HP OSMS: Tomcat Sizing Guide for HP ProLiant c-Class Blade Servers

HP Part Number: 5992-4859
Published: May 2008
Edition: 1.0

Legal Notice

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Acknowledgments

Intel and Itanium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

RED HAT READY Logo and RED HAT CERTIFIED PARTNER Logo are trademarks of Red Hat, Inc.

Table of Contents

List of Figures

List of Tables

Introduction

HP Open Source Middleware Stacks (OSMS) offer building block applications, such as the Web Server; technical blueprints, and documents such as this Tomcat Sizing Guide which describes the maximum workload that HP ProLiant c-Class Blade Servers are capable of supporting; and consulting services to speed the successful implementation of an open source strategy.

Executive Summary

The HP Open Source Integrated Portfolio (HP OSIP) comprises a range of products and services designed to verify that customers can successfully realize the cost and feature benefits of adopting open source software in their IT environments. HP Open Source foundation components include the base components of an open source-based ecosystem. HP servers and storage are validated to run the Linux operating system together with the supported commercial Linux distribution.

The results published in this document were achieved using the Apache benchmarking tool and Apche JMeter for each of the HP ProLiant c-Class Blade Servers. Various test configurations and test environments were used and documented to provide readers an opportunity to understand the testing methodology.

Intended Audience

The intended audience for this document is anyone who is interested in determining the Tomcat user connection workload that can be supported on a given HP ProLiant c-Class Blade Server and details for deploying a hardware environment based on the Tomcat environment requirements of certain applications.

Scope and Purpose

This document presents the results of a series of benchmark tests performed using the Apache Benchmarking tool and Apache JMeter. The tests were conducted running a Tomcat application server to evaluate system performance when running a web application under a heavy load. The following HP ProLiant servers were used in this test: BL460c, BL480c, and BL465c. The benchmark data provided in this sizing guide can assist customers in determining which HP Proliant c-Class Blade Server meets their Tomcat server requirements along with how-to information for configuring the HP Proliant c-Class Blade Server and storage in a specific production environment.

HP provides quality assurance from extensive integration testing with open source software and HP hardware so that you can confidently deploy the complete stack. Once you have completed a successful evaluation, you have the flexibility to “do it yourself” or get assistance from HP to incorporate open source stacks into your existing IT infrastructure.

HP Services

HP Open Source Consulting Services can help you build and integrate open source and commercial software across multiple operating system (OS) environments. Additionally, HP Open Source Support Services provide industry leading technical support for all the products HP sells, including hardware, operating systems, and open source middleware.

To learn more about HP Open Source Consulting and Support Services, contact your local HP sales representative or visit the HP Business and IT Services website at:

http://www.hp.com/hps

For the location of the nearest sales office, call:

- In the United States: +1 800 637 7740
- In Canada: +1 905 206 4725
- In Japan: +81 3 3331 6111
- In Latin America: +1 305 267 4220

- In Australia/New Zealand: +61 3 9272 2895
- In Asia Pacific: +8522 599 7777
- In Europe/Africa/Middle East: +41 22 780 81 11

Typographic Conventions

This document uses the following typographic conventions.

`Command` A command name or qualified command phrase.

`ComputerOut` Text displayed by the computer.

Ctrl-x A key sequence. A sequence such as **Ctrl-x** indicates that you must hold down the key labeled **Ctrl** while you press another key or button.

`ENVIRONVAR` The name of an environment variable, for example, `PATH`.

[ERRORNAME] The name of an error, usually returned in the `errno` variable.

Key The name of a keyboard key. **Return** and **Enter** both refer to the same key.

Term The defined use of an important word or phrase.

`UserInput` Commands and other text that you type.

`VARIABLE` The name of a placeholder in a command, function, or other syntax display that you replace with an actual value.

`\` *`(continuation character)`* A backslash (`\`) at the end of a line of code (such as a command) indicates that the following line of code is contiguous, and you must not insert a line break. This convention facilitates the typesetting of long lines of code examples on a printed page. If you cut and paste sample code from this publication, ensure that you remove backslash characters at line endings.

... The preceding element can be repeated an arbitrary number of times.

| Separates items in a list of choices.

Publishing History

The document publishing date and part number indicate the current edition of the document. The publishing date changes when a new edition is printed. Minor changes might be made without changing the publishing date. The document part number changes only when extensive changes are made. Document updates might be issued between editions to correct errors or document product changes. For the latest version of this document online, see the HP Technical Documentation website at:

http://www.docs.hp.com

Manufacturing Part Number	Supported OS	Edition Number	Publication Date
5992-4859	Red Hat Enterprise Linux 5 AS Update 1 (RHEL5ASu1)	1	May 2008

HP Encourages Your Comments

HP encourages your comments concerning this document. We are committed to providing documentation that meets your needs. Send any errors found, suggestions for improvement, or compliments to:

`feedback@fc.hp.com`

Include the document title, manufacturing part number, and any comment, error found, or suggestion for improvement you have concerning this document.

Hardware Test Environment

This benchmark was performed within an HP BladeSystem c-Class blade enclosure with enhanced backplane using C-GbE2 Network interconnect switches. Table 1 lists the specifications of the hardware test environment.

Table 1 Test Hardware Environment

Hardware	BL460c	BL465c	BL480c
CPU	(2) Quad-Core Intel® Xeon® processor X5365 (3.0 GHz/4MB L2 cache per core)	(2) Dual-Core AMD Opteron™ processor 2218 (2.6 GHz/1MB L2 cache per core)	(2) Quad-Core Intel® Xeon® processor X5365 (3.0 GHz/4MB L2 cache per core)
Memory	16GB (8x2GB PC2-5300 DDR2)	14GB (6x2GB+ 2x1GB PC2-5300 DDR2)	24GB (12x2GB PC2-5300 DDR2)
Disk Storage	2 x 73GB SAS Smart Array E200i controller with Battery Backed Write Cache (BBWC)	2 x 73GB SAS Smart Array E200i controller with Battery Backed Write Cache (BBWC)	4 x 73GB SAS Smart Array P400i controller with Battery Backed Write Cache (BBWC)
File System Partitioning	(ext3 – no LVM): Disk0: /, /boot, swap	(ext3 – no LVM): Disk0: /, /boot, swap	(ext3 – no LVM): Disk0: /, /boot, swap
Operating System	RHEL5 AS u1 SMP kernel (2.6.18-53.el5) with errata updates	RHEL5 AS u1 SMP kernel (2.6.18-53.el5) with errata updates	RHEL5 AS u1 SMP kernel (2.6.18-53.el5) with errata updates

During the test, the server and configuration that were used as the client where ab and JMeter were run is as follows:

```
Machine Model: HP ProLiant BL460c
CPU: (1)Dual-Core Intel® Xeon® processor X5160
(3.0GHz/4MB L2 cache per core)
Memory: 2GB
OS: RHEL5 AS u1
```

Tomcat Installation

The Tomcat software is available on the Apache Software Foundation website. The file to download for Linux servers is `apache-tomcat-6.0.14.tar.gz` file. The following provides the steps for installing and configuring Tomcat.

NOTE: Prior to installing Apache Tomcat, please verify that JDK is installed and configured on the system.

1. Unzip the `*.gz` file to the dictionary where you want to install tomcat.

   In the example that follows, the directory used is `/usr/local/apache-tomcat`.

2. To start Tomcat, enter the command:

   ```
   # /usr/local/apache-tomcat/bin/startup.sh
   ```

   The following messages display:

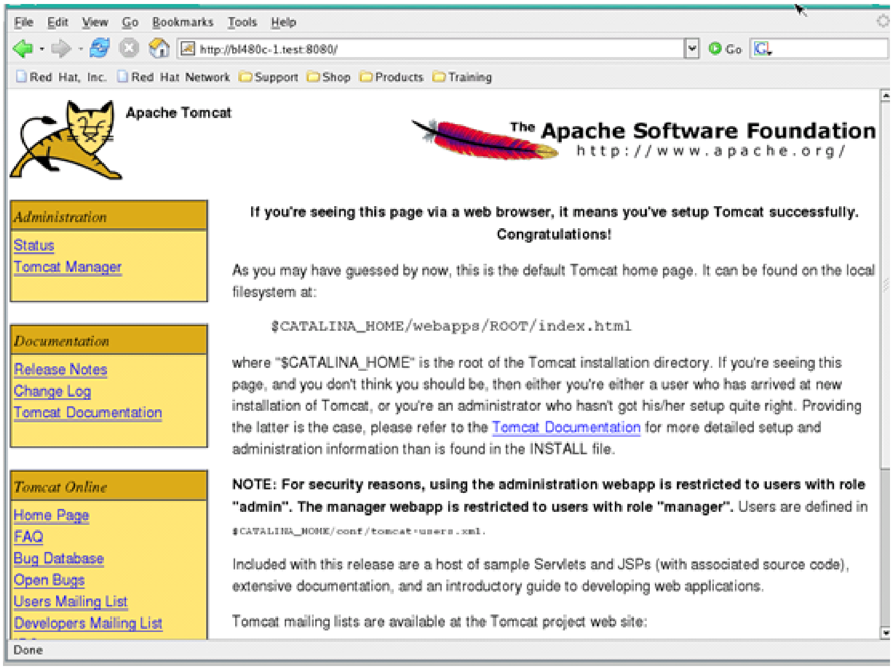
   ```
   Using CATALINA_BASE: /usr/local/apache-tomcat
   Using CATALINA_HOME: /usr/local/apache-tomcat
   Using CATALINA_TMPDIR: /usr/local/apache-tomcat/temp
   Using JRE_HOME: /usr/local/jdk
   ```

3. Verify the Tomcat installation by launching a browser and navigating to the following URL:

   `http://<YOUR_TOMCAT_SERVER_IP>:8080/`

   The Tomcat web page displays as shown in Figure 1.

Figure 1 Tomcat Successful Installation Verification



If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!

As you may have guessed by now, this is the default Tomcat home page. It can be found on the local filesystem at:

```
$CATALINA_HOME/webapps/ROOT/index.html
```

where "$CATALINA_HOME" is the root of the Tomcat installation directory. If you're seeing this page, and you don't think you should be, then either you're either a user who has arrived at new installation of Tomcat, or you're an administrator who hasn't got his/her setup quite right. Providing the latter is the case, please refer to the Tomcat Documentation for more detailed setup and administration information than is found in the INSTALL file.

NOTE: For security reasons, using the administration webapp is restricted to users with role "admin". The manager webapp is restricted to users with role "manager". Users are defined in `$CATALINA_HOME/conf/tomcat-users.xml`.

Included with this release are a host of sample Servlets and JSPs (with associated source code), extensive documentation, and an introductory guide to developing web applications.

Tomcat mailing lists are available at the Tomcat project web site:

NOTE: If the iptables firewall is enabled in the system, the Tomcat server can not be accessed from other machines.

4. If necessary, stop Tomcat by entering the following command:

   ```
   # /usr/local/apache-tomcat/bin/shutdown.sh
   ```

   The following messages display:

   ```
   Using CATALINA_BASE: /usr/local/apache-tomcat
   Using CATALINA_HOME: /usr/local/apache-tomcat
   Using CATALINA_TMPDIR: /usr/local/apache-tomcat/temp
   Using JRE_HOME: /usr/local/jdk
   ```

Tomcat Configuration

Setting JVM Options for Tomcat

Edit the `/usr/local/apache-tomcat/bin/catalina.sh` file and add the JVM options to the `JAVA_OPTS` environment variable. Table 2 lists the JVM options that need to be added to the `catalina.sh` configuration file.

Table 2 JVM Configuration

JVM Option	Value	Description
-Xmx	4g	The maximum Java heap size.
-Xms	4g	The initial Java heap size.
-Xmn	1g	The size of young generation.
-XX:+UseParallelGC	–	To use parallel garbage collection for scavenges.
-XX:PermSize	50m	The initial size of permanent generation.

Setting Tomcat Connector Attributes

Edit the /usr/local/apache-tomcat/conf/server.xml file and add the attributes listed in Table 3 to the Connector element. The following example shows the element in the xml file after the connector attributes are added.

```
<Connector port="8080" protocol="HTTP/1.1"
maxThreads="3000" acceptCount="200" ...
```

Table 3 Tomcat Configuration

Attribute	Value	Description
maxThreads	3000	The maximum number of request processing threads to be created by this connector, which therefore determines the maximum number of simultaneous requests that can be handled. If not specified, the default value for this attribute is 40. If an executor is associated with this connector, this attribute is ignored and the connector executes tasks using the executor rather than an internal thread pool.
acceptCount	2000	The maximum queue length for incoming connection requests when all possible request processing threads are in use. Any requests received when the queue is full are refused. The default value is 10.
minSpareThreads	500	The number of request processing threads that are created when this connector is first started. The connector also verifies that it has the specified number of idle processing threads available. This attribute should be set to a value smaller than that set for maxThreads. The default value is 4.
maxSpareThreads	2000	The maximum number of unused request processing threads that are allowed to exist until the thread pool starts stopping the unnecessary threads. The default value is 50.
enableLookups	false	Set to “true” if you want calls to *request.getRemoteHost()* to perform DNS lookups in order to return the actual host name of the remote client. Set to “false” to skip the DNS lookup and return the IP address in String form instead (thereby improving performance). By default, DNS lookups are enabled.

Static Web Page Benchmark

This section provides information on running static web page benchmark on BL460c, BL480c and BL465c servers. The Apache Benchmarking tool was used on web pages of different sizes to generate two types of data: 1) the number of requests (recorded per second) and 2) the response time (recorded in milliseconds). A basic analysis of these results is also included with this data to assist you in understanding the benchmark results.

Benchmark Software

The Apache HTTP Benchmarking Tool is a command line (ab) benchmarking tool that is bundled with the standard Apache package to benchmark Hypertext Transfer Protocol (HTTP) Web servers. It is a free and open source software that is distributed under the Apache license. It can simulate large numbers of connections to perform a series of requests against given pages in applications on the HTTP web server. The tool then provides performance statistics, in particular data about the number of requests per second which the server is capable of serving and the average response time for all requests.

Table 4 Main ab Options

Main ab Option	Value	Description
-c	Varies from 100 to 2500.	Number of multiple requests to perform at one time, such as concurrent connections.
-n	100 times the number of concurrent connections.	Number of requests to perform for the benchmarking session

For more information about ab, see the following web site:

http://httpd.apache.org/docs/2.0/programs/ab.html

Benchmark Results

In the static web page testing, several web pages were used with different sizes that varied from 5KB to 200KB. For each web page, ab was run with a varying number of concurrent connections and total requests to determine the maximum requests per second and the maximum response time for 80% of the fastest requests. For this test, the concurrent connections varied in number from 100 to 2500 and the total number of requests was set to 100 times the number of concurrent connections.

The benchmark results for the testing are presented in the following sections and show the maximum number of requests per second and maximum response time for 80% of the fastest requests that each system can handle, depending on the number of concurrent connections and the size of pages.

The following tables show that, with regard to requests per second, having more connections can negatively impact the value for small size pages like the 5KB page but have little influence for bigger size pages. In regard to response time, when the number of concurrent connections is larger, so is the response time. For relatively small pages such as a 5KB size page, both the HP ProLiant BL460c and BL480c servers, configured with 2 Quad-Core processors, can handle more than 10,000 requests per second even with 2500 concurrent connections. However, the HP ProLiant BL465c server, configured with 2 Dual-Core Opteron processors, can process fewer requests. For pages over 30KB, the number of requests that all three servers can handle per second is stable and does not vary as user connections are increased.

HP Proliant BL460c Server

The HP ProLiant BL460c server has features that are equal to standard 1U rack-mount servers. The two-processor, dual-core or quad-core ProLiant BL460c combines power-efficient compute power, and high density with expanded memory and I/O for maximum performance. The ProLiant c7000 enclosure supports up to 16 BL460c server blades, two more servers than the IBM BladeCenter, and each BL460c supports double the memory capacity of the HS21 server without an expansion blade. The Proliant BL460c now comes with dual-core and quad-core Intel® Xeon® processors, DDR2 fully buffered DIMMs, Serial Attached SCSI (SAS) or Serial ATA (SATA), hard drives, and support of multi-function NICs and multiple I/O cards. The BL460c provides a performance system ideal for the full range of scale-out applications. In this small form factor, the BL460c includes more features to enable high-availability, such as hot plug hard drives, mirrored memory, online spare memory, memory interleaving, embedded RAID capability, and enhanced remote Lights-Out management. Table 5 provides the benchmark results that were obtained using a ProLiant BL460c server configured with two Quad-Core Intel® Xeon® processors running at 3.0GHz.

Table 5 Static Web Page Test Results for the HP ProLiant BL460c Server

User Connections	5K		30K		80K		200K	
	req/sec	resp time	req/sec	resp time	req/sec	resp time	req/sec	resp time
100	18886.63	6	3769.51	26	1426.88	76	572.57	189
200	17643.76	9	3770.53	51	1427.35	115	572.96	328
500	17230.17	10	3773.20	64	1425.38	467	572.50	1448
1000	15180.41	12	3766.08	321	1424.56	1033	571.31	3008
1500	14174.95	69	3772.20	337	1423.59	1622	570.14	4139
2000	12617.91	123	3731.98	545	1421.59	2139	569.16	5217

Figure 2 Static Web Page Test Results for the HP ProLiant BL460c Server - Requests per Second

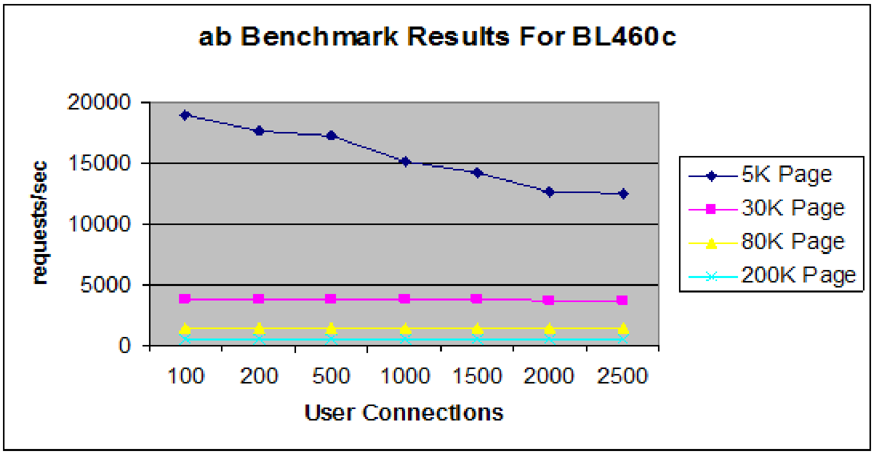
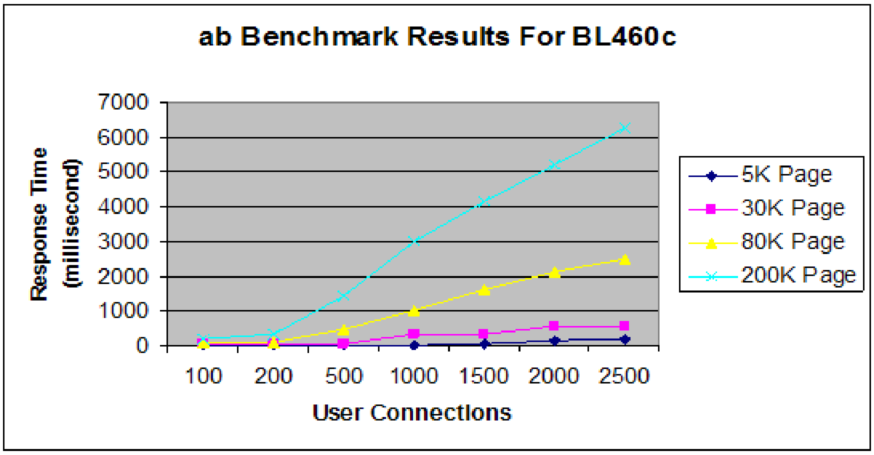


Figure 3 Static Web Page Test Results for the HP ProLiant BL460c Server - Average Response Time



HP Proliant BL465c Server

The HP ProLiant BL465c server is a two-way server blade designed for maximum compute density and power efficiencies. The small form factor accommodates two blades in each slot allowing for 16 servers in a standard 6u enclosure. Each CPU is a 95 Watt version of the Dual-Core AMD Opteron 2000 series processor, which gives the equivalent of four processing units per server. Up to 32GB of PC2-5300 DIMMs running at 667MHz is supported. Internal storage consists of two small form factor disk drives, either SAS or SATA attached to the integrated Smart Array E200i RAID controller with 64MB cache supporting RAID 0, 1. An optional StorageWorks SB40c storage blade provides connectivity to additional storage, supporting RAID 5, and ADG. The benchmark results in Table 6 were obtained with a ProLiant BL465c server configured with two Dual-Core Opteron processors running at 2.6 GHz.

Table 6 Static Web Page Test Results for the HP ProLiant BL465c Server

User Connections	5K		30K		80K		200K	
	req/sec	resp time	req/sec	resp time	req/sec	resp time	req/sec	resp time
100	9482.27	12	3762.61	26	1425.04	73	572.85	185
200	9280.24	12	3767.93	52	1426.18	115	572.62	329
500	8297.94	12	3773.08	63	1414.36	268	572.30	1348
1000	8185.66	14	3753.01	120	1409.57	624	567.19	2959

Table 6 Static Web Page Test Results for the HP ProLiant BL465c Server *(continued)*

User Connections	5K		30K		80K		200K	
	req/sec	resp time	req/sec	resp time	req/sec	resp time	req/sec	resp time
1500	6958.18	21	3721.44	122	1405.94	1466	564.29	4101
2000	5993.03	22	3680.87	362	1370.03	1554	564.67	4849

Figure 4 Static Web Page Test Results for the HP ProLiant BL465c Server - Requests per Second

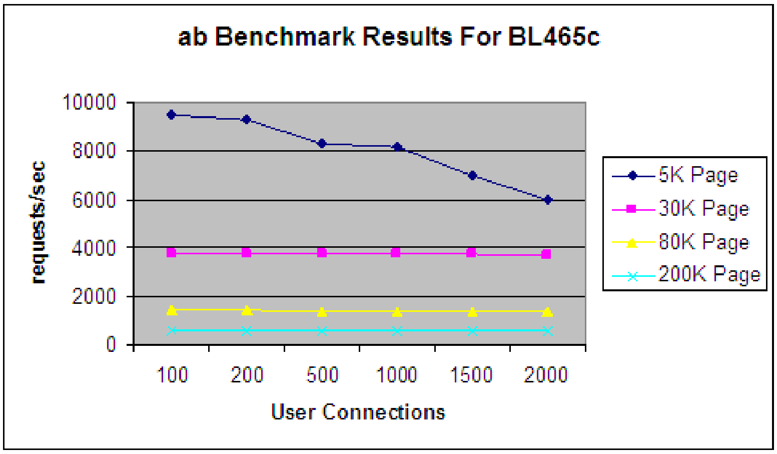
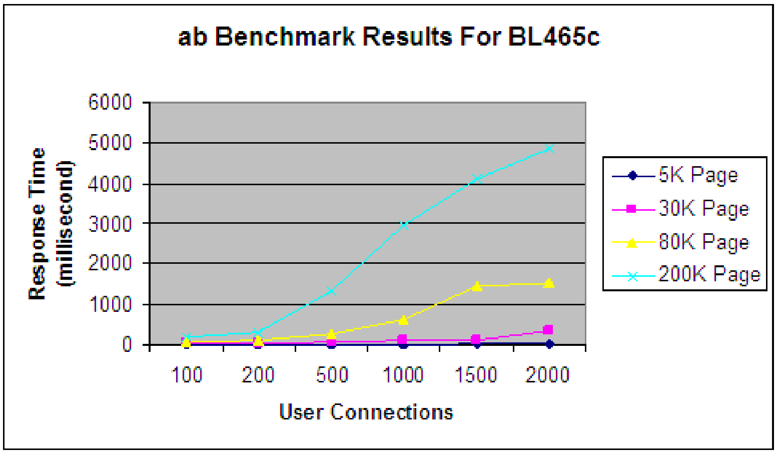


Figure 5 Static Web Page Test Results for the HP ProLiant BL465c Server - Average Response Time



HP Proliant BL480c Server

The HP ProLiant BL480c server is a 2-way server blade designed for maximum performance and scalability. The ProLiant BL480c server can support two Quad-Core Inter Xeon processors, which gives the equivalent of eight processing units per server. Up to 48GB (PC2-5300 DDR2, 667 MHz) of memory is supported. Each full-height blade has a Smart Array P400i Controller that supports four hot-swap SCSI disks in a RAID configuration. An optional dual-port Fiber Channel adapter provides connectivity to HP and third-party storage area networks (SANs). The benchmark results in Table 7 were obtained with a BL480c server configured with two Quad-Core Inter Xeon processors running at 3.0 GHz.

Table 7 Static Web Page Test Results for the HP ProLiant BL480c Server

User Connections	5K		30K		80K		200K	
	req/sec	resp time	req/sec	resp time	req/sec	resp time	req/sec	resp time
100	19446.04	5	3770.73	26	1426.77	76	572.69	187
200	18313.09	8	3768.29	54	1426.88	115	572.84	333
500	16814.04	10	3768.88	65	1425.07	461	572.37	1417
1000	15649.88	20	3766.76	320	1425.52	1010	570.90	3011
1500	14162.79	56	3768.50	339	1424.00	1616	564.46	4108
2000	13810.74	117	3768.70	551	1416.06	2223	568.92	4832
2500	13519.09	130	3718.98	581	1412.64	2734	567.78	6146

Figure 6 Static Web Page Test Results for the HP ProLiant BL480c Server - Requests per Second

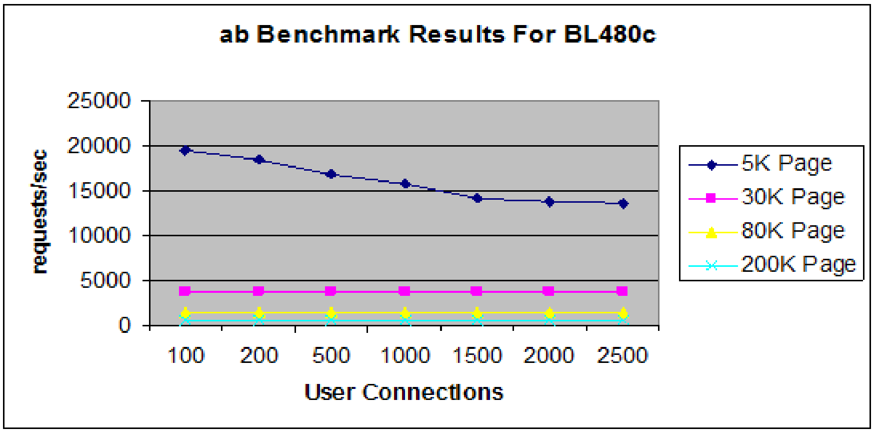
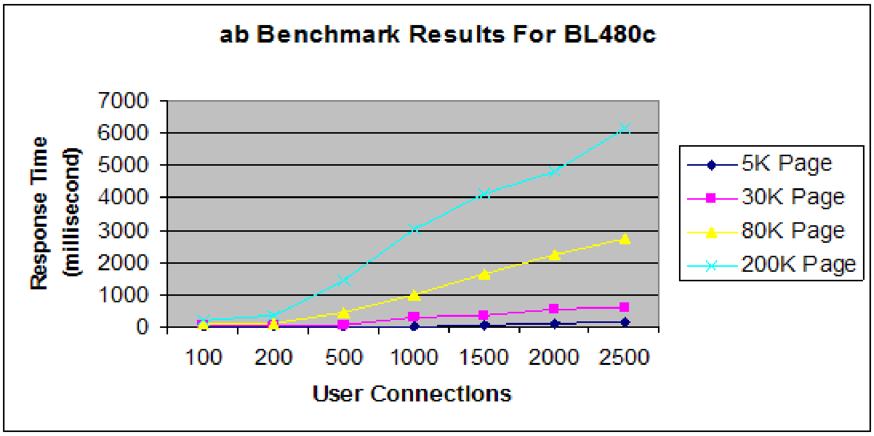


Figure 7 Static Web Page Test Results for the HP ProLiant BL480c Server - Average Response Time



Application Scenario Benchmark

To evaluate the performance of a Tomcat application server, apart from benchmark testing on single pages, the scenario testing to simulate operations on an application in the real world is another important approach to obtain the maximum number of concurrent users a Tomcat server can support in the case of the system response time being met.

JPetStore was used as the test application deployed on the Tomcat Application server. This section provides a detailed description of the test scenario for JPetStore and, how to run Apache JMeter against the test application. The benchmark data includes the number of requests the Tomcat Application server can process per second and the average response time for all requests on each ProLiant Blade server.

JMeter Installation and Configuration

Apache JMeter is an open source Java desktop application. It is designed to load test functional behavior and can be used to measure performance on static and dynamic resources like Servlets, Perl scripts, Java Objects, and so on. It can be used to simulate a heavy load on a Tomcat server to test its capabilities, and can also make a graphical analysis of performance under concurrent load. JMeter version 2.3.1 is used in the following tests. For more information, visit the JMeter site located at:

http://jakarta.apache.org/jmeter/

The steps for installing and configuring JMeter are as follows:

1. Verify the computing environment meets the JMeter test requirements.

   JMeter requires a fully compliant JVM 1.4 or later. JMeter Version 2.2 and later no longer support Java 1.3. Make sure you have the correct version of JRE/JDK installed and set the `JAVA_HOME` environment variable.

2. Download the latest JMeter version from the website and unzip the file to the directory where you want to install JMeter.
3. To run JMeter in GUI mode, run the `jmeter` file in the `JMETER_HOME/bin/` directory.

NOTE: You can edit the `JMETER_HOME/bin/jmeter` file to modify the JMeter parameters or Java Virtual Machine (VM) options. For example, to create the memory consumed by JMeter from the default 256MB to 3GB, you can set the following line in the JMeter file:

HEAP="-Xms3g -Xmx3g -Xmn1g"

For more detailed information, see the JMeter documentation located at:

http://jakarta.apache.org/jmeter/usermanual/index.html

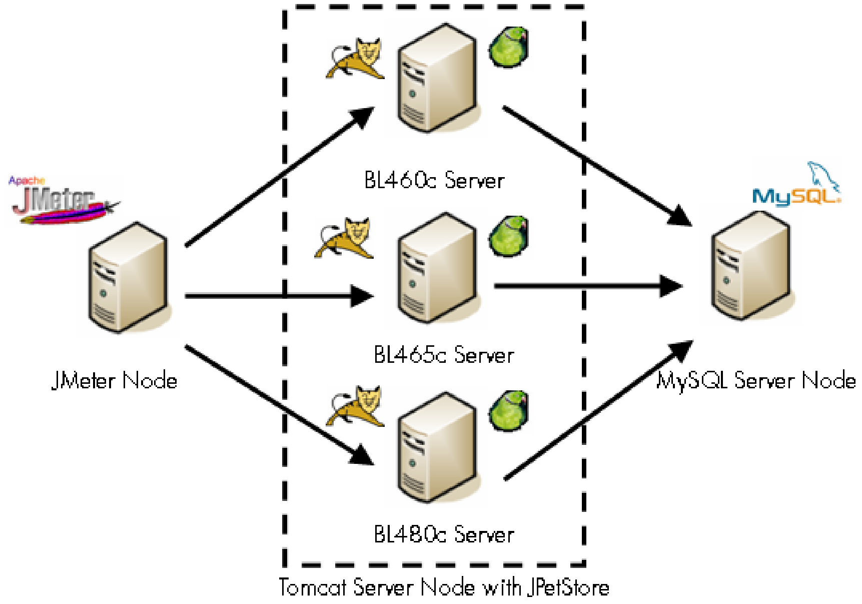
JPetStore Installation and Configuration

JPetStore is a sample application based on the Struts and iBATIS framework. It is a completely rewritten Pet Store application based on Sun's original J2EE Pet Store. For more information on iBATIS and JPetStore, see:

http://ibatis.apache.org/javadownloads.cgi

JPetStore can be deployed in Tomcat and other Java web servers. It supports MySQL, PostgreSQL and other databases. In this document, MySQL is used as the database for JPetStore. Figure 8 displays the architecture of the test environment.

Figure 8 Application Scenario Test Environment Architecture



Before performing the JPetStore installation on the Web server, verify that the following installations and configurations have been completed.

- Tomcat has been installed on the Tomcat server node and configured correctly.
- MySQL has been installed and configured on the MySQL server node. Make sure that MySQL is running correctly.
- Download the MySQL JDBC Driver file, `mysql-connector-java-5.0.4-bin.jar`. Copy the file to `TOMCAT_HOME/lib` directory.

Perform the following steps to install and configure JPetStore.

1. Download the latest version of JPetStore from the iTBATIS website.
2. Unzip the file `JPetStore-5.0.zip` to the appropriate directory.

   In the `JPetStore-5.0/src/ddl/mysql` directory, locate the following files:

   - `jpetstore-mysql-schema.sql`
   - `jpetstore-mysql-create-user.sql`
   - `jpetstore-mysql-dataload.sql`
3. On the MySQL node, run the following commands:

   ```
   mysql -uroot -p < jpetstore-mysql-schema.sql
   mysql -uroot -p < jpetstore-mysql-dataload.sql
   mysql -uroot -p < jpetstore-mysql-create-user.sql
   ```
4. Copy `JPetStore-5.0/build/wars/jpetstore.war` file to the folder `TOMCAT_HOME/webapps`.

   Tomcat automatically deploys the JPetStore application if it is running.
5. Edit the file `webapps/jpetstore/WEB-INF/classes/properties/database.properties` and change the default values to the corresponding configurations in the environment:

   ```
   Driver=org.gjt.mm.mysql.Driver
   Url=jdbc:mysql://<YOUR_MYSQL_SERVER>:3306/JPETSTORE
   Username=jpetstore
   Password=ibatis9977
   ```

Application Scenario Configuration

A JMeter test plan must be created to simulate the requests sent to the server in the scenario. There are two approaches which are generally used to create a test plan: JMeter's Proxy and Badboy. For JMeter, see the JMeter Reference Manual located at:

http://jakarta.apache.org/jmeter/usermanual/index.html

For JMeter's Proxy and recording tests, see the JMeter tutorial located at:

http://jakarta.apache.org/jmeter/usermanual/jmeter_proxy_step_by_step.pdf

For Badboy, see the Badboy website located at:

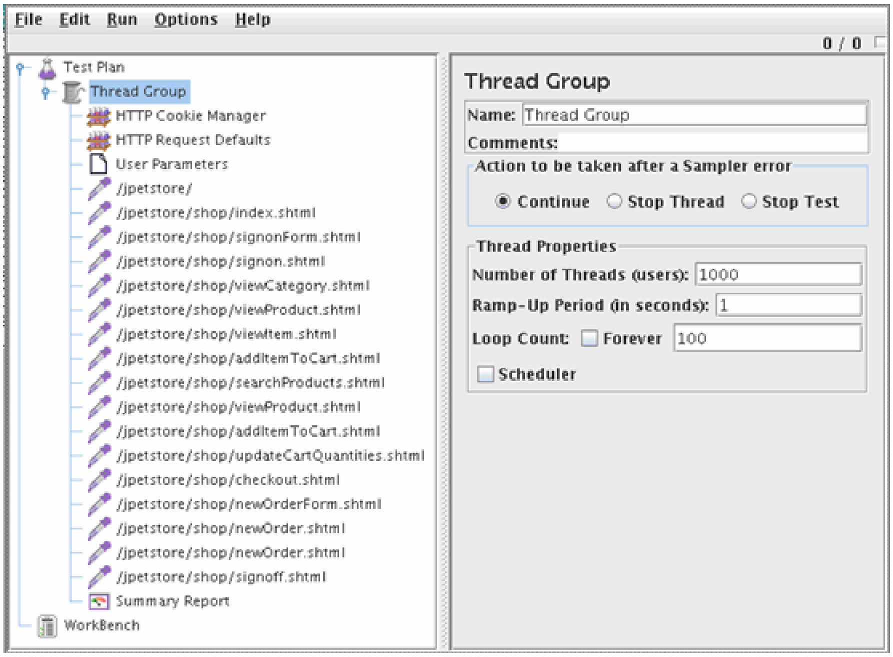
http://www.badboy.com.au/

To simulate a customer's shopping behavior, a scenario is created according to the following typical visiting steps:

1. Visit the index page, and then perform a login action.
2. Look through a kind of fish, and add it to the cart
3. Perform a search action, select a kind of goods from the search result and add it to the cart.
4. Switch to the cart page, and update the goods number to the correct number.
5. Check out and then logout.

Figure 9 displays a JMeter test plan that was created in accordance with the previously described scenario.

Figure 9 Typical JMeter Test Plan



Test Plan
- Thread Group
  - HTTP Cookie Manager
  - HTTP Request Defaults
  - User Parameters
  - /jpetstore/
  - /jpetstore/shop/index.shtml
  - /jpetstore/shop/signonForm.shtml
  - /jpetstore/shop/signon.shtml
  - /jpetstore/shop/viewCategory.shtml
  - /jpetstore/shop/viewProduct.shtml
  - /jpetstore/shop/viewItem.shtml
  - /jpetstore/shop/addItemToCart.shtml
  - /jpetstore/shop/searchProducts.shtml
  - /jpetstore/shop/viewProduct.shtml
  - /jpetstore/shop/addItemToCart.shtml
  - /jpetstore/shop/updateCartQuantities.shtml
  - /jpetstore/shop/checkout.shtml
  - /jpetstore/shop/newOrderForm.shtml
  - /jpetstore/shop/newOrder.shtml
  - /jpetstore/shop/newOrder.shtml
  - /jpetstore/shop/signoff.shtml
  - Summary Report

WorkBench

Thread Group

Name: Thread Group

Comments:

Action to be taken after a Sampler error

● Continue ○ Stop Thread ○ Stop Test

Thread Properties

Number of Threads (users): 1000

Ramp-Up Period (in seconds): 1

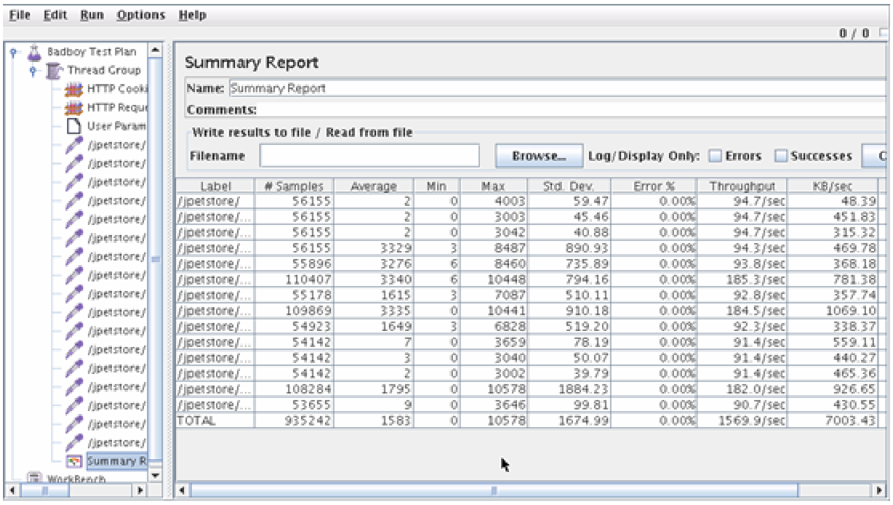
Loop Count: □ Forever 100

□ Scheduler

Running the Scenario

Select the **Thread Group** element in the JMeter tree and increase the Number of Threads (users) from the default value to the value you want to test. Next, enter a value in the **Loop Count** field. For instance, if the thread number is 1000, and the loop count is 100, then JMeter creates 1000 users at the same time and each user repeatedly runs the test plan 100 times. Before starting the test, select **Summary Report** in the tree. While the test is running, you can watch the statistics from the page until the end of the test. At the end of the test, a Summary Report is generated, similar to the example shown in Figure 10.

Figure 10 JMeter Summary Report



Summary Report

Name: Summary Report

Comments:

Write results to file / Read from file

Filename

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	KB/sec
/jpetstore/	56155	2	0	4003	59.47	0.00%	94.7/sec	48.39
/jpetstore/...	56155	2	0	3003	45.46	0.00%	94.7/sec	451.83
/jpetstore/...	56155	2	0	3042	40.88	0.00%	94.7/sec	315.32
/jpetstore/...	56155	3329	3	8487	890.93	0.00%	94.3/sec	469.78
/jpetstore/...	55896	3276	6	8460	735.89	0.00%	93.8/sec	368.18
/jpetstore/...	110407	3340	6	10448	794.16	0.00%	185.3/sec	781.38
/jpetstore/...	55178	1615	3	7087	510.11	0.00%	92.8/sec	357.74
/jpetstore/...	109869	3335	0	10441	910.18	0.00%	184.5/sec	1069.10
/jpetstore/...	54923	1649	3	6828	519.20	0.00%	92.3/sec	338.37
/jpetstore/...	54142	7	0	3659	78.19	0.00%	91.4/sec	559.11
/jpetstore/...	54142	3	0	3040	50.07	0.00%	91.4/sec	440.27
/jpetstore/...	54142	2	0	3002	39.79	0.00%	91.4/sec	465.36
/jpetstore/...	108284	1795	0	10578	1884.23	0.00%	182.0/sec	926.65
/jpetstore/...	53655	9	0	3646	99.81	0.00%	90.7/sec	430.55
TOTAL	935242	1583	0	10578	1674.99	0.00%	1569.9/sec	7003.43

Benchmark Results

Table 8 shows the requests per second and average response time of the scenario described in the previous section. The response time is defined as the time it takes the Tomcat server to send the object of an HTTP request back to the client. The requests per second is the average request number Tomcat is capable of handling (per second) in the particular scenario.

Table 8 JMeter Application Test Results

User Connections	BL460c		BL465c		BL480c	
	requests/sec	response time (min/sec)	requests/sec	responsetime (min/sec)	requests/sec	response time (min/sec)
100	1677.7	58	1166.8	83	1593.5	61
200	1763.6	112	1162.1	170	1730.4	114
500	1733.4	286	1147.3	433	1717.4	289
1000	1721.0	577	1123.9	884	1664.0	594
1500	1631.0	902	1083.6	1356	1613.5	909
2000	1619.5	1192	1110.6	1757	1575.6	1230

Figure 11 JMeter Application Scenario Test Results – Requests per Second

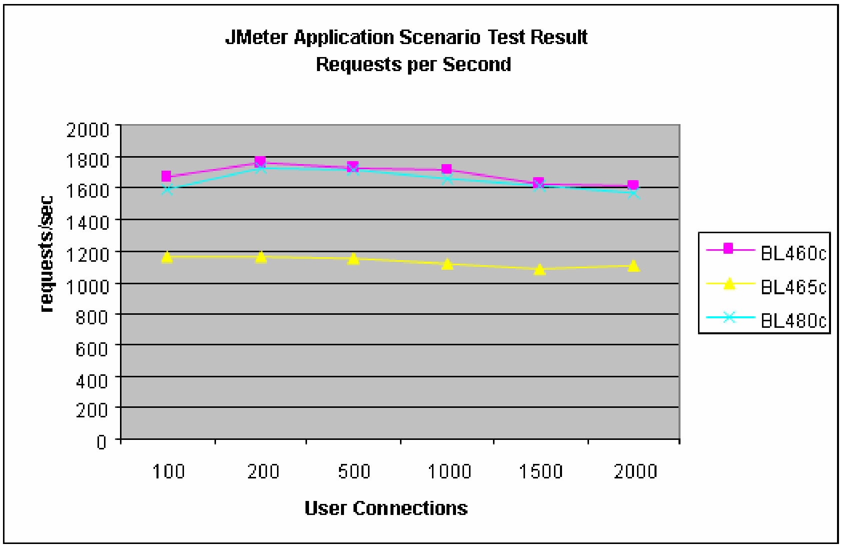
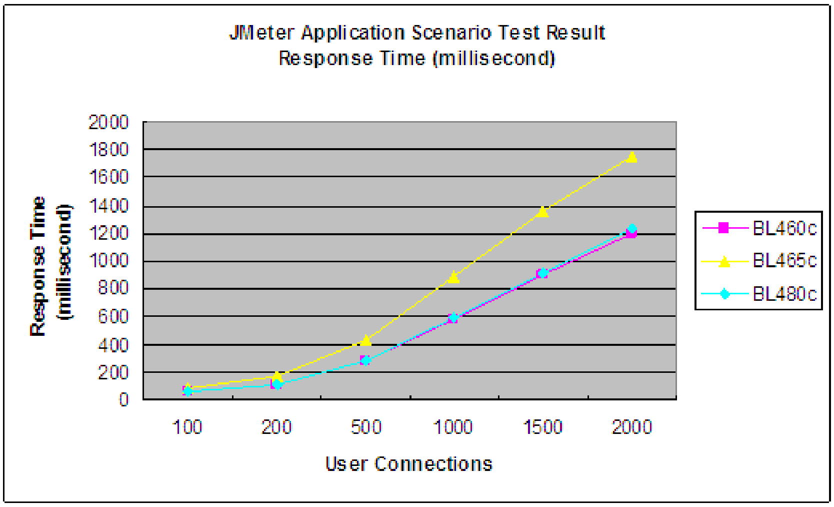


Figure 12 JMeter Application Scenario Test Results – Response Time



Summary

The performance of the Tomcat Application server is impacted by several factors such as network throughput, system processing capability, concurrent user connections, and so on. The data shown in this document were obtained under the precondition that network throughput was stable. However, it is not generally the case that all users are simultaneously requesting service from the application server. Among the different applications, the ratio of concurrent users to total users of a certain application is different, but usually varies between 10% and 30%. Another difference is the interval between requests of each user. The longer the interval is, the more users the Tomcat server can serve. In the benchmark testing described in this document, the interval between requests of each user was not considered. Therefore, for applications in the real world, the number of concurrent user connections that the Tomcat Application server can process on these three-blade servers may be much greater than shown in the document. When there are millions of concurrent user connections, the Tomcat cluster and load-balancing technology can be considered to scale out the Tomcat environment based on HP ProLiant c-Class Blade servers.

Resources

For additional information on the hardware and software used in the Tomcat tests, see the following websites:

HP Open Source Middleware Stacks (OSMS)

http://h71028.www7.hp.com/enterprise/cache/321150-0-0-0-121.html

HP BladeSystem

http://www.hp.com/go/bladesystem/

http://h18004.www1.hp.com/products/blades/components/bladeservers.html

Apache Tomcat

http://tomcat.apache.org/index.html

Apache JMeter

http://jakarta.apache.org/jmeter/