoid costly and embarrassing failures when your website application is needed most. XML, SOAP, and HTML website applications often rely upon a number of different interlinking components (e.g. Microsoft IIS and Apache webservers, BEA, WebLogic, WebSphere,Tomcat, Resin, iPlanet, Oracle, 9i, Cold Fusion application servers, Oracle or Microsoft SQL Server database servers, and miscellaneous graphics servers, Symantec ISS Checkpoint security solutions, and Cisco networking equipment), any one of which can act as a bottleneck, or point of failure, slowing down or stopping the mission-critical business processes that are running on your website. WebPartner's Stress Testing can help you quickly right-size your website application infrastructure as you enhance and deploy new functionality on your website. Rapid deployment. Easy to use. Powerful JavaScript scripting to handle the toughest transactions.	$500 for 100 users or $999 for unlimited users	Available as a downloadable software package or it can be run as an ASP Hosted solution. Not Evaluated

		Available as a downloadable software package or it can be run as an ASP Hosted solution. WebPartner's software includes Stress Testing features that quickly test the ability of your J2EE & .NET (HTML/JSP/ASP) website applications and Web Services (XML/SOAP) to keep up with an increasing number of users/requests. WebPartner Professional offers stress testing of approximately 60 simultaneous users from a single PC. You can also configure WebPartner Test & Performance Center to use as many additional load servers as you need to simulate hundreds and thousands of users/requests. Avoid costly and embarrassing failures when your website application is needed most. XML, SOAP, and HTML website applications often rely upon a number of different interlinking components (e.g. Microsoft IIS and Apache webservers, BEA, WebLogic, WebSphere,Tomcat, Resin, iPlanet, Oracle, 9i, Cold Fusion application servers, Oracle or Microsoft SQL Server database servers, and miscellaneous graphics servers, Symantec ISS Checkpoint security solutions, and Cisco networking equipment), any one of which can act as a bottleneck, or point of failure, slowing down or stopping the mission-critical business processes that are running on your website. WebPartner's Stress Testing can help you quickly right-size your website application infrastructure as you enhance and deploy new functionality on your website. Rapid deployment. Easy to use. Powerful JavaScript scripting to handle the toughest transactions.		

VENDOR:		***PARASOFT***		
TOOL NAME:		**SOAPtest [$895] and Parasoft WebKing ($??)**		
Web Page		**http://www.parasoft.com/wsorg2 and http://www.parasoft.com/jsp/products/home.jsp?product=WebKing**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**

		Automates Web services testing – WSDL validation, unit testing and functional testing of the client and server, and performance testing. **Automate Web Services Testing with Parasoft SOAPtest** Parasoft SOAPtest is the most comprehensive tool for testing Web services. SOAPtest allows users to verify all aspects of a Web service, from WSDL validation, to unit and functional testing of the client and server, to performance testing. SOAPtest addresses key Web service issues such as interoperability, security, change management, and scalability. Because of its flexible nature, SOAPtest is an ideal choice for development engineers and QA professionals alike, since its unit tests can be leveraged into scenario-based tests, as well as load tests, without any additional scripting or re-inventing of the wheel. By utilizing SOAPtest throughout the software development lifecycle, users can prevent errors, improve quality and reliability, and accelerate the time to market for their Web service initiatives. ~~~~~~ **Parasoft WebKing®** **Automated Web application testing – performs coding standards, load testing, critical path testing, functional testing, security testing & more.** **Learn more about WebKing software solutions and special promotions here.** **White Papers & Presentations:** **> White Papers / Product Info** > Product Demo > Webcasts - Lehman Bros., Transcore, others **Software Evals & Purchase Info:**	$895 $????	Not Evaluated Not Evaluated

		> Download Software Eval - Free Trial > Contact Sales For Assistance ## WebKing Product Overview An automated Web testing product that automates the most critical Web verification practices: static analysis, functional/regression testing, and load testing. Parasoft WebKing is a comprehensive Web testing product that improves the functionality, security, performance, reliability, accessibility, and presentation of a Web application. To verify functionality, WebKing allows users to record critical user click paths by following them in a browser, then it automatically configures and executes functional/regression tests that verify paths and page contents while ignoring insignificant differences. To verify security, WebKing automates security static analysis and penetration testing to determine whether the application is vulnerable to many common and devastating types of Web security attacks (including SQL injection, cross-site scripting (HTML injection), buffer overflow, improper error handling, parameter manipulation, caching problems, and "Web bugs") and to verify that the organization's security policy is implemented and operating properly. To verify how the application handles realistic traffic levels, patterns, and combinations, WebKing provides intelligent virtual users and sophisticated ready-to-run load test scenarios. Users can easily customize these initial tests to use different paths, tools, traffic combinations, load distributions, and so on. To verify that the application's front end is constructed properly, WebKing's static analysis identifies client-side code that does not comply with development rules known to prevent functionality, security, presentation, and portability problems, pages that do not comply with project/organizational branding, content, and design rules, pages that do not comply with Section 508 accessibility rules, and pages with broken links, XML problems, and spelling errors. WebKing improves Web application quality throughout the software lifecycle and improves productivity across the entire team. Developers can use it to verify each application unit (servlet, JSP, etc.) as they write it, and QA team members can use it to perform application-wide verification. Moreover, WebKing works as part of a comprehensive team-wide Automated Error Prevention solution that reduces delivery delays and improves the quality and security of complex, multi-language enterprise		

		applications. **To learn more about WebKing, see:** • WebKing Data Sheet • WebKing Technical Papers • WebKing Reviews • JAVA Technology Solutions • Web Application Technology Solutions **Benefits** • Improve application reliability, functionality, security, performance, and presentation quickly and painlessly. • Deploy new content and functionality faster, with increased confidence. • Enforce security policies • Obtain instant expert feedback on code quality and potential defects. • Perform extensive testing with minimal user intervention. • Standardize page content, design, and layout. • Integrate all Web development tools into a single solution. • Reduce the time needed to test/debug, allowing developers more time on creative tasks. • Diminish the risks that cause that cause late, over-budget, incomplete releases. • Zero learning curve — test without scripting and complex configuration. • Ensure that best practices are applied consistently and uniformly across the team. **Features** • Designs realistic load tests by analyzing user server log files. • Unique technologies — paired with preconfigured test scenarios—allow users to begin complex load tests with no setup. • Intelligent virtual users exercise the application automatically. • Supports Windows Perfmon, SNMP, and custom monitors.		

		• Automates security static analysis and penetration testing to detect security vulnerabilities (including SQL injection, cross-site scripting (HTML injection), buffer overflow, improper error handling, parameter manipulation, caching problems, and "Web bugs") • Designs realistic functional tests by analyzing user server log files. • Automatically creates and configures functional/regression tests • Verifies that different paths through the site do not contain problems. • Dynamically populates forms with data source values. • Verifies custom design and content requirements. • Verifies if HTML and CSS code complies with W3C development rules. • Prevents common JavaScript and VBScript/ASP errors. • Checks if XML is well-formed and valid. • Verifies Section 508 compliance for accessibility. • Wizards guide users through required steps. • Integrates with WebSphere Studio Application Developer and Eclipse. • Supports flexible scripting with Java, JavaScript, Python. • Generates detailed reports in HTML, XML and text formats. • Produces real-time graphs and charts for load testing. **Platforms** • Windows 2000, Windows Server 2003, or Windows XP • Linux • Solaris **System Requirements** • Windows o Windows 2000, Windows Server 2003, or Windows XP o At least 256 MB RAM per processor (512 MB is recommended) o JRE 1.4.1 or higher • UNIX o Linux or Solaris o At least 256 MB RAM per processor (512 MB is recommended) o JRE 1.4.1 or higher 		

VENDOR:		***POWERPROXY***		
TOOL NAME:		**PowerProxy [$50]**		
Web Page		**http://www.powerproxy.net/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: A low cost HTTP/HTTPs proxy, from Orderly Software Ltd., has a range of basic load-testing features to test web servers and show debugging information about every request and response received or sent. For Windows. **What is PowerProxy?** PowerProxy is a load-testing HTTP proxy with features that can help you browse or buy product from busy web servers. **OK, and in English?** PowerProxy is a program that runs on your computer. It works with your existing web browser (such as the one you're using to see this page). As you browse the internet, requests for pages and responses from web servers pass through PowerProxy on their way to (and from) your browser. If you have a problem accessing a web page, perhaps because the site you are trying to browse is busy, PowerProxy gives you the option to keep trying to receive the page. PowerProxy then resubmits your request over and over to the target web server, until a full response is received. Once PowerProxy is satisfied that your original request is complete, you are automatically shown the full page. In effect, PowerProxy transforms your computer into an army of people all trying to browse pages, or buy product on your behalf.	$50 unlimited license	

		Put another way, running PowerProxy at 1000 requests per minute is equivalent to having 1000 browser windows open on your computer and pressing Refresh/Reload on each one every minute. This hugely increases your chances of seeing pages or buying product from very busy web sites. *Note:* If you're a provider, and you're worried that your servers won't cope with a high-demand sale, we can eliminate risk, make your customers happy, and save you money with our managed service, OrderlyQ. **Is that all it does?** No. PowerProxy also has features to allow you to automatically check when a particular page changes - useful if you are waiting for a product to go on sale at a particular time. PowerProxy also has a range of load-testing features that internet professionals can use to test their web servers, and also shows debugging information about every request and response received or sent. These features can be accessed in PowerProxy's Advanced Mode		

VENDOR:		***NEOTYS***		
TOOL NAME:		**NeoLoad [$2,395]**		
Web Page		**http://www.neotys.com/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Load testing tool for web applications from Neotys with clear and intuitive graphical interface, no scripting/fast learning curve, clear and comprehensive reports and test results. Can design complex scenarios to handle real world applications. Features include data replacement, data extraction, system monitors, SSL recording, PDF and HTML reporting, IP spoofing, and more.	$2,395 for 100 users	Not Rated

		Multi-platform: Windows, Linux, Solaris Load Testing Tool for Web Applications NeoLoad simulates hundreds of virtual users on your web site, getting performance statistics and revealing errors under stress. Before deployment in production, it answers the following questions: Will my application crash under real load? Will the response times meet my requirements ? How many concurrent users can my site handle ? NeoLoad pinpoints performance issues and helps solving them. When high-end products are too expensive, When cheap and open source products are too limited, Use the best mid-range tool: NeoLoad. Make your first test in 10 mn ---------- Clear and intuitive graphical interface. No scripting: fast learning curve. Clear and Comprehensive reports and test results. Easy to Use: no required training. Powerful ---------- Allows advanced scenarios to handle real world applications. Dynamically extracts links and data while the test is running to inject them in the scenario. System Monitoring: Get statistics from your servers (CPU, Memory, Disk usage, ...) PDF & HTML Reporting. Get all the must-have functionalities to optimize your web application. NeoLoad tests your application, no matter how complex it is. Immediate Benefits ---------- Improve your application reliability. Decrease the time to solve problems. Get control on deployment issues. Predict system behavior. Meet your project deadline.		

		Reduce the cost of your projects. Go live as expected with NeoLoad. Poor application performance or worse server crash can make you lose customers and slow down your business. Discover why load testing is critical. Neotys is the Right "Performance Partner" --- We help you about all performance issues in your application system. We provide a strong experience and focused skills on every step from development to deployment of web applications to ensure the success of your projects.		

VENDOR:		***APPPERFECT.COM***		
TOOL NAME:		**AppPerfect DevSuite – LoadTester [Free edition and $499 edition]**		
Web Page		**http://www.appperfect.com/products/devsuite/lt.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: Suite of testing, tuning, and monitoring products from AppPefect Corp. that includes a web load testing module. Support for cookies, basic authentication, URL re-directing and SSL and for grouping and parameterization of URLs. Integrated monitoring of target machine's system resources. Works with a variety of servers, OS's, and IDE's AppPerfect Corp., the leading vendor of J2EE Performance Management software **Key Features**	Standard Edition FREE Professional Edition $499 per user	Not Evaluated

		Scalable architecture to simulate thousands of virtual users Easy to use browser-based recording of test scripts Support for simulating exact browser behavior for more realistic testing (MSIE and Mozilla/Firefox simulation) Support for cookies, basic authentication, URL re-directing and SSL Support for grouping and parameterization of URLs, including dynamic parameterization (source is database, text file, etc) Support for extensive validation based on return code, returned html code, etc. Comprehensive reports including custom report designer and export of reports to PDF, Excel and HTML Full IDE integration for Eclipse, NetBeans, IBM WebSphere Studio, IntelliJ, JDeveloper and JBuilder Integrated monitoring of target machine's system resources **Supported Products** • JDK : JDK 1.3.1, JDK 1.4.1, JDK 1.4.2, JDK 1.5 • OS : Windows 2000/XP/2003, Linux x86, Solaris 8,9, Mac OS X • App Server : BEA WebLogic 7.x/8.x; IBM WebSphere 4.x/ 5.x; Apache Tomcat 4.x/5.x; JBoss 3.x • IDE : Eclipse 2.1.x/3.0; NetBeans 3.5/3.6/4.0; IBM WebSphere Studio 5.1, IntelliJ 3.5/4.0/4.5; JBuilder 9.x/10.x **Price and Availability** AppPerfect Load Tester is a component of the AppPerfect DevSuite. AppPerfect Load Tester is not sold separately. The AppPerfect DevSuite is currently available. AppPerfect DevSuite is shipped in two different editions: Standard and Professional. The AppPerfect DevSuite - Standard Edition is free for use, it has no license fees. The AppPerfect DevSuite - Professional Edition is licensed for US $495		

		per user. One license allows a single user to install and use the software on a single machine.		

PRICE OR TYPE UNKNOWN……

VENDOR:		*MONIFORCE*		
TOOL NAME:		**WebStress**		
Web Page		http://www.moniforce.com/		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Load and stress testing service from MoniForce BV. Includes recommendations on how to fix performance-related problems **Why perform load and stress testing?** Today, Internet and intranet environments must handle thousands, if not hundreds of thousands, of users. Unpredictable user loads and enormous complexity in the server environment create a risk of poor performance. In practice, not until performance problems occur, do companies usually start executing performance measurements. Unfortunately, that is too late and damage has already been done. Visitors are leaving your web site. At worst, this means discontent and missed opportunities. **Load and stress testing with webStress** webStress performs load and stress testing. With webStress, the Internet infrastructure is tested while simulating different numbers of visitors to anticipate future performance. You learn when waiting time, overload or unavailability will occur, but also what are the underlying, technical causes of these problems. With webStress, you can optimize performance and speed up the implementation of new web environments. Insight into just where any bottlenecks occur makes it possible to invest wisely in your Internet infrastructure.		

VENDOR:	*EMBARCADERO TECHNOLOGIES*

TOOL NAME:		Extreme Test		
Web Page		http://www.embarcadero.com/		
Web Testing	Functional Testing	Feature pro/cons	Aprox Pricing	Our Evaluation / Rating
		Performance test tool from Embarcadero Technologies. Capabilities include capture/playback/scripting and test probes/agents that work with a wide variety of applications. Load can be generated against HTTP and JDBC, allowing testing of Web sites, Web-based applications, and relational database systems. **Embarcadero Extreme Test 1.8** Embarcadero Extreme Test is a goals-based load testing solution for measuring and analyzing enterprise application performance. With expanded cross-platform support, Extreme Test enables development teams to quickly and accurately isolate system bottlenecks, and then make intelligent decisions concerning resource allocation, production readiness, and performance contingency planning **San Francisco, Calif. – May 3, 2005 -** Embarcadero® Technologies, Inc. (NASDAQ: EMBT), a leading provider of data lifecycle management solutions, today announced expanded cross-platform support for load testing of enterprise applications with the general availability of Embarcadero Extreme Test® 1.8. This latest version introduces new monitoring agents for Oracle® Application Server 10g, Gentoo Linux™, Apache Tomcat 5.x, and PostgreSQL 7.X. Additionally, Embarcadero confirms its commitment to open source and traditional databases with the addition of support for the PostgreSQL 7.X database platform, and new automated script support for Oracle 8i, 9i, and 10g. While offering more comprehensive functionality, the test environment can be more easily customized through the product's enhanced user interface. "The ability to quickly create usage scenarios that accurately reflect the stresses of real life is critical to the successful testing of complex enterprise applications," said Robin Schumacher, vice president of product management at Embarcadero Technologies. "With expanded support for Oracle and PostgreSQL 7.X databases, as well as new monitoring agents, Extreme Test 1.8 provides a more complete load and performance testing environment."		

		Embarcadero Extreme Test is a goals-based load testing solution for measuring and analyzing enterprise application performance, offering a completely integrated test development and execution environment. With Extreme Test, development teams can quickly and accurately isolate system bottlenecks, and then make intelligent decisions concerning resource allocation, production readiness, and performance contingency planning. "Making sure that our customers can register online and get in to a game quickly is a top priority because system downtime means potential lost revenues," said Mark Rizzo, vice president of operations and platform development at Perpetual Entertainment, a San Francisco-based producer, publisher, and operator of networked multi-player games for the PC and next-generation videogame consoles. "It's absolutely critical that our web applications be rock-solid from the moment they go live. Extreme Test's goals-based testing environment allows us to successfully simulate real-world usage and make our web application as bullet-proof as possible before it is deployed." Key new features of Embarcadero Extreme Test 1.8 include: Automated script support for Oracle 8i, 9i, and 10g, which significantly reduces the amount of time needed to create use cases for accurately testing the speed and capacity of Oracle databases. New monitoring agents for Oracle Application Server 10g, Gentoo Linux, Apache Tomcat 5.x, and PostgreSQL 7.X, expanding testing capabilities across multi-tiered software solutions. Testing and monitoring of PostgreSQL 7.X databases, allowing transaction analysis for the discovery of bottlenecks in performance before they become major problems. Enhanced, intuitive, and easy-to-customize user interface for better administration of your testing environment. Availability Embarcadero Extreme Test 1.8 is available now. To request a free evaluation copy, please visit the Embarcadero download center at www.embarcadero.com/downloads/. For more information about Embarcadero Extreme Test and other products from Embarcadero Technologies, visit Embarcadero Technologies website at www.embarcadero.com, or contact sales at 415/834-3131 ext. 3 or sales@embarcadero.com.		

		About Embarcadero Technologies Embarcadero Technologies, Inc. (NASDAQ: EMBT) is a provider of data lifecycle management solutions that help leading companies build, optimize, test, and manage their critical data, database, and application infrastructure. Nearly 11,000 companies, including 97 of the Fortune 100, rely on Embarcadero Technologies products to manage the explosive growth in data and ensure optimal performance of their complex, multi-platform applications and systems. Embarcadero Technologies is headquartered in San Francisco, CA. For more information, call 415/834-3131.		

VENDOR:		***QUOTIUM***		
TOOL NAME:		**QuotiumPRO**		
Web Page		**http://www.quotium.com/qpro_overview_load_testing.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Web load testing tool from Quotium Technologies SA. Capabilities include: cookies managed natively, making the script modeling phase shorter; HTML and XML parser, allowing display and retrieval of any element from a HTML page or an XML flux in test scripts; option of developing custom monitors using supplied APIs; more. Load Testing With QuotiumPRO **Load testing is very important in today's economy.** Companies with mission critical web applications can not afford to have poorly performing web sites. As web site growth evolves, and systems upgrade, the complexity of a company's web site architecture increases. Components within the web architecture change making applications increasingly likely to encounter performance issues. Applications have to be able to face a great number of users with unpredictable load variations. Poor performance means reduced profits, lower customer satisfaction and a loss of brand equity. QuotiumPRO™ is a **load testing solution** able to measure and predict the behavior and performance of an application on a global scale. QuotiumPRO™		Not Evaluated

		lets you load test both your infrastructure and the architecture of an application by simulating a large number of users with many different profiles. QuotiumPRO™ can be combined with performance and transaction monitors in order to provide specific performance data for the different components of your web application architecture. QuotiumPRO™ identifies and pinpoints in real-time any bottleneck and its cause. QuotiumPRO™ has a simple to use Graphical User Interface (GUI) that allows companies to shorten their load testing cycle. Using QuotiumPRO™ you can quickly optimize your **web application performance,** speed up the deployment process and reduce overall cost while enhancing end user satisfaction resulting in company growth.		

VENDOR:		***SOAP STONE***		
TOOL NAME:		**Soap-Stone**		
Web Page		**http://soap-stone.sourceforge.net/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		NETWORK BENCHMARK APPLICATION **Description:** *Soap-Stone* is a Network benchmark application which can put your network under load and conduct automatic benchmark and recording activities. **Requirement:**		BENCH MARKING

		OS Independent *Soap-Stone* is a network benchmark application, putting the network under load and conducting automatic benchmark and recording activities. **How Fast Is Your Network Today?** So you just installed a new Gigabit Ethernet network. Or you just installed a WiFi access point for your intranet. It seems fast, but just how fast can you really pump data through your shiny new network? More specifically, how fast can your applications talk various protocols (Sockets, HTTP, RMI, SOAP) one-way and round-trip from inside a Java Virtual Machine? *Soap-Stone* is the network benchmark application which can answer these questions for you by putting your network under load and conducting automatic benchmark and recording activities.		

VENDOR:		***OPTIMYZ***		
TOOL NAME:		**WebServiceTester**		
Web Page		**http://www.optimyz.com/servicetester.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Given today's competitive business environment, organizations require the highest quality applications possible. Optimyz delivers the most effective automated test solutions to application developers and QA managers to ensure the most reliable, scalable, and interoperable solutions available **WebServiceTester™** WebServiceTester™ is an integrated product suite of testing solutions specifically designed for testing Web Services. WebServiceTester™ provides complete turnkey testing solutions including Functional, Regression, Load,		

		Stress, Performance, and Monitoring testing solutions for your Web Service deployments. It is the most comprehensive solution available today! Optimyz's industry leading solution additionally offers ● Automated Functional Testing ● Automated Regression Testing ● Performance Testing ● Web Services Monitoring ● Automatic Test Client Generation ● Interoperability Testing ● Automatic Test Data Generation ● Automated Load and Stress Testing WebServiceTester™ generates and simulates client(s) automatically for any (pre-deployed) Web Service. Though most customers will typically use WebServiceTester™ for testing their Web Service before public deployment, WebServiceTester™ can also be used for testing and monitoring publicly deployed Web Services. **WebServiceTester™ :** **The Complete Turnkey Web Services Testing Suite!** ***The Leading Web Services Solution for Today's Enterprise Customers*** WebServiceTester™ is an integrated product suite of testing solutions specifically designed for testing Web Services. WebServiceTester™ provides complete turnkey testing solutions including Functional, Regression, Load, Stress, Performance, and Monitoring testing solutions for your Web Service deployments. It is the most comprehensive solution available today! Optimyz's industry leading solution additionally offers ● Automated Functional Testing ● Automated Regression Testing ● Performance Testing ● Web Services Monitoring ● Automatic Test Client Generation ● Interoperability Testing ● Automatic Test Data Generation ● Automated Load and Stress Testing **WebServiceTester™ is Easy to Use** WebServiceTester™ is easy to use! You can use WebServiceTester™ to test early in the process to ensure reliability, scalability, interoperability of your		

		Web Services! Be confident that your Web Service will meet your delivery goals on time and within budget with WebServiceTester™! **Testing Without a GUI** Web Services do not inherently display a user interface that can be tested. Most test tools today test at the GUI level and do not adequately test a Web Service. GUI level testing will not be able to efficiently troubleshoot problems. Since Web Services rely on interoperability, they have to be tested using all variations of possible client applications (i.e. .NET-based, J2EE-based, python-based, PERL-based…). A Web Service can also aggregate other Web Services to provide higher-level features. For example, a Web Service could provide high-level travel features by orchestrating lower-level Web Services for car rental, air travel, and hotel services. These various scenarios are virtually impossible to test with current tools until now! Try WebServiceTester™ today! **WebServiceTester™** WebServiceTester™ is an integrated product suite of testing solutions specifically designed for testing Web Services. WebServiceTester™ provides complete turnkey testing solutions including Functional, Regression, Load, Stress, Performance, and Monitoring testing solutions for your Web Service deployments. WebServiceTester™ generates and simulates client(s) automatically for any (pre-deployed) Web Service. Though most customers will typically use WebServiceTester™ for testing their Web Service before public deployment, WebServiceTester™ can also be used for testing and monitoring publicly deployed Web Services. WebServiceTester™ searches and finds a Web Service by querying some UDDI registry (public or private) or gets the information about the Web Service through a Web Service Description Language (wsdl) file made available otherwise. It then reads/ generates the required test data, binds with the Web Service(s) under test and invokes them in order to test various aspects of these services. The responses received from the service(s) are collected, stored and analyzed. Log-files, summary reports as well as bug reports for new/ unexpected failures are generated. This helps in validating and verifying the intended functionality of the Web Service(s) being tested. The user is kept informed about the progress as well as problematic situations via e-mail(s). WebServiceTester™ reduces or eliminates the need for manual set-up or intervention in testing Web Services to maximum possible extent. It provides the user with a multi-platform, user-friendly tool to automate Web		

		Service testing tasks. It provides an end-to-end solution from kick starting hundreds of Web Service functional tests to load testing, stress testing, monitor testing, performance testing, regression testing including monitoring of the execution of these tests (for time-out, etc.) to collecting and analyzing results to generating web reports as well as bug reports.		

VENDOR:		***SLAMD***		
TOOL NAME:		**Slamd**		
Web Page		**http://www.slamd.com/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: SLAMD Distributed Load Generation Engine is a Java-based application designed for stress testing and performance analysis of network-based applications. Requirement: Any system with Java 1.4 or higher **KEY FEATURES:** SLAMD provides a number of features that make it easy to conduct benchmarks, analyze performance, stress test applications, or perform any other tasks that require load generation. These features include: **Distributed Load Generation** SLAMD is a distributed application. There is a central server used to coordinate the load generation process, but the actual work is performed by client applications. The clients can be installed on any number of systems, and if more load is needed, then additional client systems can be added. In addition, a resource monitor client application can be run on the load		

		generation client(s) and/or the server(s) under load to measure things like CPU utilization, disk I/O, network load, etc. **Cross-Platform Compatibility** SLAMD can be used on any system capable of running Java 1.4 or higher. The administrative interface is a Java Web application that can be used in any standard servlet container (it is available with an embedded Tomcat engine), and the clients are standalone Java applications. Shell scripts are provided for starting the server and clients on Linux an UNIX systems, and batch files for Windows systems. **Simple HTML-Based and Command-Line Administrative Interfaces** The primary interface for accessing SLAMD is a Web application that works with any browser capable of rendering HTML 4.01. It does not require any client-side processing (e.g., JavaScript, applets, etc.), and can even be run in completely text-based browsers like Lynx. In addition, tools are available that can be used to perform most tasks from the command line, including scheduling jobs, obtaining results, uploading configuration files, and adding new job classes. **Highly Customizable Load Generation** In order to be as widely-useful as possible, it is necessary to provide a means of generating load against a variety of network applications. SLAMD is provided with jobs for interacting with LDAP directories, Web servers and applications, messaging servers, and relational databases. If the jobs provided with SLAMD are not sufficient for your needs, then you can write your own using a Java-based API or the embedded scripting language. SLAMD is also provided with tools for capturing and replaying traffic (including special support for the LDAP protocol), so it is possible to generate load without writing any code at all. **Self-Optimizing Jobs** In most cases, the performance of a network application is higly-dependent upon the amount of client load that is hitting it. Therefore, in order to find the best possible performance for the server, it may be necessary to run the same test many times with varying levels of load. SLAMD can automate this process through the use of optimizing jobs. The same workload will be run repeatedly, increasing the number of threads per client with each additional iteration, stopping automatically when the server believes that it has identified the optimal amount of load. The logic used to decide which iteration has the best performance is defined by the optimization algorithm used, and SLAMD is provided with several of them and also includes an API that makes it possible for the end user to develop new ones. **Resource Monitoring**		

		When performing benchmarking or performance analysis, it is often useful to have information about the overall state of the system(s) under load. It can also be beneficial to understand how the clients are performing, to ensure that they do not become a bottleneck and produce inaccurate results. To address this, SLAMD includes a number of resource monitors that can measure sytstem properties like CPU utilization, free memory and process size, disk I/O, and network load. This can also be used to obtain application-specific metrics (e.g., retrieving statistics from a directory server's monitor entries or measuring replication latency). There is also an API for developing custom resource monitors that can perform virtually any other kind of monitoring. **Powerful and Flexible Job Scheduler** In order to generate load against a network application, it is necessary to schedule a job that defines how that processing should be conducted. The job can be scheduled to start either immediately or at some point in the future, and dependencies can be defined so that one job will not be eligible to start until another job or set of jobs has completed. When a job is scheduled, a number of general properties can be specified, including how long it should run, the number of client systems to use and the number of threads on each client, and any user(s) that should be notified when the job has completed. There can also be a number of parameters specific to the type of job being executed. **In-Progress Reporting** By default, whenever a job is running, there is little to no communication between the SLAMD server and the clients used to process the job so that the clients will be able to focus their efforts purely on load generation. However, in some cases (particularly for jobs that may be scheduled to run for hours or days), it may be desirable to know information about the state of the job while it is still running. In such cases, the clients can be configured to periodically report their results back to the SLAMD server so that they can be viewed in the administrative interface. This is also available for resource monitor clients. In this way, if it appears that the job is not performing as well as expected, or if a problem has occurred, it can be detected early in the process rather than having to wait until the job has completed. **Easily Accessible Job Data** SLAMD stores all job data in an LDAP directory server (although there are plans to replace this with an embedded Berkeley DB Java Edition database in the near future). This means that all results are available at any point in the future. Any previous job executed can be easily cloned to repeat the same test (with or without any modifications), and the results of that processing can be viewed either numerically or graphically. The resuls from multiple jobs can be compared, also numerically or graphically. Job data may also be exported to tab-delimited text files for use in other applications, or written out as a report in plain text, HTML, or PDF form (or any other form that is added using a Java-		

		based API for developing custom report generators). Information in the configuration directory may be organized into folders so that the results may be found more easily, and other files (e.g., configuration files or external documents) can be uploaded into those folders to provide additional information about the tests conducted, the results, obtained, or the environment configuration. **Security and Access Control** By default, SLAMD is configured to be accessible by anyone for convenience and ease of use. However, if it is to be run in a network that is accessible by many people, then it may be desirable to restrict access to the individuals that can use it. SLAMD offers a number of capabilities in this area. All communication can be encrypted using SSL, including both the access to the administration interface and the communication between the SLAMD server and the clients and/or resource monitor clients. It is also possible to require the clients to authenticate themselves to the SLAMD server, as well as users authenticating themselves to be able to access the administrative interface. If authentication is enabled, then it is possible to restrict access to certain features on a per-user basis (e.g., so some users may only view job results, while others can view and schedule jobs, while others can make configuration changes). There is also a read-only mode that makes it possible to provide minimal access to users for being able to view job information without requring authentication, and it is even possible to configure the server in this mode to only show information about a specified subset of the jobs in the server. **Complete Documentation** SLAMD includes nearly 700 pages of documentation that covers virtually all aspects of its use. The Administration and Usage Guide provides general instructions on installing, configuring, and using SLAMD. The Job Reference Guide documents the set of jobs that are included with SLAMD. The Job Developer's Guide, Resource Monitor Developer's Guide, Optimization Algorithm Developer's Guide, and Report Generator Developer's Guide all provide instructions for using Java-based APIs to write custom code for use with SLAMD, and the Scripting Language Guide provides information on using and extending the embedded scripting language. There is also a document included that discusses techniques and methodologies for benchmarking LDAP directory servers, although many of the topcis covered in that document may also be applicable to general use for other applications. **Special Features for Testing LDAP Directories** Because SLAMD was originally designed for use with LDAP directory servers (and because its primary author continues to use it for this purpose on a nearly daily basis), it includes a lot of special tools and capabilities that can provide additional value when using it to test directory servers. One tool provided for this purpose is MakeLDIF, an extremely powerful template-based utility for		

		generating LDIF data files. Another is the LDAPDecoder, which can operate as either a simple LDAP proxy or analyze tcpdump and snoop capture files to decode LDAP communication in human-readable form or even automatically generate SLAMD scripts based on the captured data so that the same communication can be automatically replayed or customized to simulate real-world directory-enabled applications. There are a number of jobs specifically designed for testing LDAP directory servers, the scripting language includes LDAP support, and several of the resource monitors are intended for use with directory servers. The document "Benchmarking the Sun ONE Directory Server with the SLAMD Distributed Load Generation Engine" provides a thourough description of the activities and methodologies involved in performing an accurate and reproducible benchmark against LDAP directories. **Special Features for Testing Web-Based Applications** Web-based applications have dramatically increased in popularity in recent years, and as a result, so has the need to be able to perform stress testing and load generation against them. To help with this, SLAMD includes a rather flexible HTTP client implementation that makes it possible to interact with these kinds of applications in a manner that can simulate many common browser behaviors, and the scripting engine makes use of this HTTP client to provide this support as well. A future release will include an HTTP proxy that can be used to record traffic from clients and write a SLAMD script that makes it possible to easily automate this testing, much like the LDAPDecoder tool does for LDAP clients. **Special Features for Testing Other Network Applications** Because of the highly extensible manner in which it has been developed, SLAMD can be used to test virtually any kind of network application. Besides LDAP and HTTP-based capabilities already discussed, jobs are provided for interacting with mail servers via POP, IMAP, and SMTP, and with relational databases using JDBC. All of these protocols are also supported by the embedded scripting engine. Beyond them, the job development API makes it possible to use Java to develop custom jobs or even to extend the scriping language. Further, The TCP record and playback tools can be used to capture requests from clients and replay them against a server to simulate many kinds of network load without writing any code at all.		

VENDOR:		*SPIRENT COMMUNICATIONS*		
TOOL NAME:		**Avalanche**		
Web Page		http://www.spirentcom.com/		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Load testing hardware + software appliance from Spirent Communications. Includes scenario recorder and integrates with Mercury's Loadrunner for more sophisticated user scripts. Supports all major protocols, SSL, forms, dynamic URL's, unique IP addresses, cookies, session ID's and more. Can simulate distributed DOS attacks, concurrent media streams, large numbers of realistic e-mail users, configurable link speed emulation and packet loss rates, HTTP aborts, and more. Web stress test appliance from with browser-based GUI from Spirent Communications. Tests scalability and robustness of web application infrastructure; can work with existing load testing software. Can maintain over 2 million simultaneous connections, 50,000 HTTP connections per second; supports cookies, forms, SSL, connection latency, noise, graduated load generation, aborts, etc.		Not evaluated

VENDOR:		*APACHE SOFTWARE FOUNDATION*		
TOOL NAME:		**Apache JMeter**		
Web Page		http://jakarta.apache.org/jmeter/		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Java desktop application from the Apache Software Foundation designed to load test functional behavior and measure performance. Originally designed for testing Web Applications but has since expanded to other test functions;		Not Evaluated

		may be used to test performance both on static and dynamic resources (files, Servlets, Perl scripts, Java Objects, Data Bases and Queries, FTP Servers and more). Can be used to simulate a heavy load on a server, network or object to test its strength or to analyze overall performance under different load types; can make a graphical analysis of performance or test server/script/object behavior under heavy concurrent load. **Apache JMeter** is a 100% pure Java desktop application designed to load test functional behavior and measure performance. It was originally designed for testing Web Applications but has since expanded to other test functions. **What can I do with it?** Apache JMeter may be used to test performance both on static and dynamic resources (files, Servlets, Perl scripts, Java Objects, Data Bases and Queries, FTP Servers and more). It can be used to simulate a heavy load on a server, network or object to test its strength or to analyze overall performance under different load types. You can use it to make a graphical analysis of performance or to test your server/script/object behavior under heavy concurrent load. **What does it do?** Apache JMeter features include: Can load and performance test HTTP and FTP servers as well as arbitrary database queries (via JDBC) Complete portability and **100% Java purity** . Full **Swing** and lightweight component support (precompiled JAR uses packages javax.swing.*). Full **multithreading** framework allows concurrent sampling by many threads and simultaneous sampling of different functions by seperate thread groups. Careful **GUI** design allows faster operation and more precise timings.		

		Caching and offline analysis/replaying of test results. **Highly Extensible:** Pluggable Samplers allow unlimited testing capabilities. Several load statistics may be choosen with **pluggable timers** . Data analysis and **visualization plugins** allow great extendibility as well as personalization. Functions (which include JavaScript) can be used to provide dynamic input to a test Scriptable Samplers (BeanShell is supported in version 1.9.2 and above)		

VENDOR:		***STEPHEN GENUSA***		
TOOL NAME:		**WebHammer 2.0**		
Web Page		**http://www.genusa.com/iis/webhamr2.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Low-cost utility by Stephen Genusa designed to test Web applications and servers. Configurable 1-16 connections per system CPU. WebHammer 2.0 is a new version of the utility designed to test Web applications and servers. **New Features** • Request methods GET, POST and HEAD supported	$49 (not available yet but will be soon)	Not Evaluated

		• Authentication supported
		• Configurable 1-16 connections per system CPU
		• Cookies supported (currently a beta feature)
		• User-Agent header supported
		• Redirects supported
		• Custom HTTP request headers supported
		• Optionally creates log file of all transactions
		• Configurable timeout
		• Calculates
		  o Seconds Per Request/Requests Per Second
		  o Maximum Request Time
		
		You can download an eval version from hamr2.zip		

VENDOR:		***WEBMETRICS [Out Sourcing Service)***		
TOOL NAME:		**SiteStress**		
Web Page		**http://www.webmetrics.com/loadtesting.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Remote, consultative load testing service by Webmetrics. Simulates end-user activity against designated websites for performance and infrastructure reliability testing. Can generate an infinitely scalable user load from GlobalWatch Network, and provide performance reporting, analysis, and optimization recommendations **Key Features** Simulated Traffic: SiteStress can generate hundreds or thousands of virtual users that interact with websites to assess how much load applications can handle, performance rates at various loads and application breaking points. Transactional Simulations: Typically anywhere from one to five different user behaviors are executed at the same time, precisely mimicking real end user behavior. Each transaction might simulate a different piece of functionality on		Not Evaluated OUTSOURCING SERVICE

		the site, i.e. login, checkout, order status, etc.. Service: Outsourced (Hosted/Managed) load testing service executed from Webmetrics' distributed load testing network with servers across North America, Asia and Europe. Expertise: A dedicated test engineer is allocated to each load test customer to provide performance analysis, diagnosis and recommendations for optimizing the performance of websites, applications and systems. Reports: Detailed HTML/Flash reports are generated after each test iteration, which include detailed performance analysis by Webmetrics' testing experts. Schedule: Testing engagements usually consist of three executions and can be as short as a few days or as long as a year, but typically take place within a 30-day window. **Key Benefits** SiteStress helps companies ensure that their websites and applications can perform for anticipated user loads. Benefits of SiteStress include: Time Saver: Outsourcing load testing saves time. In a typical engagement, the average customer spends less than four hours of their time involved with the load test. Time is money. Cost Effective: Eliminate the need to purchase expensive load testing software and infrastructure. Leverage Webmetrics' technology, GlobalWatch Network and engineering expertise. Proof of Quality: Load testing provides third party validation of the quality of online applications. Benchmarking: Know the potential of your systems by leveraging the experience of a team of engineers who have tested nearly every type of web system architecture. **Feature Details** **Outsourced Service** SiteStress is a completely outsourced service that requires no software purchases or downloads. Our engineering team manages all script development required for testing set-up and execution. In addition, Webmetrics? allocates a dedicated testing engineer to each customer to deliver hand-on testing expertise and analysis throughout an engagement. This analysis includes problem diagnosis, problem isolation and		

		recommendations for improvement. It is our goal to help each customer meet their performance objectives and will provide the information it takes to get there. **GlobalWatch** SiteStress executes testing scripts over our GlobalWatch network, which is powered by tier one infrastructure and connectivity providers and spans 21 cities worldwide. **Executive Summary Report** At the conclusion of a load test engagement, Webmetrics provides a comprehensive PDF report The customer retains reproduction rights to this (and all) reports. This report starts off with a bottom line analysis such as "The site could handle 3500 simultaneous users with an average page load time under 8 seconds and a 0.05% error rate", and then goes into detail on testing methodology, business goals, problem diagnosis and recommendations for improvement. **Server Monitoring** It is imperative that the servers be monitored during the duration of the testing. Webmetrics' engineers will closely monitor the back end server health using our VitalStats service, or obtain the back end health monitoring data from the customer in order to complete the analysis. **Benchmark Comparisons** Over the past six years Webmetrics has executed thousands of load tests. Our engineering staff has seen virtually every type of hardware and software combination. Without violating any confidentiality about any particular customer's results, our engineering staff will compare your test results with customers of similar infrastructures or applications when analyzing your result. Leverage this knowledge and expertise to Webmetrics to ensure you get the most out of your infrastructure. **Project Management** Load testing can be very complicated. Webmetrics ensures timeliness, accuracy and completeness by utilizing an internal project management system designed specifically for load testing. To ensure all customer requests and requirements are addressed for each execution all parties are kept abreast of all communication via a centralized email system.		

		Cost Effective Load Testing Although every customer has similar goals in load testing and there are standard packages available. Each package is customized to the requirements of the customer. Typical engagements include three iterations of a load test executed within a 30-day period of time.		

VENDOR:	***MINQ SOFTWARE***
TOOL NAME:	**PureLoad**
Web Page	**http://www.minq.se/**

Web Testing	Functional Testing	Feature pro/cons	Aprox Pricing	Our Evaluation / Rating
		Java-based multi-platform performance testing and analysis tool from Minq Software. Includes 'Comparer' and 'Recorder' capabilities, dynamic input data, scenario editor/debugger, load generation for single or distributed sources. **PureLoad - Load Testing made Simple!** PureLoad is a load testing tool that simulates hundreds of users executing requests against server based applications. Use PureLoad to verify that your application will meet the expected performance criteria. PureLoad reports quality and performance problems as well as detailed statistics gathered during a load test. PureLoad do also include extensive support for easy recording and testing of web based applications.		Not Evaluated

		Ease of use PureLoad is designed with ease of use as foundation. A test session is managed using the PureLoad Console which is the central point of control. Use the console to define the virtual users, design your scenarios, execute and evaluate the results. No script language to learn! **Web testing** PureLoad can be used to test a wide range of applications, but offers additional support to simplify testing web applications: **HTTP Recorder** - the PureLoad HTTP Recorder captures all requests between a web browser and the web application. This information can then be transformed into PureLoad scenarios for use in a load test session. **Web Crawler** - the PureLoad Web Crawler scans a web and creates PureLoad scenarios for all objects that are found. The PureLoad Web Crawler also shows statistics and reports broken links. The majority of web and application servers has been verified with PureLoad, including BEA WebLogic, Apache, Oracle 8i IAS, i-Planet, Microsoft IIS. **Extensive Reporting** PureLoad reports response time, failing requests, bytes transferred, and more in chart or text format. The result information from several load sessions can easily be visualized and compared in the PureLoad Result Comparer. **Fully distributed and platform independent** All components in PureLoad are platform independent and the runtime architecture is fully distributed. This powerful combination enables the use of single and multi CPU machines all mixed in a distributed environment using different OS flavours. **Available in two editions** PureLoad is licensed in two editions:		

		PureLoad Web Performance testing and analyzing tool that helps application developers to automate the process of load/stress testing web applications. **PureLoad Enterprise** This edition of PureLoad extend the web edition with the full flexibility to create custom client operations, which help application developers load/stress testing **any** server applications. **PureLoadTM Feature and Edition matrix** **PureLoad Web edition** Load/stress testing of web applications. **PureLoad Enterprise edition** Load/stress testing of server applications including support for non-web protocols as well as custom development of properitary protocols. The following table briefly lists the functionality and what is part of the PureLoad Enterprise and PureLoad Web editions. PureLoad is available in one delivarable. The actual edition is a matter of the license that is purchased. **General Features** **Distributed Architecture** Load can be generated from a single machine or distributed on multiple machines. PureLoad do also make use of machines with multi CPU's. **Platform independent** Verified on Solaris, Linux (RedHat 7) and Windows NT/2000/XP Server Editions.		

		Testing of web applications PureLoad tasks for testing of HTTP based applications including support for forms, uploading files, dynamic web sites etc. **Supported standard protocols** PureLoad is delivered with ready to use tasks for testing of NNTP, FTP, SMTP, IMAP, JDBC (relational databases), LDAP, Telnet and DNS. **Single point of control** From the intuitive graphical user interface all actions required to set up the environment, define scenarios and control the load execution is performed. **Scenario Editor** Create scenarios using the point and click interface. Includes a full blown scenario debugger including single step, break points and response introspection. **Virtual Clients** Attach virtual clients (worker threads) with a mouse click. **Real-time graphs and reports** The console instantly reports the current status of the load execution. Follow the trends and analyze during the execution. **HTML Reports** Results and graphs are exported to files in HTML format. Result metrics can also be stored in a character separated value (CSV file) for further processing in example Microsoft Excel. **Java Task API**		

		Build custom tasks using the Java Beans based Task API. **Data Driven Testing** The ability to specify dynamic parameters in tasks is crucial since a load execution must simulate real users. Dynamic data can be generated from many supplied parameter generators. **Bytes transferred statistics** Tasks that supports counting of bytes (ex. HTTP) will report various statistics of bytes transferred between the virtual client and the server application. **Load distribution** Make the amount of load change over time, assign scenarios to be executed on wpecified hosts etc. **Result Comparer** Graphically compare the results from one or several load executions. **Web Testing Support** **HTTP Recorder** The HTTP Recorder eases the process of catching all requests between a browser and the web server. **HTTPS/SSL** Supports SSL v3 **Dynamic Web Applications** Standard session handling techniques, such as cookies and URL-rewriting is supported.		

		Response Validation Support for parsing response codes or parsing page content for expected strings. **PureLoad Web Crawler** Utility that crawls a web of static content and reports various statistics and error. The requests can easily be generated into PureLoad scenarios for use in a load test session.		

VENDOR:		***LINCOLN STEIN***		
TOOL NAME:		**Bare-bones Perl script**		
Web Page		**http://stein.cshl.org/~lstein/torture/torture.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Bare-bones Perl script by Lincoln Stein for testing web server speed and responsiveness and test stability and reliability of a particular Web server. Can send large amounts of random data to a server to measure speed and response time of servers, CGI scripts, etc. **Torture-Testing Web Servers** Lincoln D. Stein Cold Spring Harbor Laboratory		Not Evaluated

		April 19, 1999 A few years ago I wrote a small Perl script called "torture.pl" whose purpose in life is to inflict pain and suffering on hapless Web servers. It sends servers increasing amounts of random data at increasingly shorter intervals until they either crashed or slowed to the point of unusability. In other words, the script launches a denial-of-service attack on Web servers. Before you call the cops and have me dragged off to the special prosecutor's office, let me explain. This script has two functions. First, it can be used to test the speed and responsiveness of a Web server. Second, the script can be used to test the stability and reliability of a particular Web server. When used for performance testing, you can measure the speed and response time of your Web servers, CGI scripts, and other Web enhancements. Although torture.pl isn't rigorously normalized for cross-server comparisons the way the WebStone metric is, it's good for measuring changes on a single Web server. Worried about the performance impact of a configuration change? Just run the test before and after the change to measure its effects. When used in torture-testing mode, torture.pl sends large amounts of random data to a server, trying to make it crash. If a server, CGI script, module, or template processor can't handle large amounts of random data, then it's not particularly well written and might even contain security holes. **Using torture.pl for Performance Testing** Torture.pl was written to take advantage of Perl's abilities to multitask on Unix (and Linux) platforms. For this reason, it won't run on Windows machines. However I've got a multithreaded prototype of the code up and running for Unix platforms, and this might be available for use on Win32 machines by the time you read this. Check my Web site (see URL resources). **URL Resources** Comprehensive Perl Archive Network (CPAN) http://www.perl.com/CPAN/ The torture.pl Home Page		

		http://stein.cshl.org/~lstein/torture Cold Spring Harbor Laboratory, Stein Lab		

VENDOR:		***CAI NETWORKS***		
TOOL NAME:		**WebSpray**		
Web Page		**http://www.redhillnetworks.com/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Low-cost load testing tool from CAI Networks; includes link testing capabilities; can simulate up to 1,000 clients from a single IP address; also supports multiple IP addresses with or without aliases. For Windows. **WebSpray^{tm}** http://www.redhillnetworks.com/products/webspray/webspray.htm Windows-based software that stress-tests web servers to determine their capacity. Simulates high levels of traffic in various patters and reports results. Detects broken links and other errors. Web server stress-tester to determine performance capacity and problems + Introduction Do you know how much traffic your Web server can handle? Can you tell if your hosting service is doing a good job? Does your Web site have any dead links? Are you just throwing money at your Web server with constant upgrades hoping it can keep up with expected traffic?		Not Evaluated

		With WebSpray™, the guessing game is over. This powerful, easy-to-use tool stress-tests your Web server and reports on errors and performance. **+ Easy to use** With WebSpray's simple, intuitive user interface, testing a Web site could not be easier: just enter the URL of the Web site to be tested and the results are just a few clicks away. **+ Simulates multiple accessors** Many less capable testing programs only use one IP address to test the target Web site; the least capable ones can not even simulate more than one client from the IP address. WebSpray is not only capable of simulating up to 1,000 clients from a single IP address but also of supporting multiple IP addresses with or without aliases. This important feature is typically found only in products costing much more. **+ Tests all pages** Some testing utilities only test the home page of the target Web site, which is not a representative real-user simulation for a number of reasons: Home pages are often cached by the Web server Accesses may be against pages other than the home page Linked pages may be on other Web servers + Tests multiple times To fully stress-test a Web site over time, WebSpray lets you to choose how many times each client will test selected pages, or you can specify how long the test should last. **+ Reports broken links** In performing its normal test operations, WebSpray reports any broken links found. + Compatible platforms **WebSpray is compatible with :** Windows 98 Windows NT 4.0 Windows 2000		

		Local area connection Dial-up connection		

VENDOR:		***SOFTWARE RESEARCH INC***		
TOOL NAME:		**eValid**		
Web Page		**http://www.soft.com/eValid/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Web test tool from Software Research, Inc that uses a 'Test Enabled Web Browser' test engine that provides browser based 100% client side quality checking, dynamic testing, content validation, page performance tuning, and webserver loading and capacity analysis. • Automated WebSite Testing & QA Solution: ***Spidering, Searching, Functional Testing, Validating, Server Loading, Performance Tuning, Timing, WebSite Metrics, Monitoring*** • 100% Realistic User-Perspective Tests • Tests Run Inside A Full-Featured Web Browser • Superior Ease-Of-Use, Power, Flexibility • Flexible Licensing For Excellent ROI **eValid** solutions help organizations maintain e-Business presence, improve website quality and integrity, enhance performance to prevent customer loss, reduce down time, and control costs. The **eValid** Web Analysis & Testing Suite is comprehensive, scalable, easy to use, and applies to any web application. Built entirely inside an IE-compatible browser, 100% realistic user experience results are guaranteed. The eValid WebSite Testing Suite Changing Your Thinking About Quality Testing Of WebSites!		Not Evaluated

		eValid is a Windows NT/2000/XP Test Enabled Web Browser™ Testing Tool Suite that performs every functions needed for detailed WebSite static and dynamic testing, regression testing, QA/Validation, page timing and tuning, transaction monitoring, and realistic & scalable server loading. Because eValid is actually implemented **as** a fully functioning browser eValid has native capabilities to handle testing of WebSite features that are difficult, awkward, or even impossible with other methods or architectures -- such as passive protocol taps, proxies, wrappers or systems that view a website only from the Windows desktop level. eValid tests every kind of web applications or website naturally and efficiently. **eValid Benefits** 100% realistic testing and loading data "from the browser" to accurately reflects true user experiences. • Simplified website analysis and superior ease of use to reduce your costs and speed your web application development. • Improved customer-perceived performance and continuity to protect your website investment. A truly universal quality assurance solution that applies to every kind of browser-viewable application. **Moderate pricing and flexible, user-friendly licensing.** **eValid Product Suite -- Major Feature Summary** The eValid suite has a very rich technical feature set that supports client-side site mapping and QA, functional testing and regression suite development, website timing and tuning, and server loading and capacity analysis. eValid General Testing Features Full capability browser (100% IE compatible). Intuitive in-the-browser GUI and on-line documentation and help. Record and playback of sessions in full ObjectMode with Adaptive Playback. Rich and powerful User Preferences. Easy-to-edit playback script files; logfiles are 100% spreadsheet, database ready.		

		A convenient Page Metrics Popup that details facts about the current page. Full support for testing every type of user-interaction: HTML/S, XML, forms, Java applets, ActiveX controls, modal dialogs, JavaScript, multiple sub-windows, pop-ups -- in short, everything that a browser can render! On-screen dashboard to simplify operation. eValid WebSite Mapping & Analysis Link checker to identify unavailable or broken pages pages. Multiple real-time report filters: slow-loading pages, unavailable links, too-old links, too-large links. Search all pages for string matches in complete HTML or visible text. Website search process can be controlled by on-screen setup of on search depth, number of pages, number of links, total time, protocols used, file extensions used, etc. Dynamic creation of a Complete Data Table that shows details of every page visited. Selectable search modes: full/foreground, normal, background/quick. Powerful 3D-SiteMap that shows connections and relations between and among all mapped pages. eValid Functional Testing, Validation, Regression Testing Multiple validation modes: content, document features, URLs, text fragments, selected images, image parts, applets, etc. Access to internal DOM details via the PageMap capability to assist in script debugging. Advanced Recording features for Java applets, ActiveX controls, Modal Dialogs. Wizards to exercise all links on a page, push all buttons on a FORM, and manipulate a FORM's complete contents. Pause/Resume, Single Step, Run Multiple, Run Forever playback control options. Integrated results logging and charting. Secure session, cookie, cache management support. Synchronization modes for for Flash, applets, variable-time download pages. Command line, batch mode, JavaScript, and interactive API interfaces. Alarm, Error, Warning, Timeout flag processing. eValid WebSite Loading & Capacity Analysis 100% browser-based user simulations -- no "virtual users!" LoadTest feature combines multiple scripts into realistic, multi-user loading scenarios. Easy LoadTest script setup process. Multiple-browser (e.g. 100+) auto-launch. Cache management feature to play back tests with no cache or an initially empty cache, with or without cookies.		

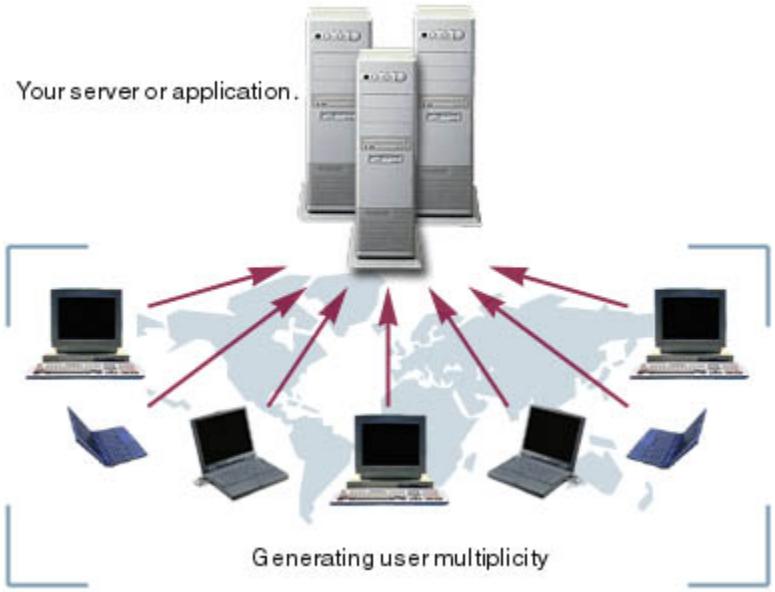
		Infinite User Key (IUK) LoadTest licensing option. eValid WebSite Timing & Page Tuning Detailed timing including analysis of 1st and 2nd tier transaction times. Timings include values for separate HTML components, images, total download time, and final page rendering. Session time limits, alarm sequences, and error flag processing. Event, timing charts, performance, and history charts. Page tuning data and performance timings to 1 msec resolution. eValid Test File Data Generation Automatic creation of sequential data to parametric script files. Generation of random data to parametric script files. Capacity for large amounts of data. Parametric value passing: scripts can call scripts. eValid Test Suite Management Creation of a test tree to handle 100's to 1000's of eValid tests. Selection of tests to execute under eV.Manager control. Automatic execution of tests with complete result logging. PASS/FAIL reporting, regression reports, complete test history. Simplified test script check-in and update.		

VENDOR:	***WEBPERFORMANCE INC***			
TOOL NAME:	**WebPerformance Trainer**			
Web Page	**http://www.webperformanceinc.com/**			
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Load test tool emphasizing ease-of-use, from WebPerformance Inc. Supports all browsers and web servers; simulates up to 750 users per playback machine at various connection speeds; records and allows viewing of exact bytes flowing between browser and server; no scripting required. Modem simulation allows each virtual user to be bandwidth limited. Can automatically handle variations in session-specific items such as cookies, usernames,		Not Evaluated

		passwords, IP addresses, and any other parameter to simulate multiple virtual users. For Windows, Linux, Solaris, most UNIX variants. Web Performance Trainer™ is web load testing and web stress testing software and tools targeted at web sites that use standard technologies such as web forms, J2EE, ASP, .NET, PHP, ColdFusion, Java, etc. Its automatic analysis can tell you how many users your web site is capable of handling at one time using your own unique performance criteria. Because data is collected at the URL level, it not only identifies slow web pages, but identifies the particular *part* of the web page that caused the problem. Simulating a load test with thousands of users with Web Performance Trainer™ is easy! Simulate up to 750 users from a single test computer, or install load generating "engines" on multiple computers at no extra cost. How do you know the simulation is accurate? Web Performance Trainer™ automatically performs load balancing on the *test machines*, measuring system load and memory usage to insure an accurate simulation. Looking for advanced features? Web Performance Trainer™ has those hard to find features such as ip spoofing, client certificates, multiple simultaneous test cases, authentication (usernames/passwords), SSL recording and playback, automated analysis and much more. #1 in customer satisfaction An independent report by Amplitude Research rated Web Performance Inc. as having the highest level of satisfaction among those currently using load and stress testing software. "...Web Performance Inc. demonstrated, on a statistical basis, significantly higher ratings than Mercury Interactive and Empirix among survey takers." price/performance leader With prices starting at $495, Web Performance Trainer™ is cost effective for small projects, and yet can scale to simulate the thousands of virtual users needed to test the largest of sites. The software includes all of the features you need to do serious testing such as automated analysis, data replacement, authentication, session management, ramping load generation, network speed simulation, and even load balancing of the test generation. consulting services are included		

		For those who'd like to run a performance analysis on their site but aren't sure where to start each Premium Maintenance/Support Contract comes with phone access to a performance testing engineer with years of experience. While our expert engineers can't actually tune your system for you, they can help you learn how to use the program, run the tests, interpret the data, and point you at the most likely performance problems.		

VENDOR:		***TECHNOVATIONS***		
TOOL NAME:		**WebSizer**		
Web Page		**http://www.technovations.com/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Load testing and capture/playback tools from Technovations. WebSizr load testing tool supports authentication, SSL, cookies, redirects. Recorded scripts can be modified manually. For Windows. **Load Testing Products** **Overview:** Our Load Testing suite consists of several modules that make testing easy. Modularization is intended to give you the ability to license the functionality you need, but with a pre integrated approach for easy installation.		Not Evaluated



Recording Modules:

Before starting any testing, you need to develop a user profile that you want to play against the target system. You can manually develop these scripts or use a recorder to accomplish this task. Recorders are the RAD (Rapid Application Development) tools used to generate test scripts. We support a automatic record-and-replay paradigm with our recorders so that the output of the recorder can be plugged into the load module and you are ready to test! The recording transcribes think times, client signatures, Authentication details, in addition to other functional characteristics. The recorded scripts can also be manipulated or optimized manually. Note that recorder modules are supported only for WebSizr and MailSizr.

Load Generator:

This is the module that generates user multiplicity against your server or application. This can create more than a 1000 users on a single PC, depending on workload. More details in next section.

Load Director Module:

		In some cases, you may need to simulate a much larger workload than what a single Load Generator Module can deliver. If you want to generate 10000 users, this is when you can use Load Director to control tests and aggregate reports from multiple **Load Modules.** **Report Generator Module:** This module is a data collector which collects data from the test and puts together a test report. **Batch Manager:** This module is used to batch a set of pre-determined tests for un-attended testing. Using this module you can do almost anything that you can do interactively. This tool again supports a record-and-replay paradigm at the test level. **Online Testing engines Vs Lab-Pack Testing engines:** We have applications that you can use online and ones that you can use in the confines of your lab. The engines that power them are the same, so your scripts will work perfectly in both scenarios. The output from both scenarios use the same analysis and monitoring modules as well. In short, we have ensured that irrespective of how you do the tests, the input and output will be 100% compatible with the lab pack version. The rest of this document shows how the Load testing modules are rendered to target different Systems Under Test. **WebSizr for Web Servers and Applications:** A Performance Analysis Application to size, benchmark and debug applications using HTTP servers and Applications, WebSizr can simulate workloads from thousands of Web browsers. WebSizr can be used to characterize Internet/Intranet and Electronic Commerce application under load, in addition to sizing HTTP servers. WebSizr, in addition to implementing various flavors of GET and POST requests, also supports advanced paradigms like SSL (Secure Sockets Layer V2/V3), TLS (V1.0), Authentication, Cookies, Redirects, etc to help the application developer trying to engineer a high-performance application (by providing the ability to scope into problems under user load) or a Systems Architect trying to optimize and		

		tailor an application to specific content and computing infrastructure. WebSizr can also be used in conjunction with DbSizr to understand/solve performance problems with developing web applications or with GroupSizr to understand Lotus Notes/Domino characteristics under user load. **Click on this link for more information on WebSizr** or on **this link to get a free 30-day evaluation copy of WebSizr.** **Modules Supported for WebSizr:** • Load Generator (included) • Load Module (included) • Report Generator (included) • Load Director (optional) • Recorder (optional) • Batch Manager (optional) **GroupSizr for Lotus Notes (formerly known as NoteSizr) Servers and Applications:** GroupSizr for Lotus Notes is an application that helps application developers design and implement high-performance and highly scalable Notes and Domino Applications. Deployment Specialists (Architects, Notes Administrators, System Engineers) to Optimize Servers and Applications to optimally field user workloads. In addition to exercising Notes Database functionality (Add, Update, Delete, Read and Navigate), GroupSizr offers capabilities to benchmark other Groupware functions and Notes applications like Indexing, Replication, Full Text, Mail and Custom Add-in tasks. It also provides an ability to help Manage Notes Servers by providing capability to call and execute commands on Remote Notes Consoles. GroupSizr can be used with WebSizr to benchmark Lotus Domino servers and Applications that can be accessed from browsers.**Click on this link for more information on GroupSizr** or on **this link to get a free 30-day evaluation copy of GroupSizr.** **Modules Supported for GroupSizr:**		

		• Load Generator (included) • Load Module (included) • Report Generator (included) • Load Director (optional) • Batch Manager (optional) **DbSizr for Relational Databases (Servers) and Applications:** Database Application developers or DBAs can use DbSizr to size their Database server and/or Database Application under load. These exercises can help developers engineer high performance applications and DBAs or architects put together Database Systems (a collection of servers) optimally. Stored Procedures or SQL calls can be simulated using DbSizr and against most popular database Servers [including SQLServer from Microsoft, Oracle, Sybase, Informix, etc]. DbSizr allows users to scope into problems at a highly granular level (SQL Statement level or Stored Procedure level). DbSizr is popularly used with WebSizr to comprehensively solve performance problems associated with web applications accessing database content.**Click on this link for more information on DbSizr** or on **this link to get a free 30-day evaluation copy of DbSizr.** **Modules Supported for DbSizr:** • Load Generator (included) • Load Module (included) • Report Generator (included) • Load Director (optional) • Batch Manager (optional) **MailSizr for Mail Servers and Messaging Applications:** Electronic mail has become an important application paradigm of today. There is an increasing number of users who depend on electronic mail and there are an increasing number of applications (workflow or otherwise) that are mail-enabled. Also, in many cases, the performance of your mail server may depend on the mail client being used (each mail client has it's own way of managing retrieval and presentation of messages). MailSizr can be used to		

		size and benchmark servers using leading Internet messaging protocols (POP3, SMTP, IMAP4 and NNTP) . MailSizr can be used to simulate workloads involving thousands of mail users. MailSizr allows application developers to prototype the implementation of high-performance mail clients and also helps in Capacity Planning of servers to handle specific mail workloads. **Click on this link for more information on MailSizr** or on **this link to get a free 30-day evaluation copy of MailSizr.** **Modules Supported for MailSizr:** • Load Generator (included) • Load Module (included) • Report Generator (included) • Load Director (optional) • Recorder (optional) • Batch Manager (optional) **Websizr Fact Sheet** WebSizr is a performance Analysis and Sizing framework for HTTP based Servers and applications. Application developers use WebSizr to engineer high-performance web applications. WebSizr helps the Architect/System Engineer to easily characterize the Performance, Scalability and Sizing of the following - • HTTP based web servers. • Internet/Intranet Applications. • Electronic Commerce Applications. WebSizr is primarily a server measurement tool but it can also help you simulate client behavior by modeling think times and client processing times using 'events'. This can also come to be useful if you are using Java Applets/ JavaScript/ Other client side scripts to offload work to the client. WebSizr supports all types of HTTP request types- GETS, POSTS, HEADS, etc. and also packs a set of easy to use but powerful test management verbs that make it easy to do advanced benchmarking exercises.		

		WebSizr supports Secure Sockets Layer (SSL V2 and SSL V3) and also supports TLS V1.0. WebSizr allows you to manipulate various user and network settings. It also gives the tester an ability to collect detailed timing statistics. It also captures the HTML/text pages that are part of the overall transaction. This makes debugging the test scripts easy. WebSizr also comes with a automatic report writer that writes test results to HTML. The report writer can document some of the most intricate test details which can help reproduce tests at a later time or help in the analysis of tests. WebSizr also supports a record and replay paradigm. Scripts that are recorded through **Technovations WebCorder** can be used by WebSizr to simulate user activity with unprecedented accuracy. **Follow this link** for a more detailed description of usage of WebSizr, including screen shots from a real test run. Better yet, you can try the application yourself by following the download link below.		

VENDOR:		***FACILITA SOFTWARE (UK)***		
TOOL NAME:		**Forecast**		
Web Page		**http://www.facilita.co.uk/main.cgi**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Load testing tool from Facilita Software for web, client-server, network, and database systems. Capabilities include proprietary, Java, or C++ scripting; windows browser or network recording/playback. Network traces can also be taken from over 15 third party tracing tools. Virtual user data can be		Not Evaluated (UK)

		parameterized. Works with a wide variety of platforms. The forecast product suite from facilita tests the performance and scalability of IT systems. forecast is a complete solution for performance and load testing across the enterprise. forecast verifies the integrity and reliability of mission-critical applications; helping businesses to ensure quality throughout the development lifecycle. forecast products can be used to load test web servers, application servers, databases, J2EE services and network infrastructure using standard or proprietary protocols. **Introduction** forecast tests IT systems by simulating the activities of multiple users or clients. forecast supports performance testing through the recording of real user interactions, the generation of scripts, the creation of test scenarios and variable test data needed to make the test realistic, the execution of tests and finally the statistical analysis of of the test results. forecast can test the performance of complex networked IT systems by using a combination of forecast's licensable components (Web, Net, Java, Database and Flex). Whatever your load testing requirements, forecast components are integrated into one framework allowing you to test the most complex of applications in a natural way. **Main Features** Fully distributed tests managed from a controlling PC. forecast has comparatively low hardware requirements and a small footprint enabling the support for thousands of Virtual Users on modest test equipment. Highly scaleable - forecast's distributed architecture and efficiency means very		

		large tests and loads can be generated. Highly extensible - forecast has an innovative and open architecture. forecast's open architecture enables support for new technology to be easily added. Live monitoring, metrics collection and control of test runs. Tests can be run from command line and batch files. Programming is not essential, forecast can be used without training or specialised knowledge. For advanced users, the full power of C++ or Java object-oriented (yet simple to use) interfaces are available for custom scripting. Support for interworking with common IDEs. Comprehensive statistical results analysis presented as graphs and charts. Supported Platforms *Central Controller:* Windows 2000. Windows XP (all versions). *Load Injectors:* Windows 2000. Windows XP (all versions). Windows NT.		

		Linux (Intel x86) e.g. Red Hat, Suse, DebianAIX Version 5. AIX 5.1 or later. Other systems on request. **The Central Controller** The Central Controller (forecast Studio) is used to record and create or generate test scripts, scenarios and test data. Manages test execution and monitoring. Gathers metrics and live test run data from the load injectors. **Load Injectors** A forecast load injector runs a compact service that will execute tests under the direction of the forecast Central Controller. Injector platforms include Windows XP (all versions), Windows 2000, Windows Server 2003, Windows NT, Linux, AIX and other platforms on request. The injectors service can run as a Windows service, a Unix daemon or a stand-alone process. Load injectors can be geographically dispersed, via firewalls or over the internet. A very lightweight TCP/IP connection is used between the Central Controller and the injectors, hence this can be over very low-bandwidth connections. **Script Recorders** Recorders for Web, TCP/IP, Database and Java applications. Automatic or manual transaction annotation.		

		Allows detailed visual inspection of the recording. Recorded traces can be edited and annotated. Test scripts are generated automatically from session recordings. For much testing, there is no requirement to edit the generated scripts. The script language is high-level and is very easy to use without advanced programming skills, but the full power of C++ or Java can be employed if required. Per-Virtual User data, such as usernames and form fields can be easily applied to the scripts so each Virtual User performs different actions. **Test Definition** Supports complex work-mixes with multiple scripts, groups of Virtual Users and varying data. Fine control over the iterations, errors, exceptions and execution of scripts. Fine control of start up delays (ramp-up) and pauses (think-times) per Virtual User. Advanced test data features e.g. per user, group, test, test iteration or injector. Comprehensive logging and tracing options. Tests can be distributed across multiple injectors of different types (Win32, Linux, AIX). **Test Execution and Monitoring** Efficient fully compiled scripts for the target injector. Test runs can be executed from command line or batch files.		

		All logs and results are optionally copied to the Central Controller. Live metrics graphs and progress status of each Virtual User. For selected Virtual Users, a real-time display of requests and server response pages. Live and completed test runs can be compared. Live display of pass/fail events, progress points, transaction timers, warnings and errors. Detailed logs of every Virtual User. **Results Analysis** forecast's Statistics Workshop analyses completed test runs and produces comprehensive graphs and statistical reports. Full control over the results output and display styles. Graphs and charts can be exported as HTML pages. Statistics data can be exported to other databases or spreadsheet tools such as Excel.		

VENDOR:	***OCLC INC***
TOOL NAME:	**WebArt [no longer on internet?]**

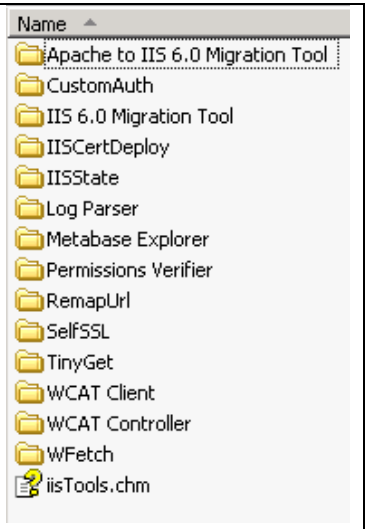
Web Page		http://www.oclc.org/webart/		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Medium-cost web test tool from OCLC Inc. includes load testing capabilities; also includes functional and regression testing capabilities, link-checking, and capture/playback and scripting language. Evaluation copy avail. Works with a wide variety of platforms. http://www.oclc.org/research/researchworks/contest/judges.htm ***Tip House*** Tip House is OCLC's chief systems analyst. He has twenty-five years of experience in software development, testing, and quality assurance. Tip is a Certified Quality Analyst, Certified Software Quality Engineer, and trained Lead Ticket Auditor. He is the creator of numerous tools for test automation and software configuration, including the WebART tool. Together with Lisa Crispin, Tip wrote *Testing Extreme Programming*. He also has authored numerous papers and presentations on software testing, software measurement, electronic document control/collaboration, and XP		WEB PAGE NO LONGER AVAILABLE CANT FIND THE TOOL ON THE INTERNET ? Not Evaluated

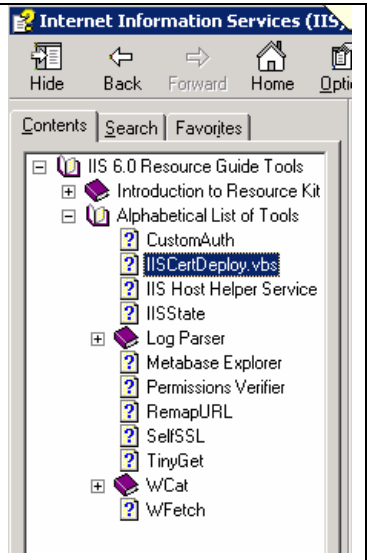
VENDOR:		***CRYANO***		
TOOL NAME:		**TestStream**		
Web Page		**http://www.azimith.com/index2.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Health care industry web applicaton performance testing tool		Not Evaluated

VENDOR:		*AZIMITH*		
TOOL NAME:		**Test Stream**		
Web Page		**http://www.azimith.com/index2.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		TestStream® by Cyrano, Inc. is the industry-renowned solution for the automated validation and capacity testing of HIS applications, uniquely providing for dramatic increases in user confidence around system integrity, performance reliability and management, deployment timelines, and patient safety 		Not Evaluated

OPEN SOURCE, SHAREWARE and FREEWARE VENDORS.....

VENDOR:		***MICORSOFT***		
TOOL NAME:		**Web Capacity Analysis Tool [Free]**		
Web Page		**http://www.microsoft.com/downloads/details.aspx?FamilyID=56fc92ee-a71a-4c73-b628-ade629c89499&DisplayLang=en**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		The following Web stress tools are available from Microsoft: • Web Capacity Analysis Tool The IIS 6.0 Resource Kit Tools include WCAT 5.2. To download the IIS 6.0 Resource Kit Tools, visit the following Microsoft Web site: http://www.microsoft.com/downloads/details.aspx?FamilyID=56fc92ee-a71a-4c73-b628-ade629c89499&DisplayLang=en • Web Application Stress Tool To download the Web Application Stress Tool, visit the following Microsoft Web site: http://www.microsoft.com/downloads/details.aspx?FamilyID=e2c0585a-062a-439e-a67d-75a89aa36495&DisplayLang=en	Free with unlimited users	The beginning of the installation was quite simple but it did not put an ICON on the desk top and there was no clue how to run the application from the file it was installed into.

		• Inetmonitor To download Inetmonitor, visit the following Microsoft Web site: http://www.microsoft.com/siteserver/site/DeployAdmin/InetMonitor.htm With these tools you can stress test your Web server to see how it reacts when several hundred users access your application at peak times. These tools can also be used to test the server-side components for performance, locks, and other scalability issues. A Web application that relies on databases can also be tested on parameters such as concurrency, transactions, number of users, locks, pooling, and so forth.		 Most of these folders in the install file directory have an EXE inside them. There is no SETUP or START or application ICON to tell you where the entry point is at. It does have a help file as the last folder in the directory which explains the application in each folder but it still was not much help.

				 After a lot trying I just gave up. This tool is likely more for System Engineers not Test Engineers.

VENDOR:	***WEBSERVICES.ORG (UK)***
TOOL NAME:	**DieselTest [FREE - Open Source]**

Web Page		http://sourceforge.net/projects/dieseltest/ **also** http://www.webservices.org/index.php/ws/content/view/full/45894		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Dieseltest is an easy-to-use load testing tool for simulating hundreds or thousands of users on a website. It has a facility for recording scripts, playing them back, and showing the results in real time. Results can be exported for further analysis. Contains the high-end features common to packages costing $50,000 or more. Dieseltest is a Windows application that simulates hundreds or thousands of users hitting a website. To run a load test, you first create a test script using our script editor. The script contains all of the requests that a real-world user would make of a website. You then load the script and run the test. The system will show you real-time results while the script is running, and produce a report analyzing the results at the conclusion. Dieseltest is an easy-to-use load testing tool for simulating hundreds or thousands of users on a website. It has a facility for recording scripts, playing them back, and showing the results in real time. Results can be exported for further analysis. Development Status: 5 - Production/Stable Intended Audience: Developers, System Administrators License: GNU Library or Lesser General Public License (LGPL) Operating System: 32-bit MS Windows (95/98), 32-bit MS Windows (NT/2000/XP), All 32-bit MS Windows (95/98/NT/2000/XP), Win2K, WinXP Programming Language: Delphi/Kylix Topic: Site Management, Software Development, Benchmark Translations: English	Open Source	Not Evaluated UK

		User Interface: Win32 (MS Windows) Contact us **WebServices.Org Contact page** **Chief Editor: Dr. Colin Adam**  Colin is an independent Technical Architect specialising in distributed systems design. Over the last five years Colin has been involved in several Web services projects with leading UK financial services companies and has indepth knowledge of enterprise architectures and several programming langauges. Colin is a regular author and conference speaker and holds the position as chief editor of WebServices.Org. Colin graduated from Edinburgh University with a PhD in Theoretical Physics and a Masters in Advanced Computer Science. Contact submissions@webservices.org for news Contact support@webservices.org for technical support		

VENDOR:	***A DEVELOPER GROUP***
TOOL NAME:	**The Grinder [FREE - Open Source]**

Web Page		http://grinder.sourceforge.net/		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		The Grinder is a Java™ load-testing framework. It is freely available under a BSD-style open-source license. The Grinder is a Java load-testing framework making it easy to orchestrate the activities of a test script in many processes across many machines, using a graphical console application. Requirement: OS Independent **What is The Grinder?** The Grinder is a Java™ load-testing framework. It is freely available under a BSD-style open-source license. The Grinder makes it easy to orchestrate the activities of a test script in many processes across many machines, using a graphical console application. Test scripts make use of client code embodied in Java *plug-ins*. Most users of The Grinder do not write plug-ins themselves, instead they use one of the supplied plug-ins. The Grinder comes with a mature plug-in for testing HTTP services, as well as a tool which allows HTTP scripts to be automatically recorded. 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 and reworked it to create **The Grinder 2**. Philip continues to enhance and maintain The Grinder, and welcomes all contributions. Recently Peter, Philip and Ted Osborne have published the book J2EE Performance Testing which makes extensive use of The Grinder. The next major version of The Grinder, The Grinder 3 is currently available as a beta quality release. The Grinder 3 uses the powerful scripting language Jython, and allows any Java code to be tested without the need to write a plug-in.	Open Source	

		The latest news, downloads, and mailing list archives can be found on SourceForge.net. **CREDITS:** I wish to thank Paco Gómez and Peter Zadrozny for the key ideas embodied in the original version of The Grinder. I am grateful to VA Software for SourceForge.net. This site was put together with txt2html, XEmacs and Apache Forrest. **Philip Aston**		

VENDOR:		***THE APACHE JAKARTA PROJECT***		
TOOL NAME:		**Apache JMeter [FREE - Open Source]**		
Web Page		**http://jakarta.apache.org/jmeter/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: Apache JMeter is a 100% pure Java desktop application designed to load test functional behavior and measure performance. It was originally designed for testing Web Applications but has since expanded to other test functions. Apache JMeter may be used to test performance both on static and dynamic resources (files, Servlets, Perl scripts, Java Objects, Data Bases and Queries, FTP Servers and more). It can be used to simulate a heavy load on a server, network or object to test its strength or to analyze overall performance under different load types. You can use it to make a graphical analysis of performance or to test your server/script/object behavior under heavy concurrent load. Requirement: Solaris, Linux, Windows (98, NT, 2000). JDK1.4 (or higher).	Open Source	

VENDOR:		*A GROUP OF DEVELOPERS*		
TOOL NAME:		**Opensource Test Suite (DOTS) [FREE - Open Source]**		
Web Page		http://ltp.sourceforge.net/		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: The Database Opensource Test Suite (DOTS) is a set of test cases designed for the purpose of stress-testing database server systems in order to measure database server performance and reliability. Requirement: Linux, POSIX	Open Source	DATABASE TESTING

VENDOR:		*D B MONSTER*		
TOOL NAME:		**DBMonster [FREE - Open Source]**		
Web Page		http://dbmonster.kernelpanic.pl/		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: DBMonster is an application to generate random data for testing SQL database driven applications under heavy load.	Open Source	

Requirement:

OS Independent

Welcome to dbMonster's cave!

dbMonster is a tool which helps database application developers with tuning the structure of the database, tuning the usage of indexes, and testing the application performance under heavy database load. dbMonster generates as much random test data as you wish and puts it into SQL database. It provides a very pluggable interface and is trivial to use. dbMonster is written in Java.

To learn how to use and how to extend dbMonster look at the manual. If you are plan to extend dbMonster you may be also interested in dbMonster's API.

Project's development page is located at http://sourceforge.net/

dbMonster subprojects

dbmonster-core	the dbMonster's engine which provides data generation framework and console mode laucher
dbmonster-ant	dbMonster ant integration project aims at providing full featured ant task for running dbMonster

Help wanted!

Although it works dbMonster is still in development stage. So far it can fill databases only with random data and with data which come from supplied dictionaries. That is far from ideal which is feeding the databases with realistic data. To become mature product dbMonster needs also a lot of testing. Therefore we are looking for volunteers who could help in the following areas:

collecting a dictionaries of realistic data (names, companies names, posts,

		titles, and much much more), testing the aplication, sharing the opinion, proposing patches, writing test cases (JUnit), writing various (i.e. database specific) data and key generators that will add more functionality. If you want to help email me (remove the fake part of the address): ant@CUT-IT-OFF.kernelpanic.pl or subscribe to developers mailing list at sourceforge.net. Quick tutorial/case study on how to use dbMonster by Pablo: http://geocities.com/firebird_tut/files/FEEDING_YOUR_DATABASE.pdf Statistical analyzis concerning randomness in dbMonster by Michael Wall http://www.mjwall.com/node/view/133		

VENDOR:		***DELUGE***		
TOOL NAME:		**Deluge [FREE - Open Source]**		
Web Page		**http://deluge.sourceforge.net/**		
Web Testing	**Functional Testing**	Feature pro/cons	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: An open-source web site stress test tool. Simulates multiple user types and counts. Includes proxy server for recording playback scripts, and log evaluator	Open Source	

		for generating result statistics. **Requirement:** OS independent **Overview** This is the project homepage for Deluge, an open-source web site stress test tool. Deluge was originally developed for the launch of IdeaForest.com by Grue at Thrown Clear Productions. After the launch, the code was released under the GPL, and the stress test report was made available for public reading. (IdeaForest.com has since been bought by Jo-Ann Stores, and the original site is now unavailable.) **Documentation** Here's how to learn more about Deluge: The stress test report for IdeaForest.com in PDF format. An example DCF (Deluge Config File), to see how the system is controlled. **Status** The latest stable version of Deluge is 0.9.12. It's available through the Deluge Project page at SourceForge, listed below. **Project** Deluge is now maintained by SourceForge: The Deluge project group page at SourceForge. The CVS respository, for the latest source code.		

		Support Most of the support for Deluge is done through the mailing lists: Deluge announcements - a low volume list to announce updates. Deluge users - for getting and giving help, making suggestions, and so on. Deluge development - for those people fixing bugs and adding features.		

VENDOR:		***DIESELTEST***		
TOOL NAME:		**Dieseltest [FREE - Open Source]**		
Web Page		**http://sourceforge.net/projects/dieseltest/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: Contains the high-end features common to packages costing $50,000 or more. Dieseltest is a Windows application that simulates hundreds or thousands of users hitting a website. To run a load test, you first create a test script using our script editor. The script contains all of the requests that a real-world user would make of a website. You then load the script and run the test. The system will show you real-time results while the script is running, and produce a report analyzing the results at the conclusion. Requirement: Windows	Open Source	

		Dieseltest is an easy-to-use load testing tool for simulating hundreds or thousands of users on a website. It has a facility for recording scripts, playing them back, and showing the results in real time. Results can be exported for further analysis. Development Status: 5 - Production/Stable Intended Audience: Developers, System Administrators License: GNU Library or Lesser General Public License (LGPL) Operating System: 32-bit MS Windows (95/98), 32-bit MS Windows (NT/2000/XP), All 32-bit MS Windows (95/98/NT/2000/XP), Win2K, WinXP Programming Language: Delphi/Kylix Topic: Site Management, Software Development, Benchmark Translations: English User Interface: Win32 (MS Windows) Project UNIX name: dieseltest Registered: 2001-07-10 21:02 Activity Percentile (last week): 79.72 View project activity statistics View list of RSS feeds available for this project		

VENDOR:	***HAMMERHEAD***
TOOL NAME:	**Hammerhead 2 - Web Testing Tool [FREE - Open Source]**
Web Page	**http://hammerhead.sourceforge.net/**

Web Testing	Functional Testing	Feature pro/cons	Aprox Pricing	Our Evaluation / Rating
		Description: Hammerhead 2 is a stress testing tool designed to test out your web server and web site. It can initiate multiple connections from IP aliases and simulated numerous (256+) users at any given time. The rate at which Hammerhead 2 attempts to pound your site is fully configurable, there are numerous other options for trying to create problems with a web site (so you can fix them). **Requirement:** Hammerhead has been used with Linux, Solaris and FreeBSD. **Hammerhead 2 - Web Testing Tool** **What does it do?** Hammerhead 2 is a stress testing tool designed to test out your web server and web site. It can initiate multiple connections from IP aliases and simulated numerous (256+) users at any given time. The rate at which Hammerhead 2 attempts to pound your site is fully configurable, there are numerous other options for trying to create problems with a web site (so you can fix them). It can be used to test the behaviour of the port under load, or the ability of the port to service a set of requests. **Warning:** Hammerhead can destroy a web site very quickly (and cost you lots of money if you pay for traffic you generate). It should only be used to generate heavy loads on local web sites for testing purposes. **Using Hammerhead** The behaviour of Hammerhead is very configurable. Hammerhead loads a set of requests from a number of files, each of which may contain a number of scenarios (one request per scenario). Scenarios may be linked into sequences, in order to simulate real users actions. **Generating Load** Once the specified scenarios are loaded, Hammerhead starts a number of threads and sends requests to the configured port. Hammerhead can simulate	Open Source	

		requests from multiple machines by exploiting IP aliasing. Hammerhead can generate a large amount of load on a web server by having a large number of sessions (200+), an IP alias for each session and setting the time between requests to be small (in the order of a second) Because Hammerhead (generally) waits for replies from the web server it will load the web server to capacity and can be used for capacity testing. **Testing** Expected results of an HTTP request may be specified in a scenario, and any result from the port which does not match with the expected result will be reported as an error. If no expected results are specified, then any result from the port will be accepted as valid. Specifying an expected result for any scenario has the effect of specifying the same result for all scenarios which have the same request. Any failure to get a connection to the port, or any failure to receive a reply to a request will also be reported. **The Source Code** The Hammerhead source code is available under the GNU Public License: Download Page **Platforms** Hammerhead has been used with Linux, Solaris and FreeBSD. It would should be relatively easy to port to any Unix environment. **Limitations** Trying to truly simulate users on the web is very difficult. Hammerhead has a number of inherent limitations. Primarily it is limited by the number of IP aliases you can add to your machine and still have it perform reasonably. Typically a big web site will get lots more users than you can generate IP aliases. Secondly Hammerhead (normally) waits on the web server to respond once it has a connection; this can cause Hammerhead to fall below the request rate you've configured for if the web server is running poorly. Really Hammerhead should disconnect from a slow connection and force a "reload" like a "real" user might do. Also data is extracted at full rate from any connections that have been made. Ideally you'd want to be able to control this rate to simulate users on modems (etc). Mick Dwyer has just begun to add in bottleneck simulation code to address some of these concerns but this is		

		currently incomplete. --- *Geoff Wong* *geoff@shattered.org* *December-2000.*		

VENDOR:		***ACME***		
TOOL NAME:		**http_load [FREE - Open Source]**		
Web Page		**http://www.acme.com/software/http_load/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: http_load runs multiple HTTP fetches in parallel, to test the throughput of a Web server. However, unlike most such test clients, it runs in a single process, to avoid bogging the client machine down. It can also be configured to do HTTPS fetches **Requirement:** tbc **http_load - multiprocessing http test client** Fetch the software. http_load runs multiple http fetches in parallel, to test the throughput of a web server. However unlike most such test clients, it runs in a single process, so it	Open Source	

		doesn't bog down the client machine. It can be configured to do https fetches as well. You give it a file containing a list of URLs that may be fetched, a flag specifying how to start connections (either by rate or by number of simulated users), and a flag specifying when to quit (either after a given number of fetches or a given elapsed time). There are also optional flags for checksums, throttling, random jitter, and progress reports. See also: http_ping, http_get, thttpd. A page of other http load-test tools. Back to ACME Labs Software. Back to ACME Labs. ACME Labs Webmaster <webmaster@mail.acme.com>		

VENDOR:		***THE DEVELOPMENT GATEWAY FOUNDATION***		
TOOL NAME:		**JCrawler [FREE - Open Source]**		
Web Page		**http://jcrawler.sourceforge.net/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: Stress-Testing Tool for web-applications. It comes with the crawling/exploratory feature. You can give JCrawler a set of starting URLs and it will begin crawling from that point onwards, going through any URLs it can find on its way and generating load on the web application. The load parameters (hits/sec) are configurable. **Requirement:**	Open Source	

		OS Independent **The What.** JCrawler is an open-source (under the CPL) Stress-Testing Tool for web-applications. It comes with the crawling/exploratory feature. You can give JCrawler a set of starting URLs and it will begin crawling from that point onwards, going through any URLs it can find on its way and generating load on the web application. The load parameters (hits/sec) are configurable. **The Why.** But, wait a second! Aren't there already a whole bunch of tools like that? Why would anybody write a new one? You could bet there are a number of such programs in the open-source and definitely have to be some kick-ass commercial ones. Well, that's what we thought, too. Frankly, we had no desire to write a load-tester tool. We are writing a web-portal system (http://www.digijava.org) not load-testing tools. But then we had a problem with one of our portlets that would only occur on the production server, during a high load, and none of the existing tools we tried was able to recreate it. Log-replay tools were not much help either, because the problem would occur in several hours and we needed some tool to really stress the application so it would crash in more reasonable (i.e. less) time. We spent a lot of time trying not to "reinvent the wheel" and find an existing wheel that would help us. There was none. We tried both OSS and commercial tools. None of them gave us the kind of result we needed. So we ended up with the JCrawler. JCrawler was irreplaceable in helping us to identify and solve the problem we had. We have released it in under an open source license because we hope it may help somebody else, too and that somebody will not have to go through what we have gone. Also, it may be a good chance for the JCrawler itself to get enhancements. We are very open to the suggestions and especially - help :) We continue to use JCrawler for testing our applications and would not		

		mind, of course, it to get as good as it can. **The How.** What features were missing in some of the similar tools, and what is the bundle that was in none of them together? Why did not any of them work for us? **Crawling -** A lot of load-test tools allow you to indicate a set of URLs and they would just hit these URLs repeatedly. For a complex web-application it may not make much sense. There are too many URLs, for testing any limited number of them to give much confidence. You may hope that indicating "typical" ones may help, but if you use caching and stuff - the one "typical" URL will soon be cached, whereas in the real life hitting 10 different ones like that would have a surprisingly different effect. **"Human" pattern** - Most of the existing tools allow you to indicate how many threads they start up for load-testing. This isn't always an accurate way to replicate real load. People measure performance in the terms of the number of hits per second on a website, not some geeky "number of threads". Having 200 threads hit your site does not mean that you are generating 200 hits/second. The load tester tool may be generating just 2 hits per second with that setting, if your application pages are slow. Threads have to wait for the pages to load and not all pages load in 1/200th of a second, for example. We wish they did, of course but that wish is often far from the reality. JCrawler follows hits-per-second pattern, guaranteeing indicated load and will fire-up as many threads, as needed to keep the load constant. **Http Redirects and Cookies** - Some of the tools were not able to properly handle it. These can leave your application's authentication completely untested and give you another set of surprises in the production. This is especially true if you are using a single-sign-on system of some kind, which usually employs transparent HTTP Redirects. **OSS and tested** - JCrawler is open-source and comes with thorough unit-tests. So, one has reasonable confidence that it is not buggy itself, as well as the ability to fix, customize, or enhance it. Some tools we tried (especially, proprietary ones, of course) - we could not even understand how exactly they worked and it was not clear if they were doing what the documentation claimed they would .		

		What are the features that may not be critical but we like having in the JCrawler? **Console mode** - All these graphs and pie-charts look cool and can help for a presentation to the management but when you have a real problem to solve they are not the ones that matter. JCrawler is easy to run remotely and monitor, using little bandwidth. It can be very useful if your testing point is secured i.e. if you can access your testing environment only from a limited network, hence your test tool has to be in that network, too and the only access you get is SSH. Also, GUI-based applications have this tendency of hanging up on you when application gets real busy, which can quickly become quite annoying. And a load-tester is definitely one very busy application. **Easy to configure -** Easy to configure - The entire configuration happens in a central XML file. You can keep different XML files in a handy place and have several configurations ready to go, whenever you need them. It may be just us, but we find using a neat XML file more convenient then jumping from one tab to another of an overloaded GUI configuration. (Also, by using an XML formatted file from the start, it will be easier to create a GUI configuration tool if others don't agree with us on the convenience of this.) **Platform independent -** JCrawler is a tool for the developers and QAs. In our team, we found that people prefer different operating systems so having tool that runs on any of them was nice. 		

VENDOR:	***ADAM SLOSARSKI (Poland)***
Tool Name:	**Ntime [FREE - Open Source]**
Web Page	**http://www.codeproject.com/dotnet/ntime.asp also http://www.polbox.com/a/aslo**

Web Testing	Functional Testing	Feature pro/cons	Aprox Pricing	Our Evaluation / Rating
		Description: The NTime tool is very similar to NUnit tool to perform repeatable tasks that help managers, architects, developers and testers to test an application against its performance. **Requirement:** Windows 98 or above, .Net framework 1.1 or 2.0 **Introduction** This article presents a unique tool to run repeatable performance tests in actually developed application by programmers. The NTime tool is very similar to NUnit tool - another unit testing tool, so currently users that use NUnit should see almost the same GUI and functionality. **Background** This is the first release of the tool to perform repeatable tasks that help managers, architects, developers and testers to test an application against its performance. This version may work inconsistent and report some errors in different situations that may exist while running this tool, however it was tested for many reasons. **Tool Usage** This section provides information for application management team that wishes to use NTime tool to keep applications working at desired performance. **Managers** Program managers are welcome to specify application performance boundaries at the beginning of the application development process. They may note which modules and functions of application have to work at which time frames. Their written performance specifications may be helpful when designing application performance by developers. Further, NTime tool may	Open Source	Not Evaluated

		test their performance specification against real, developed applications. Those tests performed by NTime tool may be reproduced in different time periods to see whether their application is working at expected performance. But the main advantage of using this tool by managers is to create configuration test files for developers and testers to give them a starting point on how to optimize the application. **Developers** Developers use NTime tool to test whether their application works with performance specification designed by managers or sometimes by themselves. They need to run NTime tool when they aren't sure whether their rewritten code still works well. NTime tool may have been open always on the desktop even when programmers develop, compile or debug application -- every application build will be reloaded in NTime automatically and shown with updated tests. **Testers** This team will be the often role in performance test stage. They will run repeatable tests after each application alpha release. Testers will review hundreds or thousands of tests and register them to bug tracing applications while tests reflect to be failed. **Architecture** NTime architecture is similar to NUnit tool, therefore some implementations that were introduced in NUnit may be similar to its users. This section shows features found in NTime tool. **Simultaneous Compile and Test Time** NTime tool may still be running while there is currently a compilation process of the application which we want to test. After application has been built, NTime will reflect all changes, you just need to run tests and wait for performance test results whether they are accepted or rejected. **Test Timings** Actually, NTime's timings are high frequency timers that will rely on different onboard chipsets' architecture. This difference will generate more or less		

		precision in tests. The one disadvantage while profiling tests is system processes and CPU load. Although you've closed heavy loaded applications, there may be other bad things you may meet -- NTime needs some CPU and threads to perform complex tests and therefore you can indicate about 10 microseconds overhead, so if you want to test high frequency running functions, then write inside of test code "for" loop statement i.e. 100 loops, and then test the result as specified time multiplied by 100. **SDK Documentation** NTime tool is supported with two documentation files. The NTime Framework SDK will be your starting point to test your applications against performance tests by adding .NET attributes into your code. **NTime Framework SDK** NTime Framework SDK will be your often SDK documentation. This help file shows you available .NET attributes you can use in your testing applications. Many attributes are similar to NUnit attributes, so when you've used NUnit tool before, then starting to work with NTime will be easy. **NTime GUI SDK** After NTime installation, you will find NTime source documentation. This will help you add some functionality to NTime for your own, but remember that every change you've made to this tool is for your private and non-commercial use. Read *License.rtf* file for further information about NTime licensing. **Further Development** This NTime version is a GUI only application. I have in future plans to develop a console application to provide batch script usability. For better NTime tool improvements, you may send comments to adamslosarski@tlen.pl with your suggestions to help me get the application work better. Bugs found in this release will be fixed in the next version, so earlier found errors will be corrected as fast as possible. NTime sources will be available soon on the sourceforge.net website, so you can track the next versions. **Points of Interest** This tool contains some .NET technologies used inside the code, including		

		application domains and .NET remoting - these technologies were introduced to allow simultaneous compile and run test functionality, so assemblies may be loaded into separate application domains which run shadowed assemblies, so the original ones may be overwritten by compiler, also assemblies can be unloaded from application domain anytime. Second thing is a file monitoring which tracks file changes and refreshes project test tree to reflect changes in assemblies when a new DLL is deployed. **History** 2005-04-05 **Bug fixes:** Tested assemblies could not read application configuration settings. *NTimeSetup.exe* installer failed when source files were being selected for installation. **Improvements:** New option dialog box allows to specify application configuration settings file. *NTime.Framework.dll* was promoted to GAC to allow better shared reference to it. 2004-12-08 - Updated NTime application with bug fixes and additional console mode. 2004-05-14 - Added source code and documentation download separately. 2004-05-11 - First release of NTime tool. **About Adam Slosarski**		

Born in Poland, living there as employeed developer, in free time writing much .net stuff and designing applications.

Click here to view AdamSlosarski's online profile.

Other popular articles:

The 30 Minute Regex Tutorial
Learn how to use regular expressions in 30 minutes with Expresso.

Edit Almost Anything in a DataGrid
Allow your DataGrid to edit pictures and a whole range of other data types.

Expresso - A Tool for Building and Testing Regular Expressions
For learning, building, and debugging .NET Framework regular expressions

Floating Point in .NET part 1: Concepts and Formats
Introducing the basic concepts of floating-point arithmetic: number formats, accuracy and precision, and round-off error. Includes a discussion of the .NET floating-point types.

C++ (VC7, VC7.1), C#, VB.NET
Windows (Win2003, NT4, Win2K, WinXP, Win95, Win98, WinME),
.NET (.NET 1.0, .NET 1.1)
Win32, VS
Dev

Posted 11 May 2004 0:00

Updated 5 Apr 2005 7:45

Articles by this author

VENDOR:		*OPEN LOAD*		
TOOL NAME:		**OpenLoad [FREE - Open Source]**		
Web Page		**http://openload.sourceforge.net/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: OpenLoad is a tool for load testing web applications. It aims to be easy to use and providing near real-time performance measurements of the application under test. Requirement: Linux, DOS **What is OpenLoad?** OpenLoad is a tool for load testing web applications. It aims to be easy to use and providing near real-time performance measurements of the application under test. This is particulary useful when you are doing optimization as you can see the impact of your changes almost immediately. **News** 07-jul-2001: First UI Design A first attempt of designing an UI for OpenLoadhas been made. Have a look HERE. A GUI design discussion has been started HERE. 02-jul-2001: Release Plan Updated The release plan has been updated after the 0.1.2 release. It is available in the developer area, or directly from HERE. 27-jun-2001: Ver. 0.1.2 Released This release adds a number of new command line options for specifying request headers, setting a time limit, doing test runs which displays the HTTP response, and more.	Open Source	Not Evaluated

		Download You can download the latest release of OpenLoad from SourceForge http://sourceforge.net/project/showfiles.php?group_id=21343 . **Installation** Linux For i386 systems you can use the binary rpm file. Download the file and install using: rpm -i <filename> This installs openload in /usr/local/bin. For other systems download the .tar.gz file, unpack it using: tar xvzf <filename> and compile using the normal procedure: ./configure make make install see the INSTALL file for details. **Win32** Download the executable (openload.exe) and save it in a directory which is in your path. You can now start OpenLoad from any command (DOS) window. **How do I use it?** OpenLoad is (currently) a commandline tool,		

VENDOR:	***WEB POLYGRAPH***
TOOL NAME:	**Web Polygraph [FREE - Open Source]**
Web Page	**http://www.web-polygraph.org/**

Web Testing	Functional Testing	Feature pro/cons	Aprox Pricing	Our Evaluation / Rating
		Description: BENCH MARKING TOOL Benchmarking tool for caching proxies, origin server accelerators, L4/7 switches, content filters, and other Web intermediaries. Requirement: C++ compiler Web Polygraph is a freely available benchmarking tool for caching proxies, origin server accelerators, L4/7 switches, content filters, and other Web intermediaries. Polygraph's features include: • high-performance HTTP clients and servers • realistic HTTP and SSL traffic generation • flexible content simulation • ready-to-use standard workloads • powerful domain-specific configuration language • portable open-source implementation **WORK LOADS** Web Polygraph is capable of simulating numerous workloads. Here we just give a few useful examples, including standard workloads. Note that while overall description of a workload does not depend on Polygraph version, the corresponding PGL configuration may change as the benchmark evolves. You have a choice of using the version of Polygraph specified in the workload description or adjusting the given configuration for newer versions of Polygraph. Workload names usually end with a number (e.g., *PolyMix-3* or *WebAxe-4*). That number represents "generation" of the workloads. Usually, workloads from the same generation have similar base features and are supported by the	Open Source	Not Evaluated BENCH MARKING TOOL

		same Polygraph version. If possible, use the latest generation of workloads. If you are looking for Web Polygraph alternatives or complements, here are a few web sites that collect related testing tools. • Open Source Testing • Web Caching Testing Tools • ApTest Testing Resources • Better Software Tools Collection		

VENDOR:		***OPEN SYSTEM TESTING ARCHITECTURE***		
TOOL NAME:		**OpenSTA [FREE - Open Source]**		
Web Page		**http://OpenSTA.org/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: A distributed software testing architecture based on CORBA. Using OpenSTA (Open System Testing Architecture) a user can generate realistic heavy loads simulating the activity of hundreds to thousands of virtual users. OpenSTA graphs both virtual user response times and resource utilization information from all Web Servers, Application Servers, Database Servers and Operating Platforms under test, so that precise performance measurements can be gathered during load tests and analysis on these measurements can be performed. **Requirement:** Windows 2000, NT4 and XP **Community Supported** **Development Driven by the Users**	Open Source	Not Evaluated

		Much more information can be found out about OpenSTA by checking the online documentation or simply downloading and installing the toolset. The FAQ contains lots of other useful background information and helpful tips, this should be the first place you look if you need help with anything not covered in the documentation. There is no need to stop at reading the FAQ either, it is hosted on the OpenSTA Community Portal and, in common with every other resource on this site, it is user editable. This site is a great place for every OpenSTA user to share their experiences with the product and help others learn and use OpenSTA: Remember, the toolset is completely free and any time the developers spend helping users is time they are not enhancing, or fixing problems with, the toolset. By helping other users you are in fact helping OpenSTA and its community become stronger. The premier place for free OpenSTA support and discussions is the OpenSTA Users Mailing List, here the developers and many long time users of this toolset give as much help as their freetime will allow		

VENDOR:		***SIEGE***		
TOOL NAME:		**Siege [FREE - Open Source]**		
Web Page		**http://www.joedog.org/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: SIEGE is an http regression testing and benchmarking utility. It was designed to let web developers measure the performance of their code under duress, to see how it will stand up to load on the internet. It lets the user hit a web server with a configurable number of concurrent simulated users. Those users place the web server "under siege." SCOUT surveys a web server and prepares the urls.txt file for a siege. In order to perform regression testing, siege loads URLs from a file and runs through them sequentially or randomly. Scout makes the process of populating that file easier. You should send out the scout, before you lay siege.	Open Source	BENCH MARK UTILITY Not Evaluated

		Requirement: GNU/Linux, AIX, BSD, HP-UX and Solaris.		

VENDOR:		***SIPp***		
TOOL NAME:		**Sipp [FREE - Open Source]**		
Web Page		**http://sipp.sourceforge.net/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: Sipp is a performance testing tool for the SIP protocol. Its main features are basic SIPStone scenarios, TCP/UDP transport, customizable (xml based) scenarios, dynamic adjustement of call-rate and a comprehensive set of real-time statistics. Requirement: All POSIX (Linux/BSD/UNIX-like OSes), Linux, HP-UX **1. Welcome to SIPp** SIPp is a free Open Source test tool / traffic generator for the SIP protocol. It includes a few basic SipStone user agent scenarios (UAC and UAS) and establishes and releases multiple calls with the INVITE and BYE methods. It can also reads custom XML scenario files describing from very simple to complex call flows. It features the dynamic display of statistics about running tests (call rate, round trip delay, and message statistics), periodic CSV statistics dumps, TCP and UDP over multiple sockets or multiplexed with retransmission management and dynamically adjustable call rates. Other advanced features include support of IPv6, TLS, SIP authentication, conditional scenarios, UDP retransmissions, error robustness (call timeout, protocol defense), call specific variable, Posix regular expression to extract and re-inject any protocol fields, custom actions (log, system command exec, call stop) on message receive, field injection from external CSV file to	Open Source	Not evaluated

		emulate live users. While optimized for traffic, stress and performance testing, SIPp can be used to run one single call and exit, providing a passed/failed verdict. Last, but not least, SIPp has a comprehensive documentation available both in HTML and PDF format. SIPp can be used to test many real SIP equipements like SIP proxies, B2BUAs, SIP media servers, SIP/x gateways, SIP PBX, ... It is also very useful to emulate thousands of user agents calling your SIP system.		

VENDOR:		***STRESS DRIVER***		
TOOL NAME:		**stress_driver [FREE - Open Source]**		
Web Page		**http://sourceforge.net/projects/stress-driver/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: General-purpose stress test tool. **Requirement:** Windows NT/2000, Linux **Features:** • Development Status: 3 - Alpha • Intended Audience: Quality Engineers • License: Other/Proprietary License • Operating System: Cygwin (MS Windows), All POSIX (Linux/BSD/UNIX-like OSes), OS X • Programming Language: Perl • Topic: Testing • Translations: English	Open Source	Not Evaluated

		• User Interface: Command-line Project UNIX name: stress-driver Registered: 2002-08-17 21:04 Activity Percentile (last week): 91.76 View project activity statistics View list of RSS feeds available for this project Need support? See the support instructions provided by this project.		

VENDOR:		***TP TEST (Sweden)***		
TOOL NAME:		**TPTEST [FREE - Open Source]**		
Web Page		**http://tptest.sourceforge.net/about.php**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Description: The purpose with TPTEST is to allow users to measure the speed of their Internet connection in a simple way. TPTEST measures the throughput speed to and from various reference servers on the Internet. The use of TPTEST may help increase the consumer/end user knowledge of how Internet services work. **Requirement:** MacOS/Carbon and Win32 **PURPOSE:** TPTEST was originally developed by the Swedish ICT-commission, then later by the Foundation for Internet Infrastructure (iis.se), the Swedish Consumer Agency (konsumentverket.se), and the Swedish National Post- and Telecom Agency (pts.se). The latest development has been to separate the platform-independent test method software (the test engine) from the platform-dependent user interface software in order to make it easier for anyone to	Open Source	Not Evaluated

		write a test client or server that uses the TPTEST testing method. The test engine code is to be regarded as a library module and is released under the LGPL license while the reference client/server applications is released under the GPL license. **TPTEST 3.0 does the following:** • Measures TCP throughput, incoming and outgoing • Measures UDP throughput, incoming and outgoing • Measures UDP packet loss, incoming and outgoing • Measures UDP roundtrip times • Measures UDP out-of-order packet reception For a UDP test you can select test time, the number of packets to be transferred, and the packet size. For a TCP test you can only select the number of bytes to be transferred. The Win32 and MacOS reference client applications include "auto" test modes that try to determine the performance of your particular Internet connection. The results are a good guesstimate for the not-so-technical user who wishes to see if s/he gets what s/he is paying for but for really accurate results you need to know what you're doing and do it manually. The auto modes determine TCP throughput by sending increasingly large amounts of data over the connection until the reception time is long enough (currently ten seconds) that the program judges the result to be fairly accurate. UDP throughput is determined by performing repeated 5-second tests with higher and higher data rates until the calculated reception speed doesn't increase anymore. UDP throughput is usually significantly higher than TCP throughput, due to the TCP algorithm backing off when it encounters a congestion, thus "making way" for UDP transmissions.		

VENDOR:	***VINCENT SHEFFER***
TOOL NAME:	**LoadSim - Web Application Load Simulator [FREE - Open Source]**
Web Page	**http://freshmeat.net/projects/loadsim/**

Web Testing	Functional Testing	Feature pro/cons	Aprox Pricing	Our Evaluation / Rating
		http://opensourcetesting.org/performance.php **Web Application Load Simulator** Description: LoadSim is a web application load simulator. It allows you to create simulations and have those simulations run against your webserver. Requirement: JDK 1.3 or above http://freshmeat.net/projects/loadsim/ **About:** LoadSim allows you to record a Web browser session and play it back with as many virtual users as your network and computing resources allow. A single simulation can be run on different machines (to simulate large numbers of virtual users) and managed from a single console. Raw results consist of time-to-first-byte (TTFB), time-to-last-byte (TTLB), size of file downloaded, a label for the link, and the content type. It also comes with basic statistical analysis support for calculating means, standard deviation, and histograms for the raw results. It supports datasets so you can realistically simulate users (or use LoadSim as a way to initialize a database) with different form data. **Author:** Vincent Sheffer [contact developer]	Open Source	Not Evaluated

VENDOR:	***KEYNOTE***
TOOL NAME:	**Test Perspective Load Test [FREE]**
Web Page	**http://www.keynote.com/**

Web Testing	Functional Testing	Feature pro/cons	Aprox Pricing	Our Evaluation / Rating
		Do-it-yourself load testing service from Keynote Systems for Web applications. Utilizes Keynote's load-generating infrastructure on the Internet; conduct realistic outside-the-firewall load and stress tests to validate performance of entire Web application infrastructure. Even the best Web sites are sometimes hard to use. It's easy to get lost or confused while trying to research an investment, buy a gift, review sales forecasts or complete a business transaction online. The 150,000 member Keynote Research Panel gives you the chance to speak out and make everyone's online experience more enjoyable. As a panelist, you'll evaluate leading Web sites and provide vital feedback on your experience. Join now and help improve the Web!	Free	Not Evaluated

VENDOR:		***OPENLOAD***		
TOOL NAME:		**OpenLoad [Free – Open Source]**		
Web Page		**http://openload.sourceforge.net/#what**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Open source load testing tool from SourceForge.net/Open Source Development Network, distributed under GNU General Public License (GPL). Command-line tool; runs on Linux and Win32 systems. This project aims to provide a tool for load testing web applications. The goal is a tool that is easy to use and provides near real-time performance measurements of the application under test which is very usefull during	Open Source	

optimization.

- Development Status: 4 - Beta
- Intended Audience: Developers, System Administrators
- License: GNU General Public License (GPL)
- Operating System: All 32-bit MS Windows (95/98/NT/2000/XP), All POSIX (Linux/BSD/UNIX-like OSes), Linux
- Programming Language: C++
- Topic: WWW/HTTP, Software Development, Benchmark
- Translations: English

OpenLoad is a tool for load testing web applications. It aims to be easy to use and providing near real-time performance measurements of the application under test. This is particulary useful when you are doing optimization as you can see the impact of your changes almost immediately.

OpenLoad is (currently) a commandline tool, that you execute from a prompt like this:

```
openload [options] http://testapp.site.com 10
```

The 2 parameters are:

- The url of the web page you want to test.
- Number of simultanous clients to simulate. This is optional and defaults to 5.
- A number of options is also available. See here for a detailed description of all the options.

You will then get output similar to this:

```
$ openload localhost 10
URL: http://localhost:80/
Clients: 10
MaTps 355.11, Tps 355.11, Resp Time  0.015, Err   0%, Count   511
MaTps 339.50, Tps 199.00, Resp Time  0.051, Err   0%, Count   711
MaTps 343.72, Tps 381.68, Resp Time  0.032, Err   0%, Count  1111
MaTps 382.04, Tps 727.00, Resp Time  0.020, Err   0%, Count  1838
MaTps 398.54, Tps 547.00, Resp Time  0.018, Err   0%, Count  2385
MaTps 425.78, Tps 670.90, Resp Time  0.014, Err   0%, Count  3072
```

		Total TPS: 452.90 Avg. Response time: 0.021 sec. Max Response time: 0.769 sec Where: • **MaTps:** a 20 second moving average of **TPS**. • **Tps:** (Transactions Per Second) is the number of completed requests during that second. • **Resp Time:** the average response time in seconds for the elapsed second. • **Err:** the percentage of responses that was erronous, i.e. didn't return a HTTP 200 Ok staus. • **Count:** the total number of completed requests. • **Total TPS** is the average TPS for the whole run, i.e. (Total completed requests) / (Total elapsed time). • **Avg. Response time:** the overall average response time in seconds. • **Max Response time:** the highest response time during this run. **Note:** you stop the run by pressing Enter.		
VENDOR:		***HEWLETT-PACKARD***		
TOOL NAME:		**Httperf [Free – Open Source]**		
Web Page		**http://www.hpl.hp.com/personal/David_Mosberger/httperf/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Web server performance/benchmarking tool from HP Research Labs. Provides a flexible facility for generating various HTTP workloads and measuring server performance. Focus is not on implementing one particular benchmark but on providing a robust, high-performance, extensible tool. Available free as source code. httperf---A Tool for Measuring Web Server Performance It provides a flexible facility for generating various HTTP workloads and for	FREE	

		measuring server performance. The focus of httperf is not on implementing one particular benchmark but on providing a robust, high-performance tool that facilitates the construction of both micro- and macro-level benchmarks. The three distinguishing characteristics of httperf are its robustness, which includes the ability to generate and sustain server overload, support for the HTTP/1.1 protocol, and its extensibility to new workload generators and performance measurements. In addition to reporting on the design and implementation of httperf this paper also discusses some of the experiences and insights gained while realizing this tool. **Source code:** ftp site. The httperf tool is available in source code form and free of charge from the following URL: ftp://ftp.hpl.hp.com/pub/httperf/		

VENDOR:		***AGILENT TECHNOLOGIES***		
TOOL NAME:		**NetworkTester [Free – Open Source]**		
Web Page		**http://advanced.comms.agilent.com/networktester/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Tool (formerly called 'NetPressure') from Agilent Technologies uses real user traffic, including DNS, HTTP, FTP, NNTP, streaming media, POP3, SMTP, NFS, CIFS, IM, etc. - through access authentication systems such as PPPOE, DHCP, 802.1X, IPsec, as necessary. Unlimited scalability; GUI-driven management station; no scripting; open API. Errors isolated and identified in real-time; traffic monitored at every step in a protocol exchange (such as time of DNS lookup, time to logon to server, etc.). All transactions logged, and detailed reporting available. **OVERIVEW:** NetworkTester provides Service Providers and Network Equipment Manufacturers with a powerful layer 4-7 test solution to accelerate development and deployment of network security and content-switching	Open Source	Not Evaluated

		devices.
		
		• NetworkTester Brochure
		• VoIP and IPv6 Flyer
		
		Key Features
		
		• **Broad range of protocol bricks**
		• **Client and server emulation** - one system, one user interface
		  o Simulate millions of real users and services
		• **Powerful "Test Plan" design and management environment**
		  o Set-up tests in minutes; no need for scripting!
		• **Stateful traffic over integrated IPSec**, **IPsecv6**, **PPPoE**, **DHCP** and **802.1x**
		  o Integrated access protocols for faster and easier test set-up
		• **Integrated VLAN support**
		  o Rapidly test VLAN-capable devices and virtual firewalls
		• **Transaction Variability and Real-Time Control**
		  o Randomize and cycle parameters such as address lists, and attach real files
		  o Dynamically change parameters while the test is running
		• **Integrated IPv6 support**
		  o Rapidly test next-generation IPv6 and IPsecv6 devices and networks
		• **Over 150 real-time statistics**
		  o Correlate measurements and characterize system performance		

VENDOR:		***SIEGE***		
TOOL NAME:		**Stress Tester [Free – Open Source]**		
Web Page		**http://joedog.org/siege/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Open source stress/regression test and benchmark utility; supports basic authentication, cookies, HTTP and HTTPS protocols. Enables testing a web server with a configurable number of concurrent simulated users. Stress a single URL with a specified number of simulated users or stress multiple URL's simultaneously. Reports total number of transactions, elapsed time, bytes transferred, response time, transaction rate, concurrency, and server response. Developed by Jeffrey Fulmer, modeled in part after Lincoln Stein's torture.pl, but allows stressing many URLs simultaneously. Distributed under terms of the GPL; written in C; for UNIX and related platforms. **ABOUT SIEGE** Siege is an http regression testing and benchmarking utility. It was designed to let web developers measure the performance of their code under duress, to see how it will stand up to load on the internet. Siege supports basic authentication, cookies, HTTP and HTTPS protocols. It allows the user hit a web server with a configurable number of concurrent simulated users. Those users place the webserver "under siege." **PLATFORM SUPPORT** Siege was written on GNU/Linux and has been successfully ported to AIX, BSD, HP-UX and Solaris. It should compile on most System V UNIX variants and on most newer BSD systems. Because Siege relies on POSIX.1b features not supported by Microsoft, it will not run on Windows. Of course you can use Siege to test a Windows HTTP server. **NEWS & RELEASE INFORMATION** SIEGE-2.61 November 19, 2004 19:15 EDT This is the current stable release. It backs off chunked transfer support and makes it a configurable option. If you may continue to use an older version and switch to HTTP/1.0 protocol if you are having problems with inconsistent data transfers. Developers, beta testers and adventurous souls, click **HERE**	OPEN SOURCE	Not Evaluated

		for the latest BETA version.		

VENDOR:		***CLAN PRODUCTIONS (Wales)***		
TOOL NAME:		**JBLITZ [Free and paid versions]**		
Web Page		**http://www.clanproductions.com/jblitz/index.html**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Affordable load testing tool from Clan Productions aimed at small web site developers. Each part of a site's functionality can be tested apart or together with up to 500 threads to simulate many users. Can request anything normally addressable through browser, including regular web pages, ASP scripts, JSP scripts, Servlets, CGI scripts etc. Web testing with JBlitz 4.2 Unleash the power of 1000s of virtual users... to seek out those elusive but embarrassing bugs and defects to improve performance and responsiveness to enhance the quality of your websites and web applications to help you gain satisfied customers to see your web presence blossom JBlitz 4.2 effectively tests dynamic websites, web services and web applications. JBlitz carries out stress, performance and functional testing by effectively simulating web traffic at various levels of loading. JBlitz detects, logs and acts on errors as they occur. JBlitz is a breeze to setup and run and is available for Windows, Solaris and Linux.	There is a free version and the paid version is from $99 for one license to $600 for small organization sight license - Unlimited Virtual Users	Not Evaluated

VENDOR:		*THE MEASUREMENT FACTORY, INC (U of CA)*		
TOOL NAME:		**Web Polygraph [Free]**		
Web Page		http://www.web-polygraph.org/		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Freely available benchmarking tool for caching proxies, origin server accelerators, L4/7 switches, and other Web intermediaries. Other features: for high-performance HTTP clients and servers, realistic traffic generation and content simulation, ready-to-use standard workloads, powerful domain-specific configuration language, and portable open-source implementation. C++ source available; binaries avail for Windows. Web Polygraph is a freely available benchmarking tool for caching proxies, origin server accelerators, L4/7 switches, content filters, and other Web intermediaries. Polygraph's features include: • high-performance HTTP clients and servers • realistic HTTP and SSL traffic generation • flexible content simulation • ready-to-use standard workloads • powerful domain-specific configuration language • portable open-source implementation Web Polygraph is capable of simulating numerous workloads. Here we just give a few useful examples, including standard workloads. Note that while overall description of a workload does not depend on Polygraph version, the corresponding PGL configuration may change as the benchmark evolves. You have a choice of using the version of Polygraph specified in the workload description or adjusting the given configuration for newer versions of Polygraph. Workload names usually end with a number (e.g., *PolyMix-3* or *WebAxe-4*). That number represents "generation" of the workloads. Usually, workloads from the same generation have similar base features and are supported by the	Free	Not Evaluated BENCH MARK TOOL User manual is some what cryptic for a normal user Created by the University of California

		same Polygraph version. If possible, use the latest generation of workloads.		

VENDOR:	***ACME LABS SOFTWARE***
TOOL NAME:	**Http_Load [Free]**
Web Page	**http://www.acme.com/software/http_load/**

Web Testing	Functional Testing	Feature pro/cons	Aprox Pricing	Our Evaluation / Rating
		Free load test application from ACME Labs to generate web server loads, from ACME Software. Handles HTTP and HTTPS; for Unix. http_load runs multiple http fetches in parallel, to test the throughput of a web server. However unlike most such test clients, it runs in a single process, so it doesn't bog down the client machine. It can be configured to do https fetches as well. You give it a file containing a list of URLs that may be fetched, a flag specifying how to start connections (either by rate or by number of simulated users), and a flag specifying when to quit (either after a given number of fetches or a given elapsed time). There are also optional flags for checksums, throttling, random jitter, and progress reports.	Free	Not Evaluated

VENDOR:		*TCNOW.COM*		
TOOL NAME:		**Open System Testing Architecture [Free – Open Source]**		
Web Page		http://www.opensta.org/		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		'Open System Testing Architecture' is a free, open source web load/stress testing application, licensed under the Gnu GPL. Utilizes a distributed software architecture based on CORBA. OpenSTA binaries available for Windows. **What is OpenSTA?** **Open, Systems Testing Architecture** OpenSTA is a distributed software testing architecture designed around CORBA, it was originally developed to be commercial software by CYRANO. The current toolset has the capability of performing scripted HTTP and HTTPS heavy load tests with performance measurements from Win32 platforms. However, the architectural design means it could be capable of much more. **Web Load Testing** **HTTP Stress & Performance Tests** The applications that make up the current OpenSTA toolset were designed to be used by performance testing consultants or other technically proficient individuals. This means testing is performed using the record and replay metaphor common in most other similar commercially available toolsets. Recordings are made in the tester's own browser producing simple scripts that can be edited and controlled with a special high level scripting language. These scripted sessions can then be played back to simulate many users by a high performance load generation engine. Using this methodology a user can generate realistic heavy loads simulating the activity of hundreds to thousands of virtual users. **Data Collection** **Timers, Windows Performance** **& SNMP Statistics** Results and statistics are collected during test runs by a variety of automatic and user controlled mechanisms. These can include scripted timers, SNMP data, Windows Performance Monitor stats and HTTP results & timings. Much of the data logged can be monitored live during the test runs; once test runs are complete, logs can be viewed, graphed, filtered and exported for use by more sophisticated report generation software.	Free Open Source	Not Evaluated

		Completely Free & **Open Source** The OpenSTA toolset is Open Source software licensed under the GNU GPL (General Public License), this means it is free and will always remain free. If you wish to build your own customized version of OpenSTA or take part in its development then the complete toolset source code, buildable in Microsoft Visual Studio 6, and all related information is available from OpenSTA.SourceForge.net, the developer home site. **Community Supported** **Development Driven by the Users** Much more information can be found out about OpenSTA by checking the online documentation or simply downloading and installing the toolset. The FAQ contains lots of other useful background information and helpful tips, this should be the first place you look if you need help with anything not covered in the documentation. There is no need to stop at reading the FAQ either, it is hosted on the OpenSTA Community Portal and, in common with every other resource on this site, it is user editable. This site is a great place for every OpenSTA user to share their experiences with the product and help others learn and use OpenSTA: Remember, the toolset is completely free and any time the developers spend helping users is time they are not enhancing, or fixing problems with, the toolset. By helping other users you are in fact helping OpenSTA and its community become stronger. The premier place for free OpenSTA support and discussions is the OpenSTA Users Mailing List, here the developers and many long time users of this toolset give as much help as their freetime will allow. hosting donated by tcNOW.com		

VENDOR:		***AVE WRIGLEY***		
TOOL NAME:		**ApacheBench [Shareware]**		
Web Page		**http://www.cpan.org/modules/by-module/HTTPD/**		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		Perl API for Apache benchmarking and regression testing. Intended as foundation for a complete benchmarking and regression testing suite for transaction-based mod_perl sites. For stress-testing server while verifying	Shareware	Not Evaluated Perl testing tool

		correct HTTP responses. Based on the Apache 1.3.12 ab code. Available via CPAN as .tar.gz file. This is ApacheBench version 0.51, the Perl API for Apache benchmarking and regression testing. This project is meant to be the foundation for a complete benchmarking and regression testing suite for an advanced, transaction-based mod_perl site. We need to be able to stress our server to its limit while also having a way to verify the HTTP responses for correctness. Since our site is transaction-based (as opposed to content-based), we needed to extend the single-URL ab model to a multiple-URL sequence model. ApacheBench is based on the Apache 1.3.12 ab code (src/support/ab.c). Please see the POD for complete details. Adi Fairbank adi@certsite.com September 26, 2000		

RESOURCE WEB PAGES TO FIND TOOLS....

VENDOR:		*SOURCEFORGE.net*		
TOOL NAME:		**SourceForge - Hosting open source application tools**		
Web Page		http://sourceforge.net/docman/display_doc.php?docid=6025&group_id=1		
Web Testing	**Functional Testing**	**Feature pro/cons**	**Aprox Pricing**	**Our Evaluation / Rating**
		What is SourceForge.net? SourceForge.net is the world's largest Open Source software development web site, providing free hosting to tens of thousands of projects. The mission of SourceForge.net is to enrich the Open Source community by providing a centralized place for Open Source developers to control and manage Open Source software development. To fulfill this mission goal, we offer a variety of services to projects we host, and to the Open Source community. **Open Source software** SourceForge.net provides free hosting to Open Source software development projects. The concept of 'Open Source' promotes the benefits of collaborative development by ensuring that potential end-users are able to obtain and use software, and that the software may be improved and expanded to meet the needs of its users. Collaboration within the Open Source community (developers and end-users) promotes a higher standard of quality, and helps to ensure the long-term viability of both data and applications. Additional information regarding the Open Source concept may be found on the Open Source Initiative (OSI) web site. **OSTG (Open Source Technology Group)** SourceForge.net is owned by OSTG, Inc. ("Open Source Technology Group"). OSTG, Inc (Open Source Technology Group) is the most dynamic community-driven media network on the Web. OSTG publishes two world-renowned networks of Web sites: the OSTG technology network, and the MediaBuilder network. OSTG delivers more than 160 million page views and reaches 9 million unique visitors per month. OSTG technical sites attract all levels of IT decision maker and technical buyer, from C-level to project managers. Technologists, enterprise architects, developers and system administrators all turn to OSTG to create, debate, and make or break IT news, and learn about the latest tools, technologies and techniques. OSTG sites include		HOSTING OF ALL FREEWARE OF SHAREWARE SOFTWARE OR AS THEY CALL IT OPEN SOURCE SOFTWARE

		Slashdot.org, the award-winning news discussion site; and SourceForge.net, the world's largest collaborative open source software development site. OSTG also owns ThinkGeek.com, the leading e-commerce site featuring innovative products "for smart masses". The MediaBuilder network reaches Web and print designers and consumers seeking original animations, presentation tools, design tools, online greetings, and other media enabling them to express themselves with powerful communications. Visitors download and share images from Animation Factory and other MediaBuilder sites to share via their cellular phones, PDAs and e-mail applications all over the world. OSTG is the No. 1 network for delivering people who look for technology news online, and the No. 1 network for delivering visitors who have shopped for or purchased software online in the past 6 months, based on composition*. OSTG.com / Devchannel / freshmeat / Geocrawler / Linux.com / NewsForge / Slashdot / SourceForge / MediaBuilder / ThinkGeek / Animation Factory OSTG, Inc. is a wholly owned subsidiary of VA Software Corporation. * Source: Nielsen/NetRatings' Winter 2003 @Plan **Additional information** The SourceForge.net site is powered by SourceForge collaborative software development tools from VA Software (OSTG's parent company). The SourceForge software platform has allowed SourceForge.net to provide a scalable solution for empowering development teams across the diverse worldwide Open Source community. Additional information regarding the SourceForge.net site and its services may be found throughout the SourceForge.net web site, and within the SourceForge.net Site Documentation collection. Further inquiries may be directed to the SourceForge.net staff team by submitting a support request. **SourceForge for Corporate Use** For corporate use, VA Software provides SourceForge Enterprise Edition, a secure, enterprise-grade solution for optimizing distributed enterprise development. SourceForge Enterprise Edition was architected and built expressly to accelerate distributed enterprise development via secure, robust operation behind a corporate firewall and by providing an integration-ready solution with the extra capabilities, security, reporting features and ease-of-use required by a broad range of enterprise users. Information is provided by VA Software about the SourceForge Enterprise Edition and how it differs from SourceForge.net.		

VENDOR:	***OPENSOURCETESTING.ORG***
TOOL NAME:	**Lists of tools available**
Web Page	**http://www.opensourcetesting.org/**

Web Testing	Functional Testing	Feature pro/cons	Aprox Pricing	Our Evaluation / Rating
				Not Evaluated

VENDOR:	***FINDING TOOLS ON THE INTERNET (contributed by William T. Prost)***
TOOL NAME:	**Where to look for tools available**
Web Page	**See below**

Web Testing	Functional Testing	Feature pro/cons	Aprox Pricing	Our Evaluation / Rating
		Cordell I know in the meeting yesterday you talked about how you were limited on what software you could use to load test b/c of the limited budget. I was wondering if you looked into any of the open-source software packages available to do this? I did a quick search and some promising ones are: http://sourceforge.net/projects/dieseltest/ http://sourceforge.net/projects/deluge/		Not Evaluated

		http://hammerhead.sourceforge.net/ And here's a whole site about different open source testing tools: http://opensourcetesting.org/performance.php And a good hosting site of open source software is: http://sourceforge.net/ I think people often dismiss open source software b/c they have a hard time believing they can get similar software for free (or almost free). There is actually a decent amount of good open source software out there, especially when it comes to programming and testing. *William T. Prost* Systems Analyst Washington School Information Processing Cooperative 425.349.6663 wprost@wsipc.org		

OUR CONTACT INFORMATION:

If you have tested any of these tools yourself, or if you know of other similar tools, especially if you have tested them, please email your evaluation and the vendor information along with your contact information to Cordell Vail at this email address:

MakingSuccessWork@yahoo.com

and we will add your evaluation to this table and post it to our web page at:

www.vcaa.com

If you do not want to have credit for the evaluation, please state that in your email and we will just add the information.

More good things to come

Cordell Vail and Joe Towns