

Success story with OpenSTA

Author: jcrvs@hotmail.com

Table of Contents

1. Problem statement..... 3
1.2 The Solution 3
1.3 Planning..... 4
 1.3.1 Approach 4
 1.3.2 The Teams 4
 1.3.3 OpenSTA tool 4
 1.3.4 OpenSTA scripting using Script Modeler..... 4
 1.3.5 OpenSTA Test configuration using Commander..... 5
1.4 The Broadvision Architecture and test bed diagrams 6
 1.4.1 The Broadvision architecture and Test bed details for small Architecture 6
 1.4.2 The Broadvision architecture and Test bed details for medium Architecture..... 7
1.5 Test Configuration and Execution..... 8
 1.5.1 Machines and network in test bed 8
 1.5.2 Test execution 8
 1.5.3 Problems encountered with OpenSTA..... 8
1.6 The Results..... 9
 1.6.1 Number of Errors 9
 1.6.1.1 For Medium test 9
 1.6.1.2 For Small test 9
 1.6.2 Response time 10
 1.6.2.1 Page/Transaction: Login 10
 1.6.2.2 Page/Transaction: Update_profile transaction 12
1.7 Conclusion..... 14
1.8 ROI..... 14

1. Problem statement

The client requests a Broadvision architecture (small and medium) verification tests to be performed for the sample applications.

The consulting company accepts it since it has: -

1. The hardware resources,
2. The capability to configure the broadvision for various architecture as medium and small, and
3. The finest and the costliest load test tool,
4. Execute load test.

The catch is: Client is not bear the cost for the load test tool licenses since the load test tool licensing policy is stringent.

1.2 The Solution

The company neither wants to loose the client nor the opportunity to prove the client that they are experts in broadvision application development. The manager asks the test lead to evaluate cheaper tools.

None of the cheaper tools supported the cause, as they would lead the consulting company bear some costs.

Finally, the company has one hope Open source tool, OpenSTA.

1.3 Planning

The client had all the application deployed on broadvision server and the server was high on CPU and memory. Client wanted to scale the architecture but wanted some load statistics for the different architectures as small and medium.

1.3.1 Approach

A requirements analysis was conducted with client for their expectation and then outcome was as follows: -

1. Perform a load test of 1000 users for the small and medium architecture,
2. The example application that comes with the broadvision to be used for the load test,
3. Measure the response time on both architecture,
4. Report errors, if any and do not repeat the tests since time is short

1.3.2 The Teams

Two teams were formed for the same. The development team was assigned the task of configuration of the small and medium architecture for Broadvision server on Solaris machine. The test team was assigned the task for leaning the OpenSTA tool, creating scripts and scenarios for load test, and setting the load test bed.

1.3.3 OpenSTA tool

The OpenSTA, a tool that implements a distributed software testing architecture based on corba, designed to be used by performance testers. OpenSTA is Open source software licensed under the GNU General Public License.

The source code and executable is available at <http://www.opensta.org>.

1.3.4 OpenSTA scripting using Script Modeler

Since the team was familiar with the load test tool scripting, catching with OpenSTA was not at all the problem. Scripting directives were created and followed.

Refer the URL

1. <http://portal.opensta.org/modules.php?op=modload&name=News&file=article&sid=17&mode=thread&order=0&thold=0>
2. <http://www.qaforums.com/Forum36/HTML/000116.html>

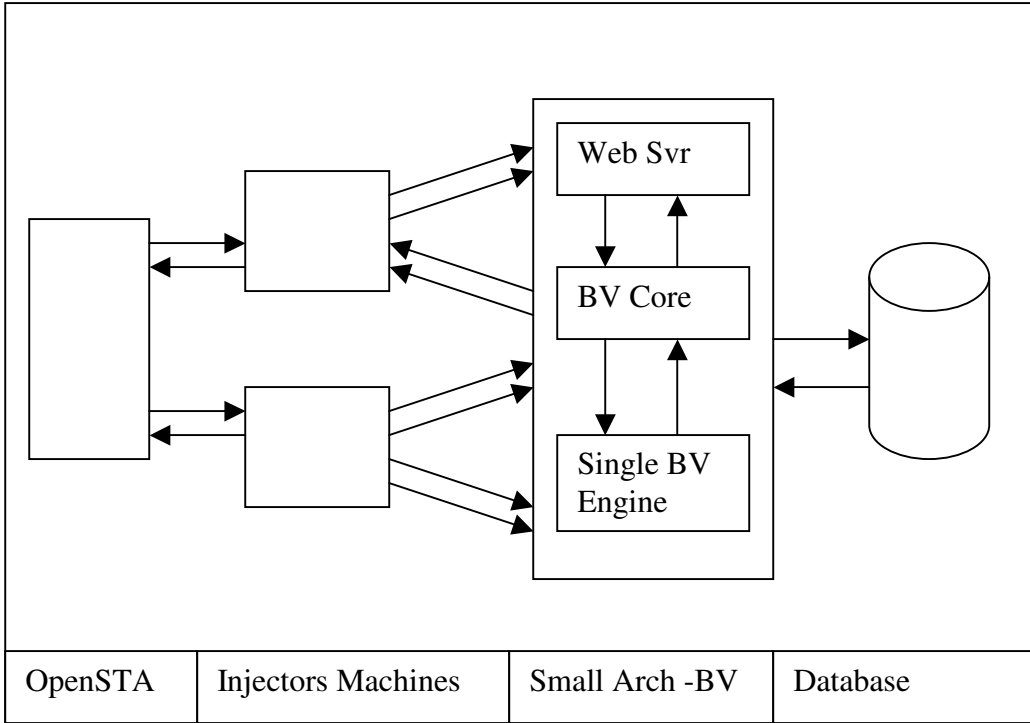
1.3.5 OpenSTA Test configuration using Commander

The test configuration was configured as follows using the commander: -

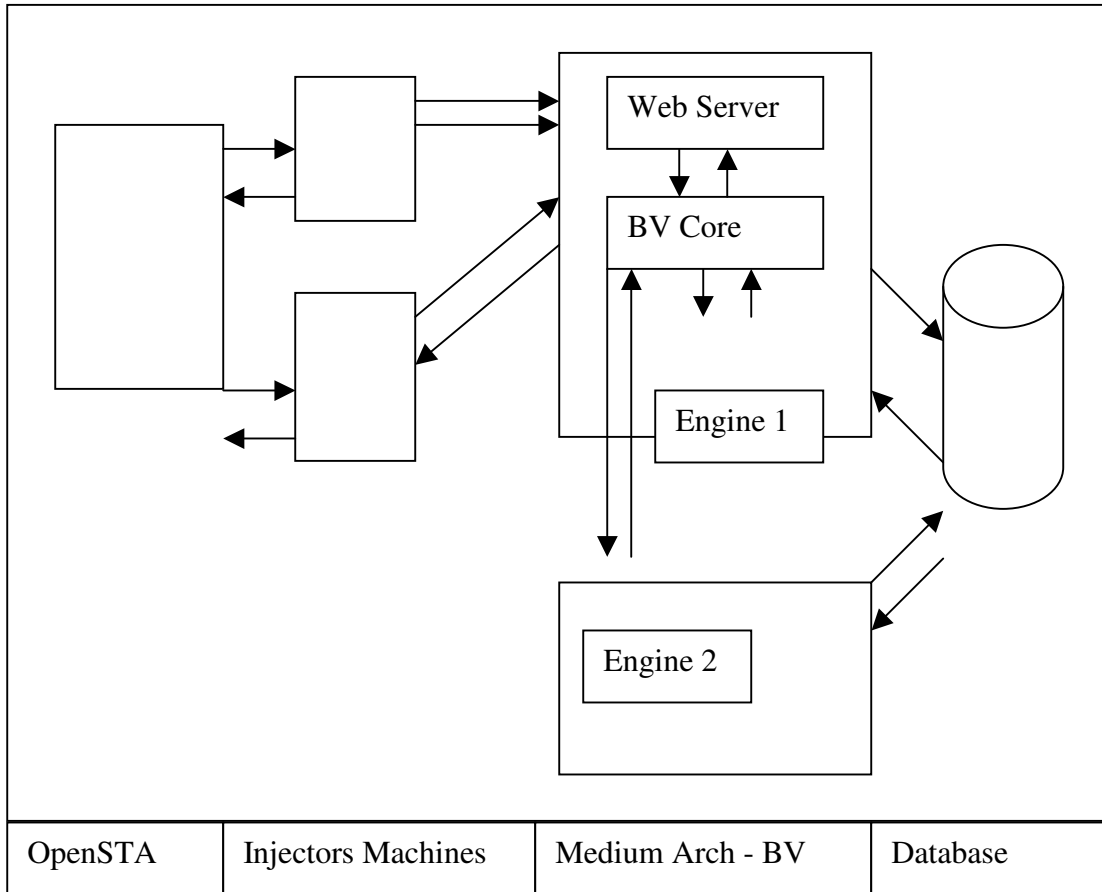
1. 500 users each to be run from two individual Injector machines
2. The total vusers to collect timers and http requests are 10% of total users
3. Each user iterating forty times and task termination status set to 'On completion'
4. A fixed delay add as breathing time for server in-between two iteration
5. A delay of five hours for 'Start Task group' so the test is run overnight when the network traffic is low

1.4 The Broadvision Architecture and test bed diagrams

1.4.1 The Broadvision architecture and Test bed details for small Architecture



1.4.2 The Broadvision architecture and Test bed details for medium Architecture



1.5 Test Configuration and Execution

1.5.1 Machines and network in test bed

The Commander was running on machine having P2, Windows NT Server 4.0, 256 MB RAM.

The Injectors machines: one Windows XP machines with 128 MB RAM, P2 and the other was WindoesNT4.0 workstation, P2 with 128 MB RAM.

The Network was 100 Mbps corporate network with switch speed of 100 Mbps and each machine had '3COM Ethernet link 10/100 PCI' NIC.

In our case since the user load is 100 vusers, these values were 100.

1.5.2 Test execution

The tests were configured with OpenSTA with Commander running on high memory machine and about 500-user load was configured to be run from the two injector machines.

1.5.3 Problems encountered with OpenSTA

The only problem encountered during OpenSTA was that the scripts failed to execute on the Injectors machines. Reverse setting of test (between Commander machine and the Injectors machines) was done and executed. (It was probably due to stale or loose connection between machines and needed as a sort of refresh.) This solves the problem.

Tests were executed on both the small and medium BV architecture.

1.6 The Results

The result analysis was performed on two parameters the number of errors and the response time for the various transactions.

1.6.1 Number of Errors

The number of errors were documented for the both the tested and are summarized as follows:

1.6.1.1 For Medium test

Number of Errors: 146

Types of Errors

- ERROR 10060: (A connection attempt failed because the connected party did not properly respond after a period of time, or established connection failed because connected host has failed to respond.) for VU 2-233 while establishing connection 232/1
- ERROR 10060 while re-connecting 361/10 to 10.236.133.27 on port 9065.
- ERROR 10038: (An operation was attempted on something that is not a socket.) while initiating a READ operation after sending a request (cannot send the request again). Connection ID is 98/4294967295.

The above errors were attributed to overloading of the server due to large user load and hence “server timeouts observed”.

1.6.1.2 For Small test

No of Errors: 1374

Types of Errors

- ERROR 10061: (No connection could be made because the target machine actively refused it.) for VU 2-126 while establishing connection 125/1.
- ERROR 10061: (Error message not found) for VU 1-10 while establishing connection 9/1.
- ERROR 10061: (Error message not found) for VU 1-46 while establishing connection 45/1.

The above errors typically denote overloading of the server due to “Server refusing the incoming connection”.

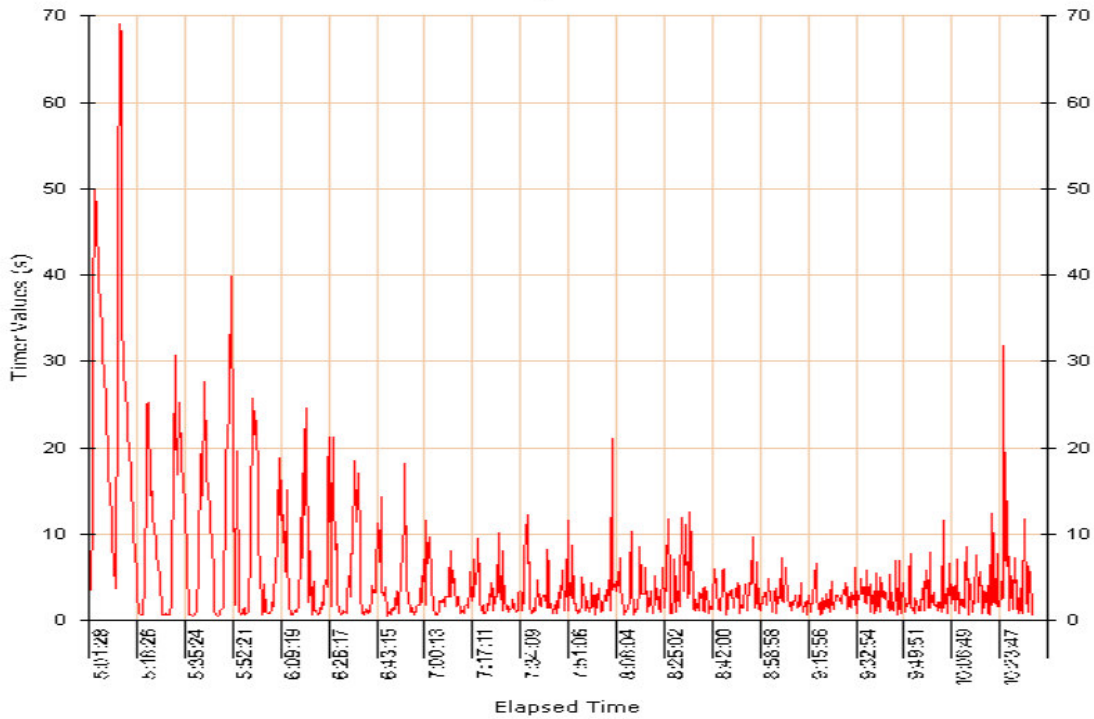
Note: the large number of errors for the small architecture test and the server is actually refusing connection where in case of medium architecture number of errors has significantly reduced.

1.6.2 Response time

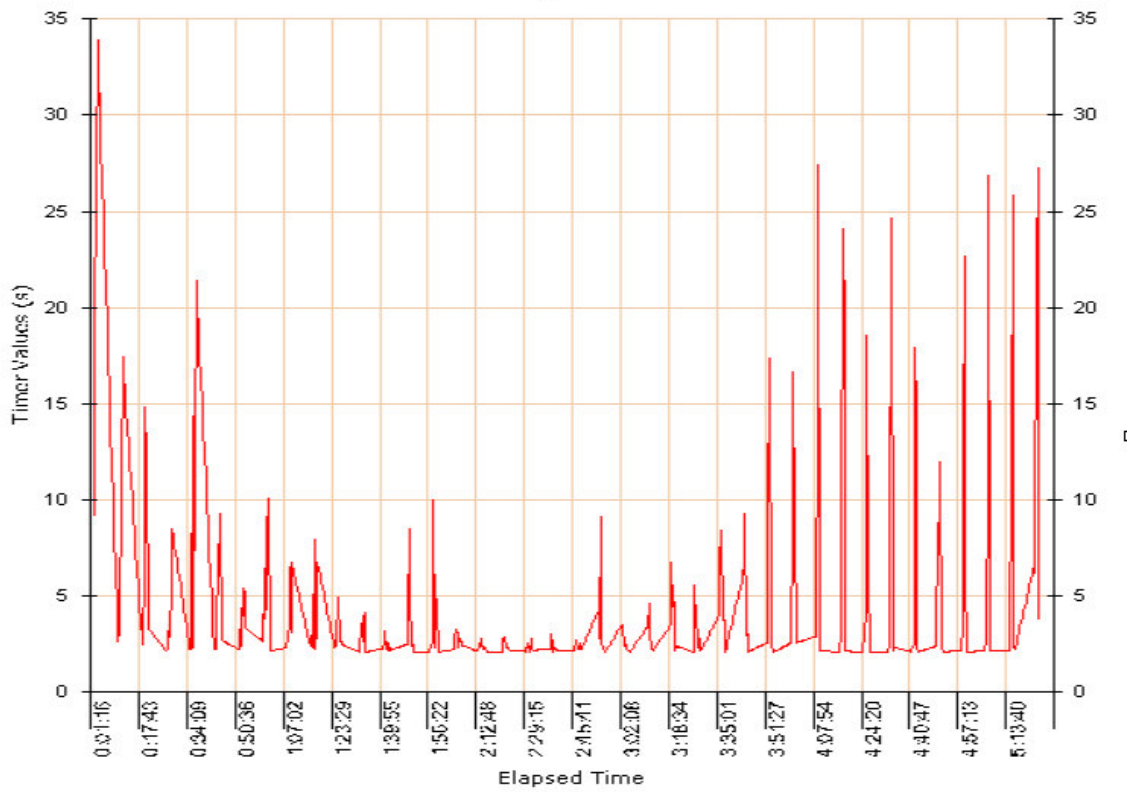
The timer graphs were generated from OpenSTA for various HTML pages of the Broadway application. Few of the graphs for the transactions are described below.

1.6.2.1 Page/Transaction: Login

For Small Architecture

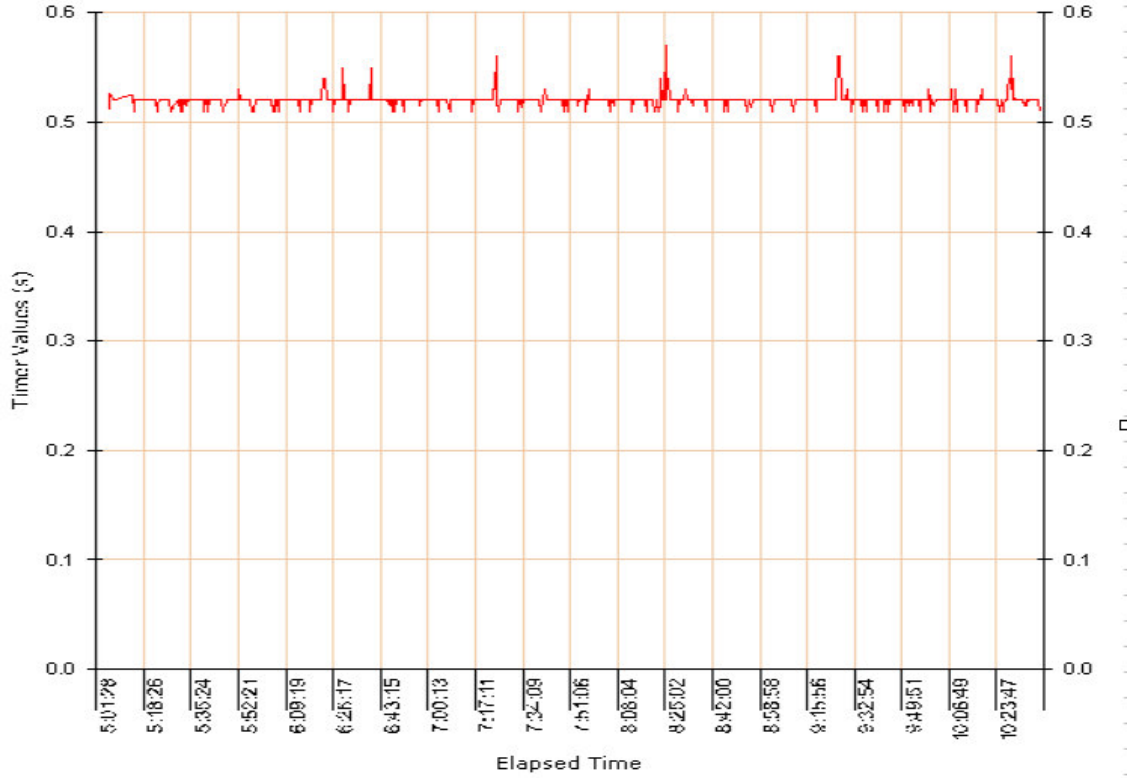


For Medium Architecture

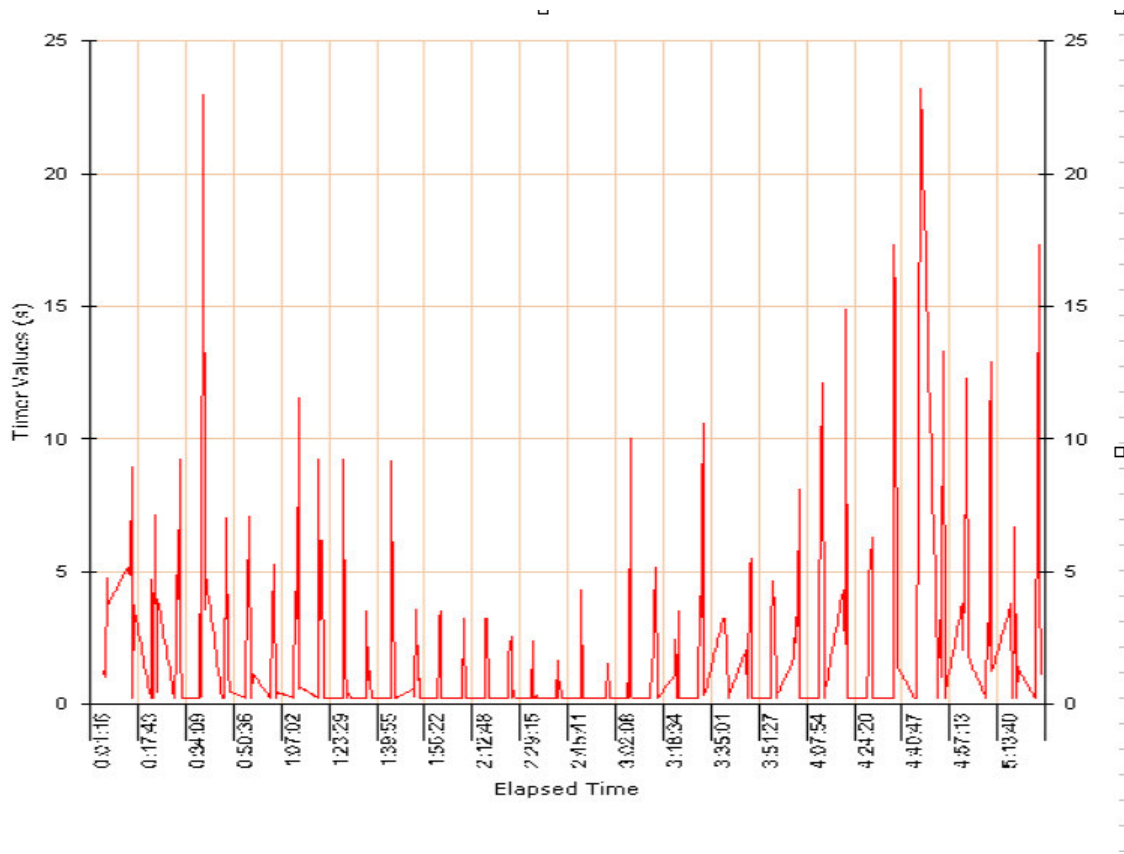


1.6.2.2 Page/Transaction: Update_profile transaction

For Small Architecture



For Medium Architecture



1.7 Conclusion

Conclusion 1 based on number of errors: - Notice in case of small architecture the number of 'server connection refused' errors are high (about 1374) but in case of medium architecture there is no server connection refused trace observed instead the error denote timeouts (around 146). This indicates the scaling of architecture is successfully (to some extent as the next step would be reduce the timeouts and large response time).

Conclusion 2 based on response time: - Note the increase in the response time for both transactions in the medium architecture. This increase is due to the fact that the medium architecture is able to handle more number of users than the small architecture and probably the network delay between the two machines running engine 1 and 2 (refer the section 1.4.2). This allows us to conclude that the medium architecture is a good design to scale the BV applications and sets new goals for the company to evaluate the scaling of the broadvision application server using the large architecture and eliminate all errors/problems observed for the medium architecture.

1.8 ROI

By using OpenSTA open source tool, the company saved around \$2000 that it would have spent on the costly load tool. OpenSTA is a hero!