

A-B-C-A-B (1)

A-B-C-F (2.29)

A-B-D (5.71),

which has a sum of occurrences of 9. Because 9 is less than 10 (the number of Vusers), the process exits the program loop at step 352.

With reference to block 364, the occurrence values are then adjusted, if necessary, to produce a set of integer values for which the sum is equal to the number of Vusers. Any scaling/truncating technique that maintains the general proportions between the occurrence values may be used for this purpose. This step may, for example, produce the following:

A-B-C-A-B (1)

A-B-C-F (3)

A-B-D (6).

With reference to blocks 366 and 368, each route is then outputted as a respective Web script file. In addition, the process generates a scenario file that specifies the allocations of Vusers to Web scripts. In this example, the scenario would include a single Vuser that runs the A-B-C-A-B script, a group (Sgroup) of three Vusers that run the A-B-C-F script, and a group of six Vusers that run the A-B-D script. The scenario file would also include the loop information specified by the user.

(d) Source Code Listing

Included as Appendix C is a source code listing which includes an implementation of the two-phase process. The listing also includes code for implementing the Vusers.

XII. Conclusion

While certain preferred embodiments of the invention have been described, these embodiments have been presented by way of example only, and are not intended to limit the scope of the present invention. For example, although the present invention has been described with reference to the standard protocols, services and components of the World Wide Web, it should be recognized that the invention is not so limited, and that the various aspects of the invention can be readily applied to other types of web sites and server applications (including, for example, mainframe terminal applications). Accordingly, the breadth and scope of the present invention should be defined only in accordance with the following claims and their equivalents.

In the claims which follow, reference characters used to designate claim steps are provided for convenience of description only, and are not intended to imply any particular order for performing the steps.

What is claimed is:

1. A method of load testing of a web site, the method comprising the computer-implemented steps of:

processing site access information stored within an access log to generate at least one test script, the access log generated by a server application that runs on a computer system of the web site, the server application configured to serve informational content over a network in response to requests by visitors of the site and to record visitor accesses to the site within the access log, the site access information representing accesses to the site by multiple different visitors during ordinary, post-deployment usage of the web site, the test script including addresses of informational content entities of the site; and

running the at least one test script to exercise the site, the step of running comprising submitting informational requests to the server application.

2. The method of claim 1, wherein the step of processing comprises preserving a general load distribution represented by the access log, so that the step of running produces a load that reflects actual usage patterns of visitors.

3. The method of claim 1, wherein the step of processing is performed such that the step of running produces a load on the site that has generally the same distribution as a load represented within the access log.

4. The method of claim 3, further comprising prompting a user to enter a control parameter, wherein the load has a magnitude which is specified by the control parameter.

5. The method of claim 1, wherein the step of processing comprises identifying a plurality of routes taken by visitors to the site.

6. The method of claim 5, wherein the step of identifying a plurality of routes comprises using access timestamps and visitor identifiers contained within the access log to determine navigational links taken by individual visitors.

7. The method of claim 5, wherein the step of processing further comprises translating the plurality of routes into a plurality of test scripts.

8. The method of claim 7, wherein the step of running comprises running the plurality of test scripts in parallel to emulate multiple concurrent visitors.

9. The method of claim 1, further comprising displaying the site access information on a display screen in conjunction with a graphical map of the web site.

10. The method of claim 1, wherein the step of running comprises submitting the informational requests to the server application using an implementation of the HTTP protocol.

11. The method of claim 1, wherein the access log is stored in a standard file format for server access log files.

12. The method of claim 1, wherein the web site is implemented as an intranet.

13. A method of testing of a server application that runs on a computer system of a computer network, the method comprising the computer-implemented steps of:

processing client request information stored within an access log to generate at least one data structure, the access log generated by the server application, the server application configured to serve informational content over the computer network in response to requests from client applications, and configured to record requests from the client applications within the access log, the client request information representing requests from multiple users during ordinary, post-deployment usage of the server application within a multi-user network, the data structure including information for performing a load test of the server application; and

load-testing the server application, the step of load-testing comprising submitting client requests to the server application from a testing application based on the information stored within the at least one data structure.

14. The method of claim 13, wherein the at least one data structure comprises a sequence of addresses of content entities that are served by the server application, and the step of load-testing comprises using the addresses to request the content entities from the server application.

15. The method of claim 14, wherein the step of processing comprises preserving a general load distribution represented by the access log, so that the client requests for the content entities are generally distributed among the content entities as in the access log.

16. The method of claim 13, wherein the step of processing is performed such that the step of load-testing produces