

Agility Database Scalability Testing

V1.6

November 11, 2012

Prepared by



on behalf of



Table of Contents

1	Introduction.....	4
1.1	Brief	4
2	Scope	5
3	Test Approach.....	6
4	Test environment setup	7
5	Test data used	7
6	Test Schedule.....	8
6.1	Baseline PIM Sessions against the server. 250,000 SKU's	8
6.1.1	Test 1.	8
6.1.2	Test 2.	8
6.1.3	Test 3.	8
6.2	Baseline PIM Sessions against the server. 2.5 Million SKU's	8
6.2.1	Test 4.	8
6.2.2	Test 5.	9
6.2.3	Test 6.	9
7	Results Summary	10
7.1	250k SKU database End User Summary.....	10
7.2	2.5 Million SKU database End User Summary	10
8	Conclusion	11
9	Summary.....	11
10	Appendix A	11
10.1	Weblogic Service Monitor Descriptions	11
10.2	SQL Advanced Service Monitoring Descriptions	14
11	Results - Detailed.....	15
11.1	Test 1	15
11.2	Test 2	20
11.3	Test 3	25
11.4	Test 4	29
11.5	Test 5	33
11.6	Test 6	37

Document Scope

This document is to provide information on scalability testing regarding the PIM environment developed by Agility Multichannel.

The documentation is designed for the reader to have a medium knowledge of IT to understand the concepts of environments and performance statistics.

The following information will be included within this document:

Introduction – Which details why and what we are testing.

Test Approach – How we are going to test.

Test Schedule – Timelines of tests run.

Results – Completed and stored data analysis from each test run within the test schedule.

Conclusion – What the information has given us.

1 Introduction

1.1 Brief

Agility Multichannel is a software company that develops a product information management system.

Agility Multichannel has approached Fusion Systems to help it benchmark and document the scalability of its software. This is detailed in section 3. Fusion Systems was brought in as a consultant to provide a third party unbiased view of the testing requirements and results.

Fusion Systems is a technology company providing support and consultancy to third party companies. Fusion Systems has been involved with full cycle application testing within the broadcasting, insurance and financial business sectors.

2 Scope

Out of the many different types of testing this exercise will be focusing on scalability testing.

- Functional Testing
- Non Functional Testing
- Performance Testing
- **Scalability Testing**
- Tuning
- Penetration testing

The scope of the scalability testing will be focused on a non-customer specific data set and infrastructure for the following reasons:

The Agility application operates with different business processes, different infrastructure and different data and data model for each customer.

The platforms used at each customer can vary in a number of ways. Examples of permutations are as follows:

- Weblogic + Oracle
- Weblogic + SQL
- Apache + Websphere + Oracle
- Apache + Websphere + SQL
- IBM HTTP Server + Websphere + Oracle
- IBM HTTP Server + Websphere + SQL

In addition to the above there can also be different subversions of these components in use as well as a mixture of operating systems and environments on both server and client sides.

Within each customer organisation there can also be different network configurations including load balancing / Server locations / Different network standards (Lease lines / MPLS)

Also each customer can have different data and data model and amount of data

Finally each customer has different numbers of users with different roles and responsibilities and use cases and usage patterns.

With the above in mind we plan a generic subset of tests which will not be representative of a specific client environment but will show the scalability of the Agility system against a subset of data defined internally consisting of 250,000 SKU records with at least 40 attributes (10 million attribute instances) then ramping up to 2.5 Million with at least 100 Attributes (250 million attribute instances).

3 Test Approach

The test will be comprised of a content management (PIM) harness – To represent 'business as usual' traffic for managing product information through the web user interface.

The PIM Harnesses will be run to show performance against the database. 5 Runs will be used at the following loads:

- 10 PIM Sessions
- 30 PIM Sessions
- 50 PIM Sessions

The above tests will be run on two different data sets as follows:

- 250,000 Sku's x 40 attribute instances = 10 million attribute instances
- 2.5 Million Sku's x 100 attribute instances = 250 million attribute instances

The attribute instance is a key unit of measure for data in Agility as it equates to a record in the area of the database used to store attribute values.

Timing points will be taken from the PIM user harness. Performance statistics will be completed on the server using Performance monitor showing the following:

- CPU
- Memory
- Network
- Weblogic Stats
- SQL Stats

The test harnesses will report transaction times and response times that the end user would experience. During each test the environment will be rolled back to the original settings / database content to show clean runs against a clean environment so not to skew individual results.

All test harnesses include random wait times. These wait times have been designed to mimic random end user delays during normal use to give varying load patterns. Otherwise the scenarios wouldn't be as representative as they would be in a production environment.

In the opinion of Agility Multichannel one PIM test session in the test harness equates to more than the load from one real world user because the tests perform actions more frequently. However it is hard to put a specific value on that e.g. one test harness session might be equivalent to the load from 2 real world users.

4 Test environment setup

The following information details the environment setup:

Application Server

HP DL360 G7

Processor Xeon E5506 – Quad Core 2.13Ghz

Memory – 72GB (Capped at 4GB)

Drive Configuration – 2 x SATA 250GB (5400 rpm) in a mirror

Database Server

HP Proliant DL380 G7

Processor Xeon E5620 – Quad Core 3.40Ghz

Memory – 72GB (Capped at 4GB)

Drive Configuration – 4 x SAS 500GB (7200 rpm) in a raid 5

Both application and database servers will be run within a Virtual environment using VMWARE ESXi 4. These will be configured to use all the processing power of each machine for test purposes. The network interface will be bound to each server to simulate full network interface usage. The memory will be capped at 4GB within the VM environment and increase when and IF required.

- Both systems will have Windows Server 2008 R2 64-bit including all patching up to the 1st November 2012.
- The application server will consist of Weblogic 10.3.5
- The database server will consist of SQL Server 2008 R2 SP2

The above configuration will allow for hardware changes if we deem the performance is not adequate during testing to give a baseline minimum requirement for hardware.

The network path between the environments consists of a gigabyte switch. Each environment is hosted on a dedicated network interface connecting directly to the switch.

5 Test data used

The initial 250,000 SKU data set was based on a customer database. This data set consists of rich data with an average of 40 Attributes associated to each SKU.

The second data set consisting of 2.5 Million records with 100 attributes against each SKU to provide adequate test data to test against. This also includes the previous data set of 250,000k.

6 Test Schedule

The following information details the test schedule which was completed during the testing phase. All tests include random waits between 3 and 10 seconds between tests:

6.1 Baseline PIM Sessions against the server. 250,000 SKU's

6.1.1 Test 1.

10 Sessions run concurrently on a clean environment running the following:

The load consists of the following - each connection runs a sequence of 10 tasks randomly selected from a predefined set consisting of:

- Create object and attributes then unlink
- Create object and attributes then delete
- Create object with no attributes then unlink
- Create object with no attributes then delete
- Global search

6.1.2 Test 2.

30 Sessions run concurrently on a clean environment running the following:

The load consists of the following - each connection runs a sequence of 10 tasks randomly selected from a predefined set consisting of:

- Create object and attributes then unlink
- Create object and attributes then delete
- Create object with no attributes then unlink
- Create object with no attributes then delete
- Global search

6.1.3 Test 3.

50 Sessions run concurrently on a clean environment running the following:

The load consists of the following - each connection runs a sequence of 10 tasks randomly selected from a predefined set consisting of:

- Create object and attributes then unlink
- Create object and attributes then delete
- Create object with no attributes then unlink
- Create object with no attributes then delete
- Global search

6.2 Baseline PIM Sessions against the server. 2.5 Million SKU's

6.2.1 Test 4.

10 Sessions run concurrently on a clean environment running the following:

The load consists of the following - each connection runs a sequence of 10 tasks randomly selected from a predefined set consisting of:

- Create object and attributes then unlink
- Create object and attributes then delete
- Create object with no attributes then unlink
- Create object with no attributes then delete
- Global search

6.2.2 Test 5.

30 Sessions run concurrently on a clean environment running the following:

The load consists of the following - each connection runs a sequence of 10 tasks randomly selected from a predefined set consisting of:

- Create object and attributes then unlink
- Create object and attributes then delete
- Create object with no attributes then unlink
- Create object with no attributes then delete
- Global search

6.2.3 Test 6.

50 Sessions run concurrently on a clean environment running the following:

The load consists of the following - each connection runs a sequence of 10 tasks randomly selected from a predefined set consisting of:

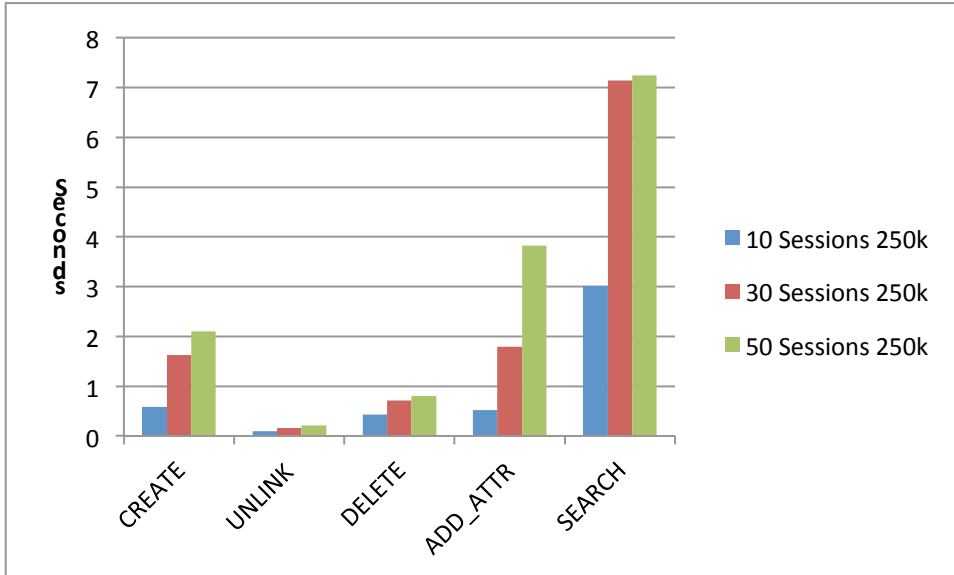
- Create object and attributes then unlink
- Create object and attributes then delete
- Create object with no attributes then unlink
- Create object with no attributes then delete
- Global search

7 Results Summary

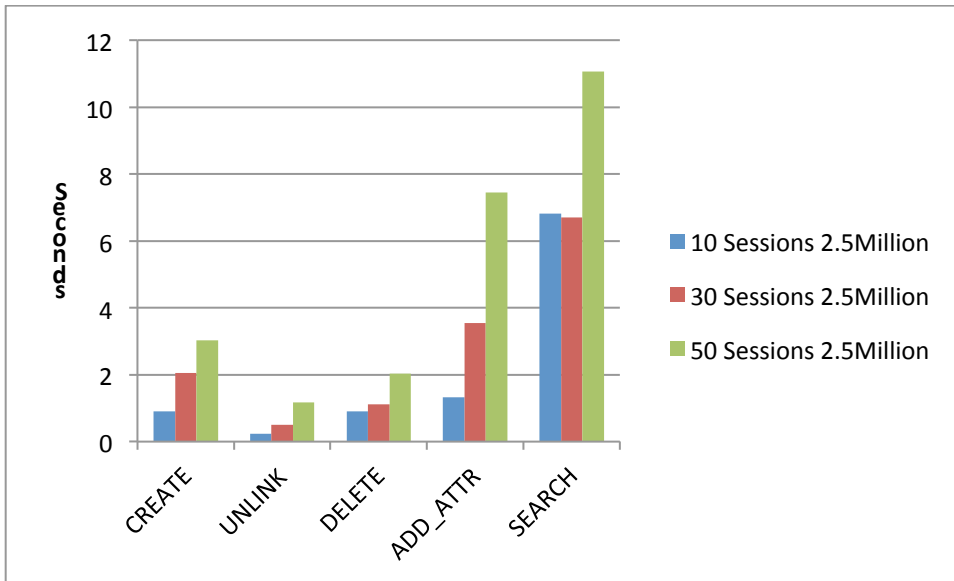
The results showed the following end user times against each trigger using the two sets of data for 250k SKU's and 2.5 Million records.

The vertical scale is seconds.

7.1 250k SKU database End User Summary



7.2 2.5 Million SKU database End User Summary



8 Conclusion

All results and statistics have been stored and can be accessed on request to validate findings.

All reports used have been stored since only a subset will be shown within the document to report against to reduce content. These can be provided on request.

You can see clearly that the end user experience profile changes the more loads which are introduced to the system. This is normal and what you would expect within any environment.

What is noticeable within the test is that the search does degrade as expected when the datasets are expanded, but not as much as we had anticipated.

During the test phase using the 2.5 Million SKU data set you can see obvious peaks within the performance graphs. We believe that these peaks were due to the agent we are using to monitor statistics. You can see the agent did not impact the testing as the end user experience did not change dramatically during each of the tests. The tests were run multiple times to deduce this conclusion.

Tests 1 and Test 4 were compared showing that the database completed more work with the second data set; however the environment ran as expected with no major issues.

The environment showed adequate health when running up to 50 Sessions concurrently on the two data sets.

9 Summary

This document provides results for PIM users for Scalability testing based on a generic environment.

With the information we have we can make a statement that on our test environment the Agility SQL Database shows little performance degradation against a database with 2.5 Million SKU's.

10 Appendix A

10.1 Weblogic Service Monitor Descriptions

Per Connection Pool

The following metrics are collected per connection pool on a WebLogic server:

Metric	Explanation	MBean Name
FailuresToReconnectCount	The number of times that the connection pool failed to reconnect to a data store.	JDBCConnectionPool Runtime
ActiveConnectionsCurrentCount	The current number of active connections in a JDBC connection pool.	JDBCConnectionPool Runtime
ActiveConnectionsHighCount	The highest number of active connections in a JDBC connection pool.	JDBCConnectionPool Runtime
LeakedConnectionsCount	The total number of connections that have been checked out of, but not returned to, the connection pool.	JDBCConnectionPool Runtime
CurrCapacity	The current number of database connections in the JDBC connection pool.	JDBCConnectionPool Runtime
NumAvailable	The number of available sessions in the session pool that are not currently being used.	JDBCConnectionPool Runtime
WaitingForConnectionCurrentCount	The current number of requests that are waiting for a connection to the connection pool.	JDBCConnectionPool Runtime

Per EJB

The following metrics are collected per Enterprise JavaBean (EJB) on a WebLogic server:

AccessTotalCount	The total number of times an attempt was made to get an Enterprise JavaBeans (EJB) instance from the free pool.	StatelessEJBRuntime
BeansInCurrentUseCount	The number of EJB instances in the free pool which are currently in use.	StatelessEJBRuntime
CachedBeansCurrentCount	The total number of EJBs that are in the execution cache.	StatefulEJBRuntime
ActivationCount	The number of EJBs that have been activated.	StatefulEJBRuntime
PassivatedCount	The number of EJBs that have been passivated.	EJBCacheRuntimeMBean

cacheAccessCount	The number of attempts that were made to access an EJB in the cache.	EJBCacheRuntimeMB ean
cacheMissCount	The number of times that an attempt to access an EJB in the cache failed.	EJBCacheRuntimeMB ean
Other		
The following metrics are collected from the WebLogic system, the Java Virtual Machine, or the thread pool:		
HeapSizeCurrent	The amount of memory, in bytes, that is in the WebLogic server's Java Virtual Machine (JVM) heap.	JVMRuntime
HeapFreeCurrent	The current amount of free memory, in bytes, that is in the WebLogic server's JVM heap.	JVMRuntime
OpenSocketsCurrentCount	The current number of sockets on the server that are open and receiving requests.	ServerRuntime
AcceptBacklog	The number of requests that are waiting for a TCP connection.	Server
ExecuteThreadCurrentIdleCount	The number of threads in the server's execution queue that are idle or which are not being used to process data.	ExecuteQueueRuntime
PendingRequestCurrentCount	The number of pending requests that are in the server's execution queue.	ExecuteQueueRuntime
TransactionCommittedTotalCount	The total number of transactions that have been processed by the WebLogic server.	TransactionResourceRuntime
TransactionRolledBackTotalCount	The total number of transactions that have been rolled back.	TransactionResourceRuntime
InvocationTotalCount	The total number of times that a servlet running on the WebLogic server was invoked.	ServletRuntime

10.2 SQL Advanced Service Monitoring Descriptions

Metric	Explanation	Performance Object
Lock Wait / Sec.	The amount of time, in seconds, that the server waits for a database lock.	SQL ServerLocks
Lock Requests / Sec.	The number of new database locks and lock conversions that are requested from the lock manager every second.	SQL ServerLocks
Average Wait Time	The average time, in milliseconds, that it takes for database locks to clear.	SQL ServerLocks
User Connections	The number of user connections to the database that are allowed.	SQL ServerGeneral Statistics
Transactions / Sec.	The number of transactions started for the databases across the host per second.	SQL ServerDatabase
Data File(s) Size / KB	The cumulative size of all the files in all of the databases on the host system.	SQL ServerDatabases
Total Latch Wait Time	The total time, in milliseconds, that it takes to complete the latch requests that were waiting over the last second.	SQLServerLatches
Latch Waits / Sec.	The number of latch requests that were not immediately granted, and which waited before being granted.	SQLServerLatches
Average Latch Wait Time	The average time, in milliseconds, that latch requests had to wait before being granted.	SQLServerLatches
Maximum Workspace Memory	The maximum amount of memory, in kilobytes, that the server has available to execute such processes as sort, bulk copy, hash, and index creation.	SQL ServerMemory Manager
Connection Memory	The total amount of dynamic memory, in kilobytes, that the server is using to maintain connections.	SQL ServerMemory Manager
SQL Cache Memory	The amount of memory, in kilobytes, that the server is using for the dynamic SQL cache.	SQL ServerMemory Manager
Total Server Memory	The total amount of committed memory from the buffer pool, in kilobytes, that the server is	SQL ServerMemory Manager

using.

Appendix B

11 Results - Detailed

Please see Appendix A for graph descriptions. All statistics for each description is available on request for each test.

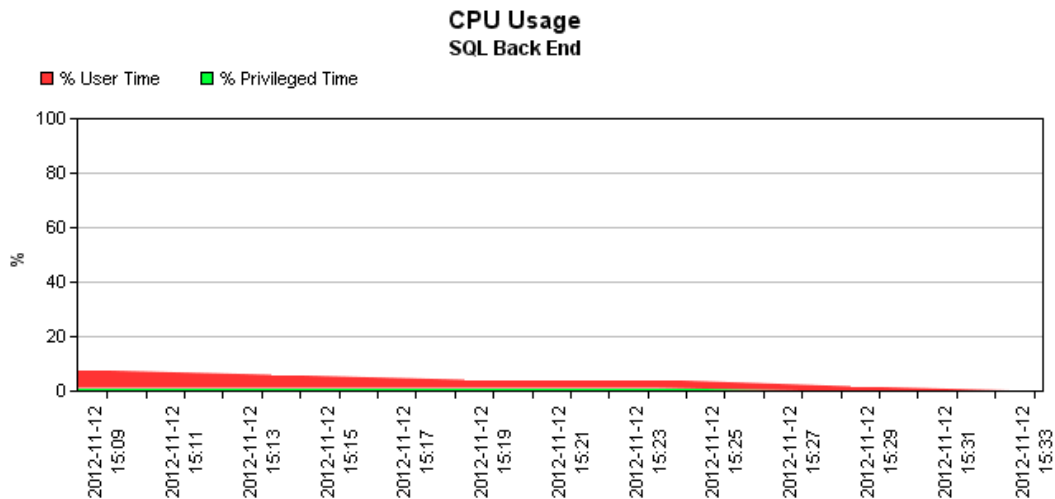
11.1 Test 1

The end user timings were as follows:

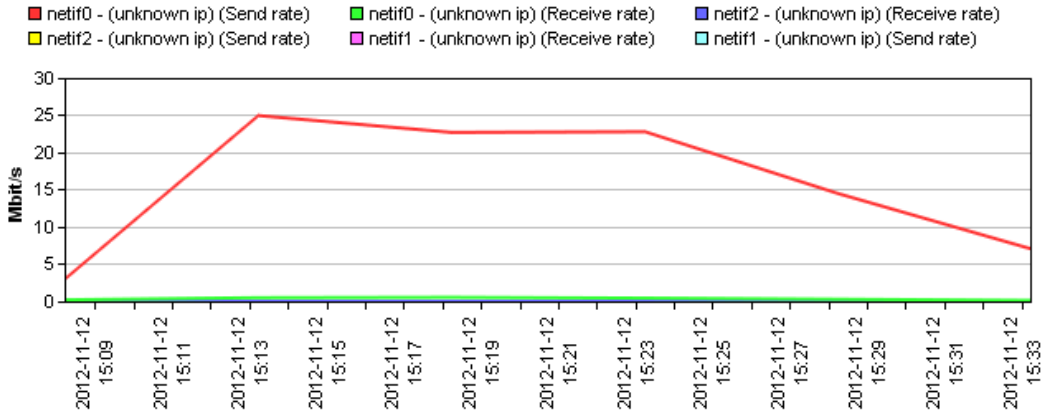
Throughout the stats Minimum, Average and Maximum times in seconds were provided as follows:

Description	Min (Secs)	Average (Secs)	Max (Secs)
CREATE	0.198	0.586	12.304
UNLINK	0.027	0.102	3.212
DELETE	0.212	0.425	3.961
ADD_ATTR	0.098	0.514	2.145
SEARCH	0.116	3.009	28.74

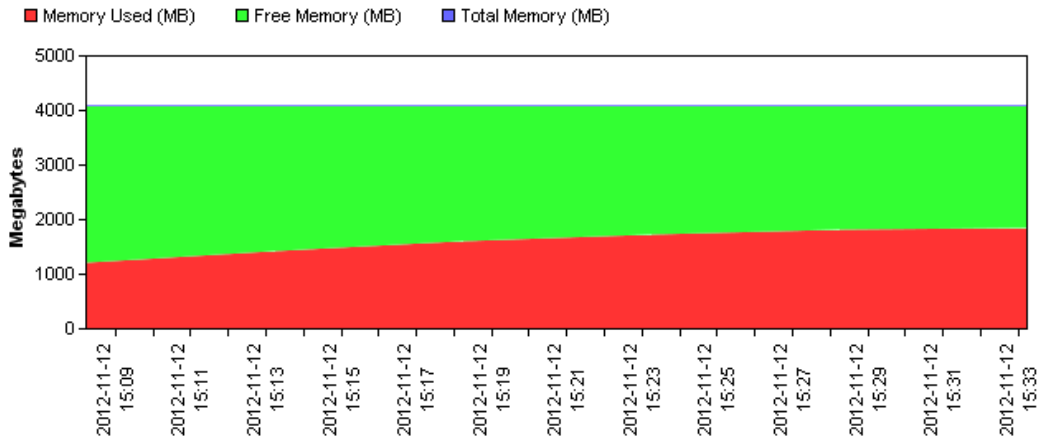
The Environment statistics were as follows:



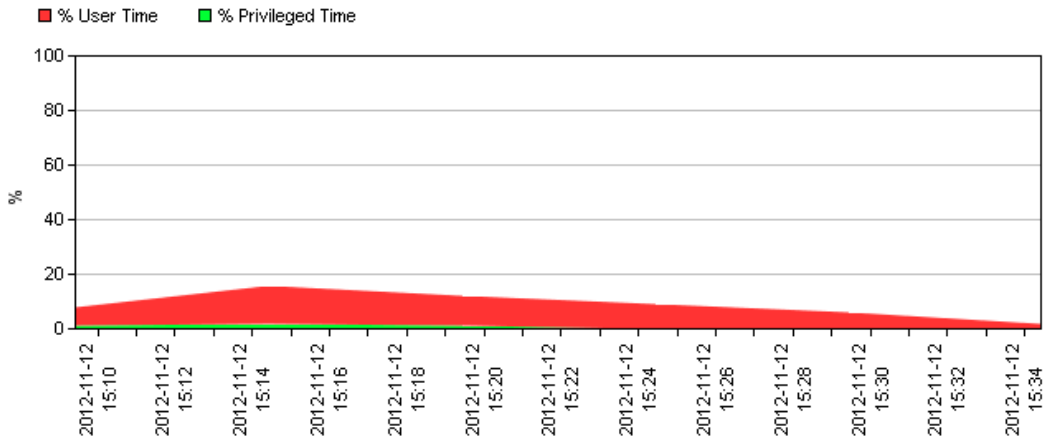
**Network Usage
SQL Back End**



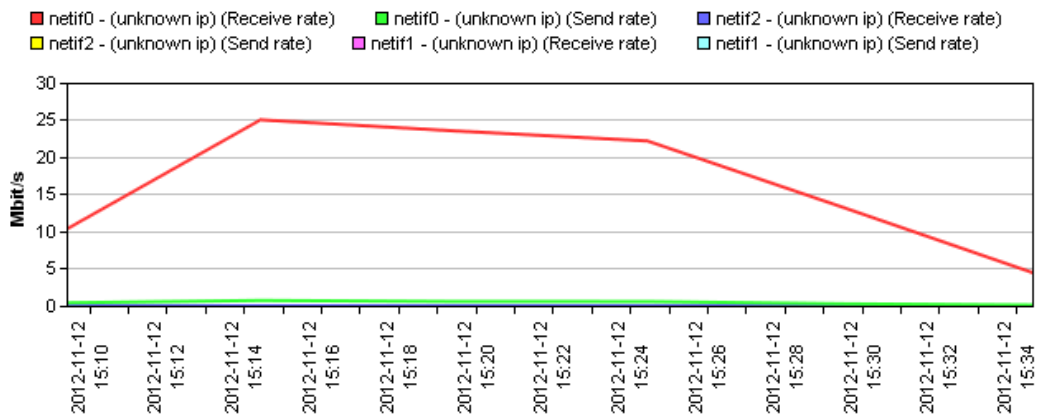
**Free Memory
SQL Back End**



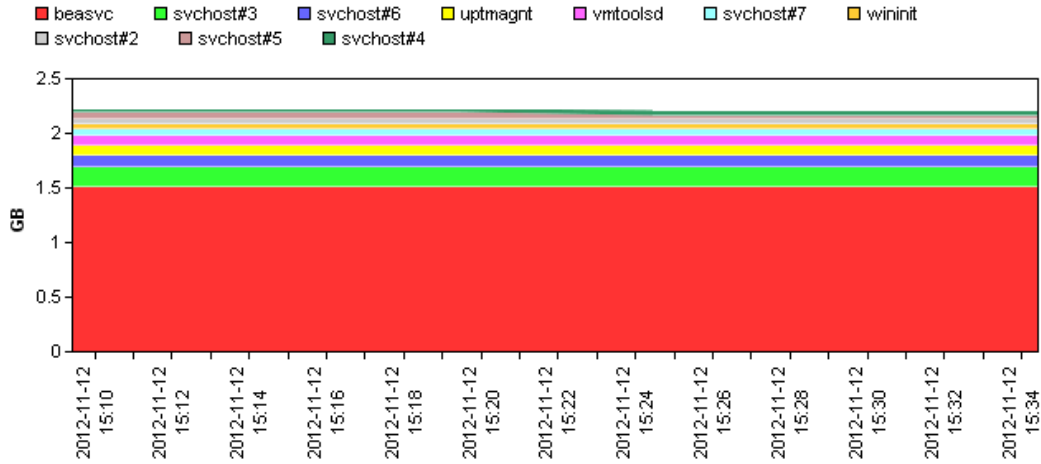
CPU Usage
Weblogic Front End



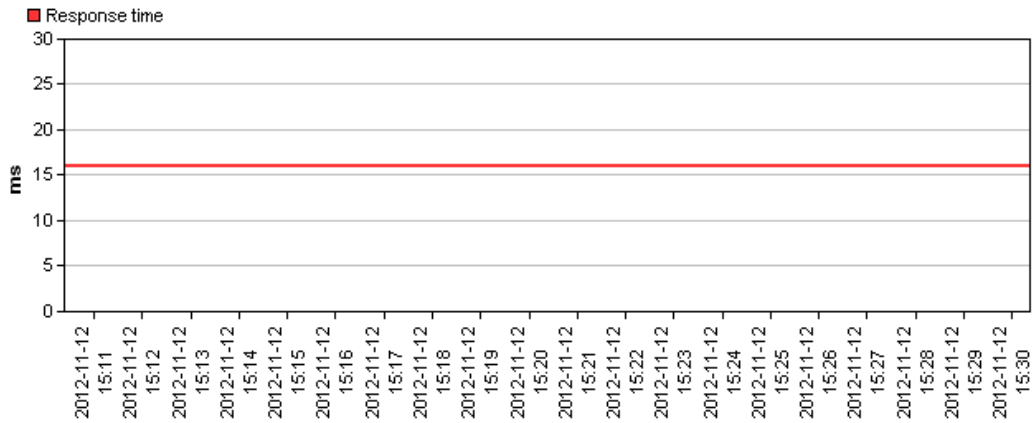
Network Usage
Weblogic Front End



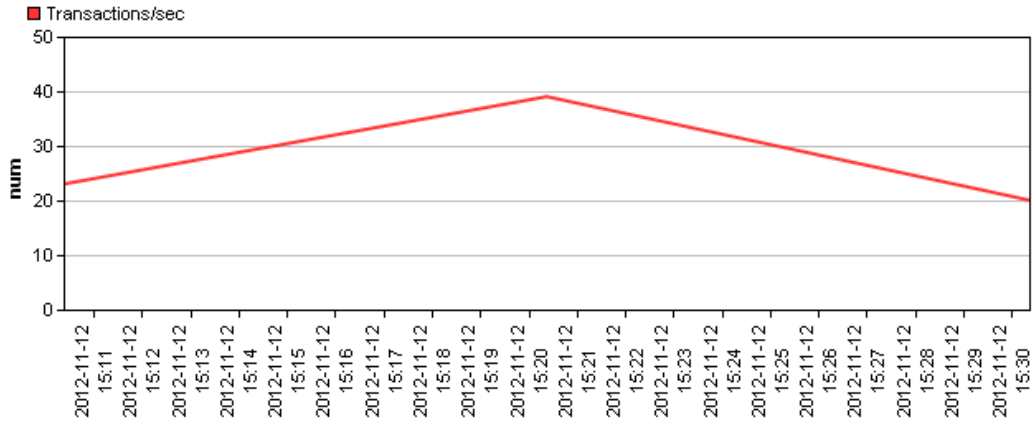
Memory Size Workload Characterization
Weblogic Front End



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Response time



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Transactions/sec



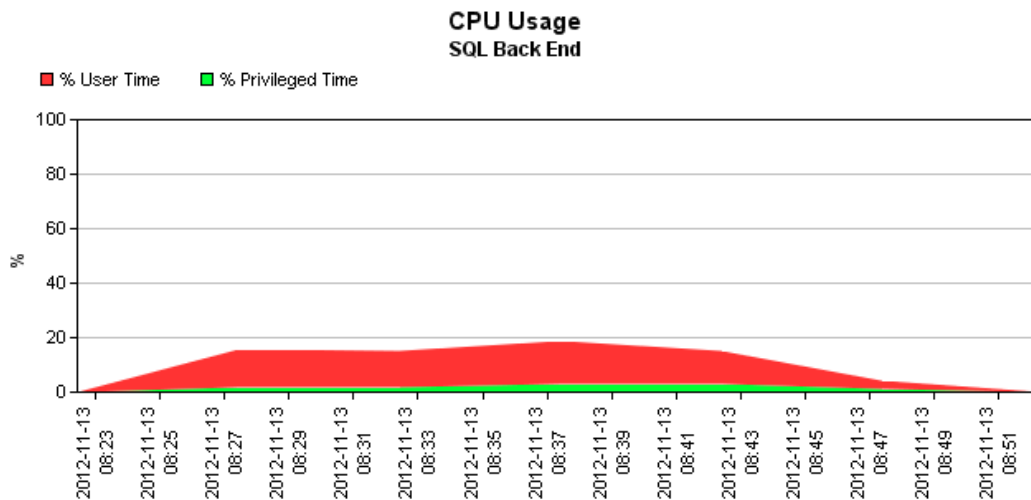
11.2 Test 2

The end user timings were as follows:

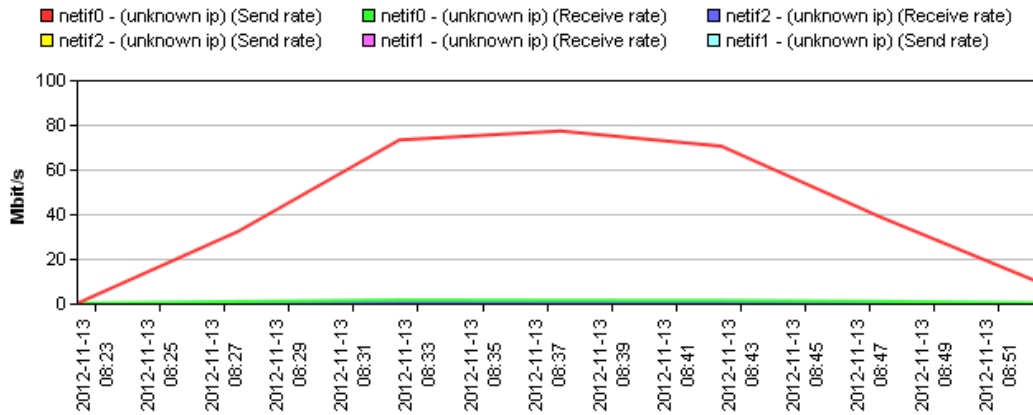
Throughout the stats Minimum, Average and Maximum times in seconds were provided as follows:

Description	Min (Secs)	Average (Secs)	Max (Secs)
CREATE	0.202	1.627	25.87
UNLINK	0.028	0.162	2.107
DELETE	0.26	0.709	5.454
ADD_ATTR	0.106	1.799	10.861
SEARCH	0.115	7.138	67.223

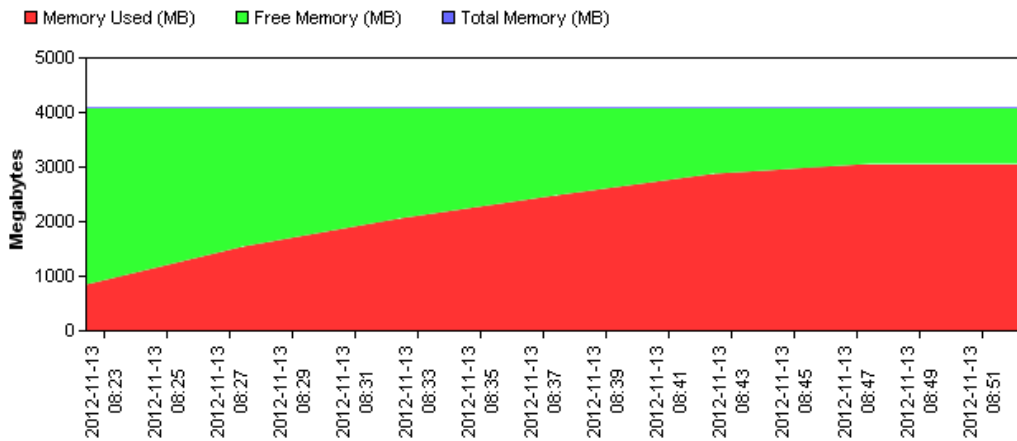
The Environment statistics were as follows:



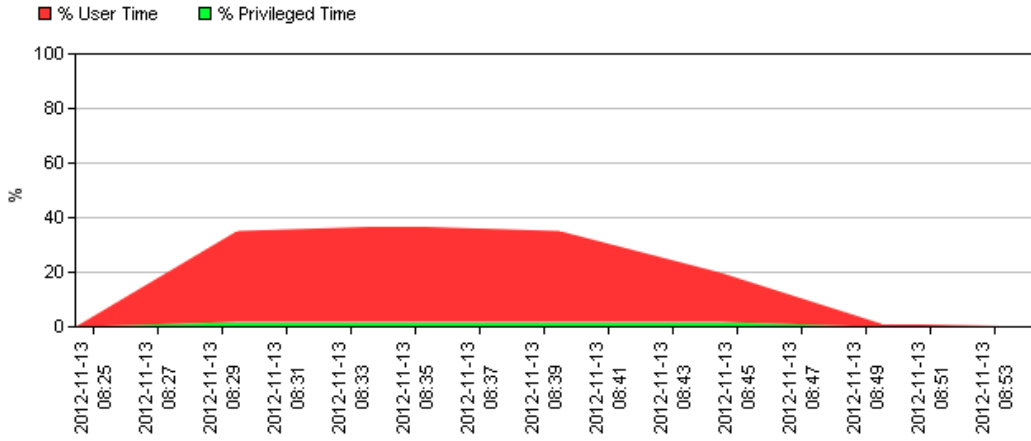
**Network Usage
SQL Back End**



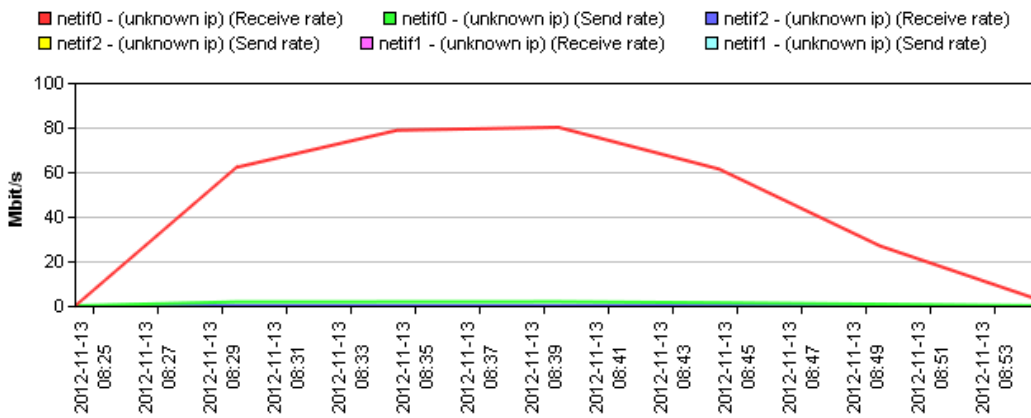
**Free Memory
SQL Back End**



CPU Usage
Weblogic Front End

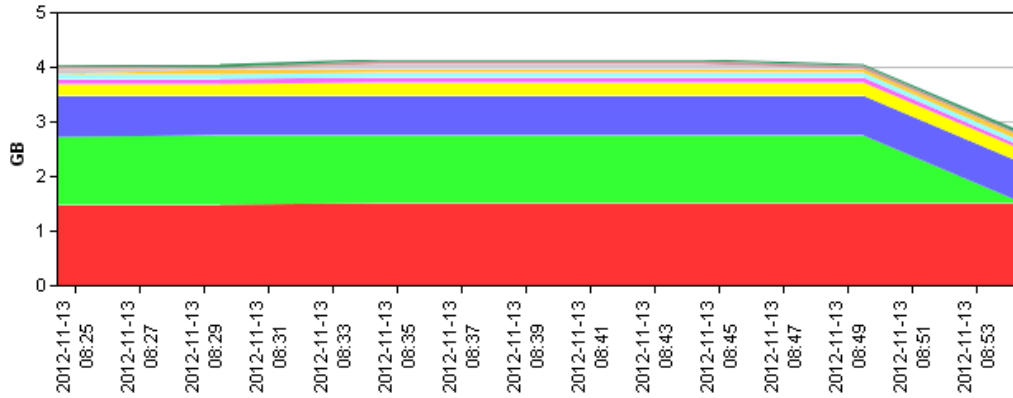


Network Usage
Weblogic Front End

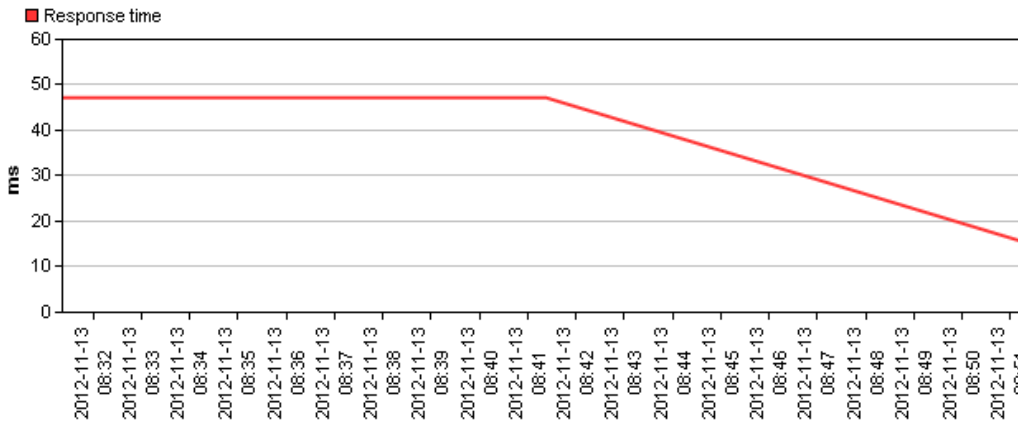


Memory Size Workload Characterization
Weblogic Front End

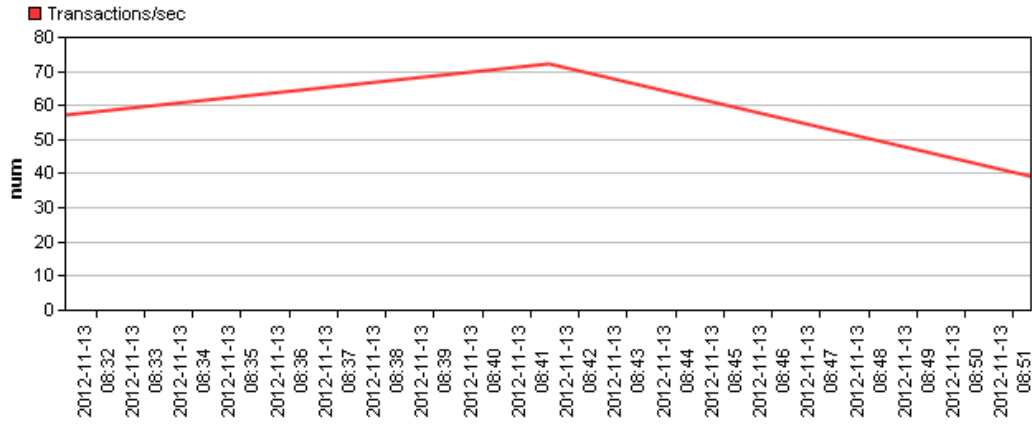
■ beasvc ■ java ■ mmc ■ svchost#3 ■ vntoolsd ■ uptmagnt ■ svchost#5 ■ VMwareUser
■ svchost#2 ■ svchost



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Response time



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Transactions/sec



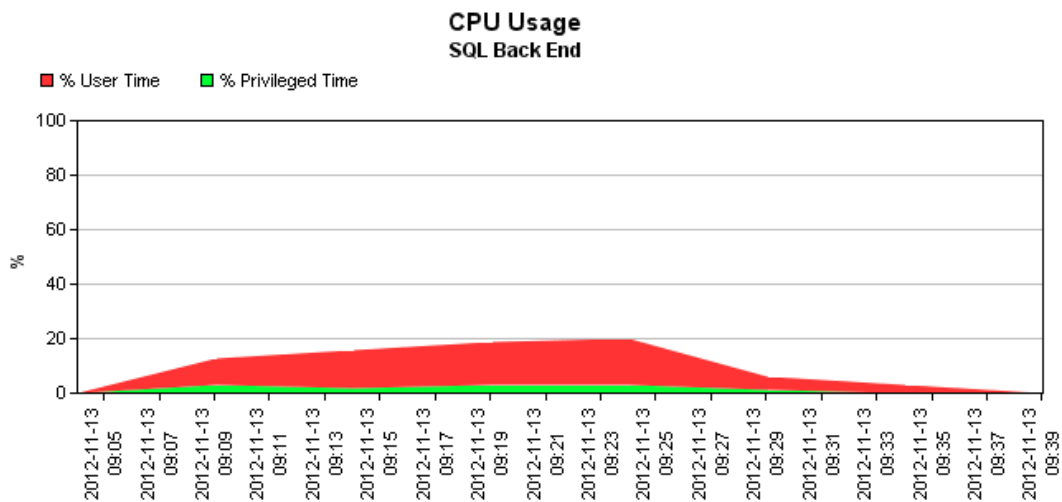
11.3 Test 3

The end user timings were as follows:

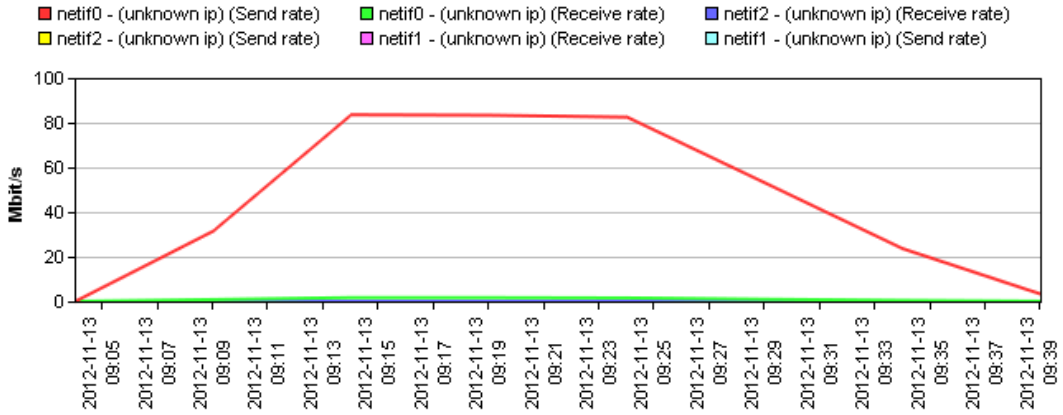
Throughout the stats Minimum, Average and Maximum times in seconds were provided as follows:

Description	Min (Secs)	Average (Secs)	Max (Secs)
CREATE	0.193	2.106	28.06
UNLINK	0.031	0.217	4.431
DELETE	0.262	0.804	8.032
ADD_ATTR	1.007	3.827	20.445
SEARCH	0.267	7.25	64.158

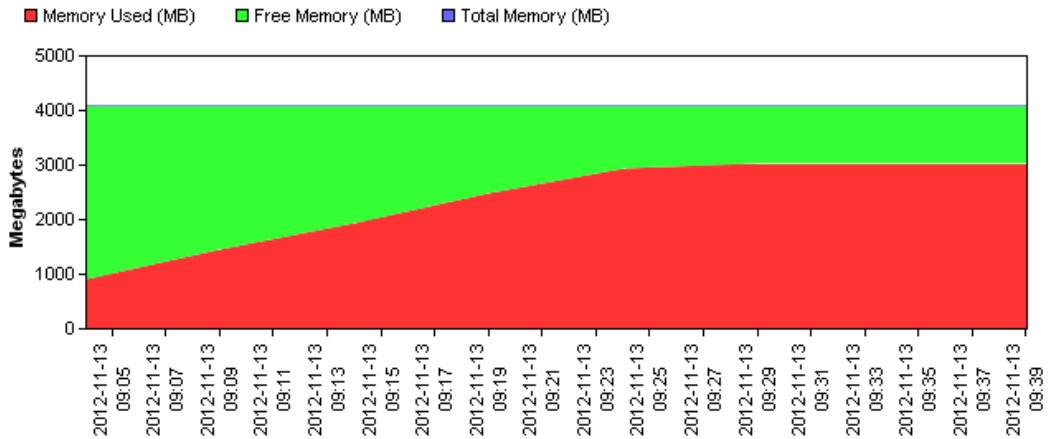
The Environment statistics were as follows:



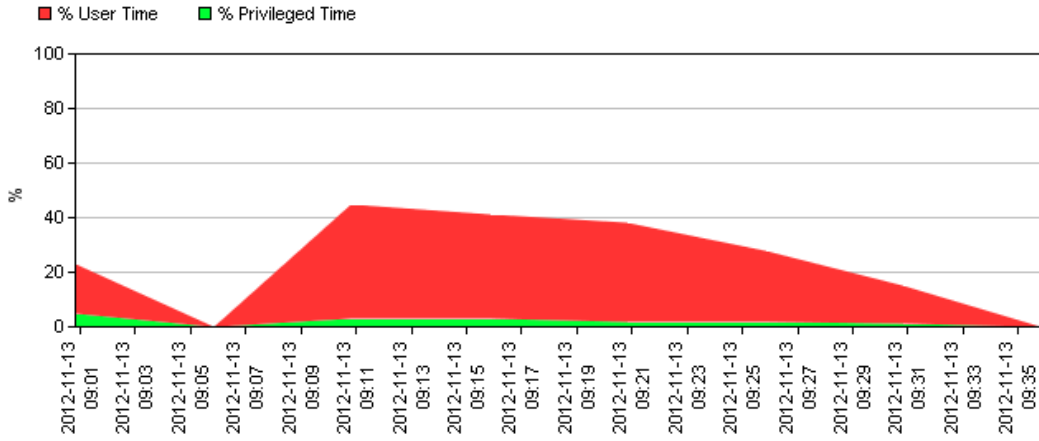
**Network Usage
SQL Back End**



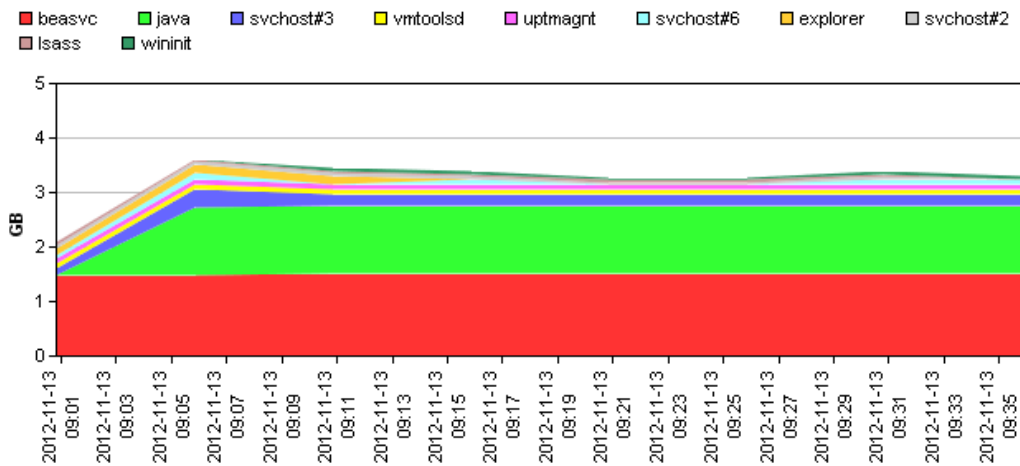
**Free Memory
SQL Back End**



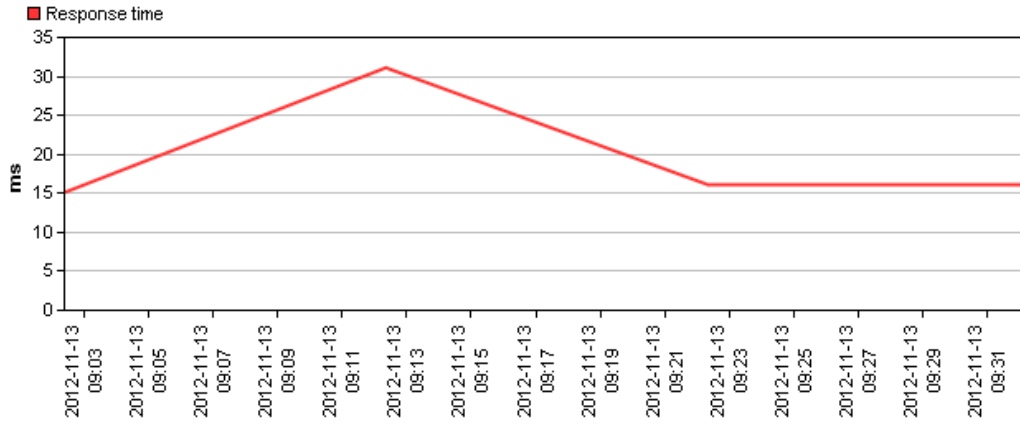
CPU Usage
Weblogic Front End



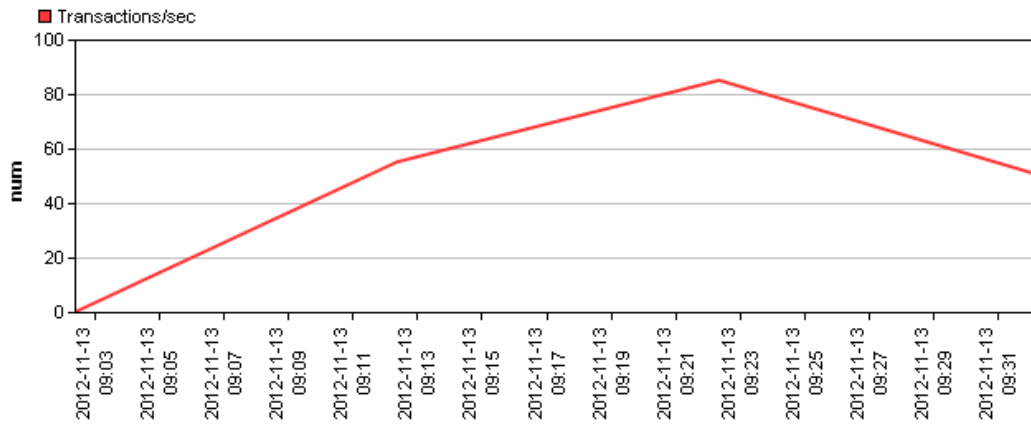
Memory Size Workload Characterization
Weblogic Front End



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Response time



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Transactions/sec



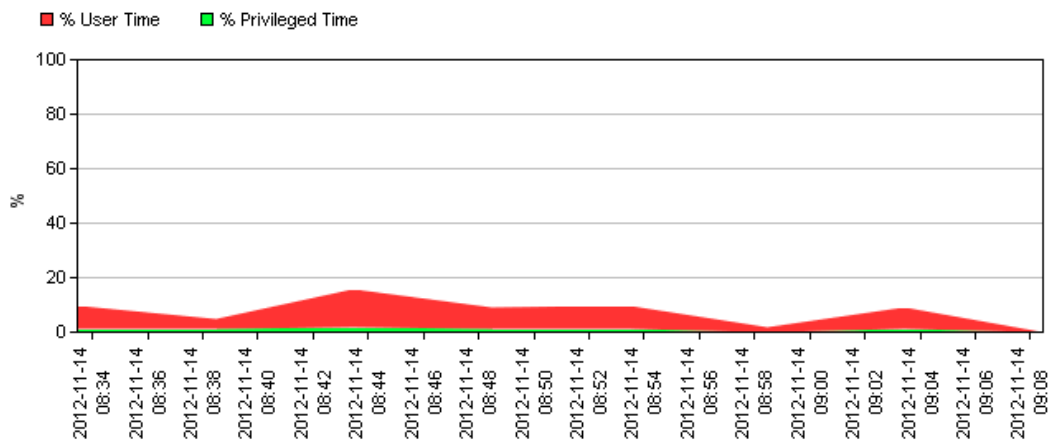
11.4 Test 4

The end user timings were as follows:

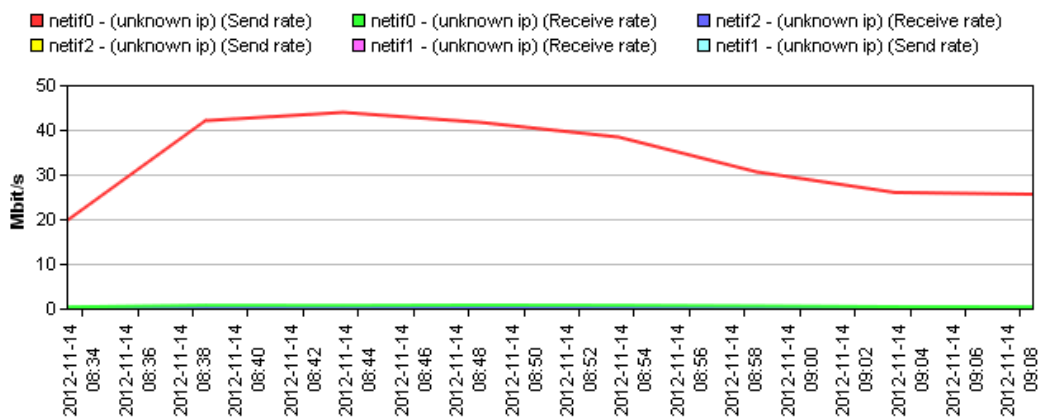
Throughout the stats Minimum, Average and Maximum times in seconds were provided as follows:

Description	Min (Secs)	Average (Secs)	Max (Secs)
CREATE	0.194	0.898	5.73
UNLINK	0.028	0.241	1.597
DELETE	0.293	0.902	3.649
ADD_ATTR	0.105	1.325	8.976
SEARCH	0.131	6.811	39.199

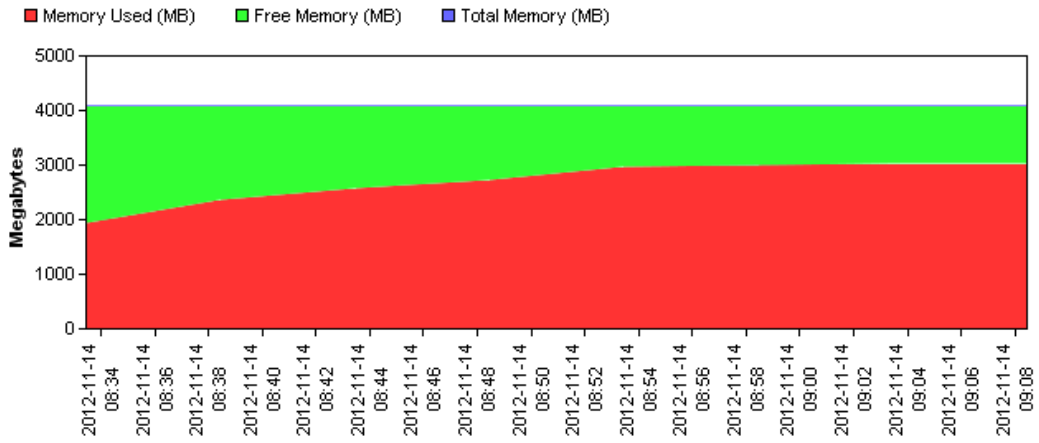
**CPU Usage
SQL Back End**



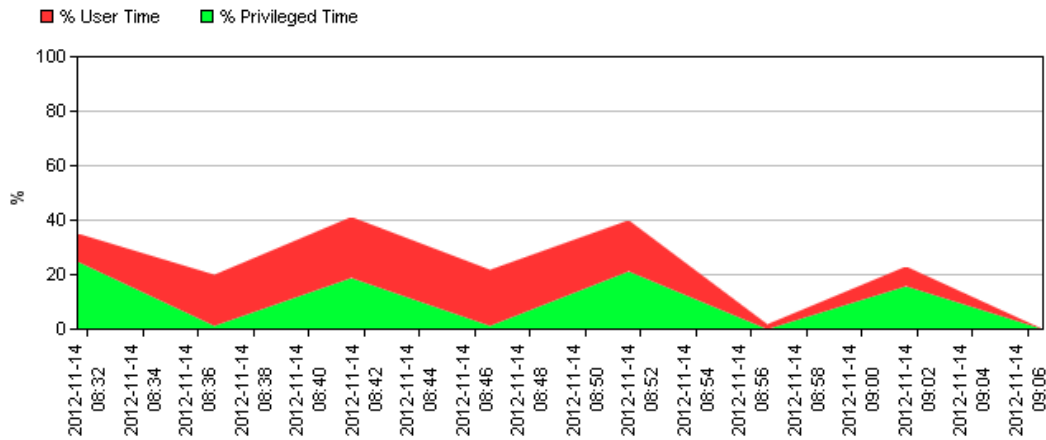
**Network Usage
SQL Back End**



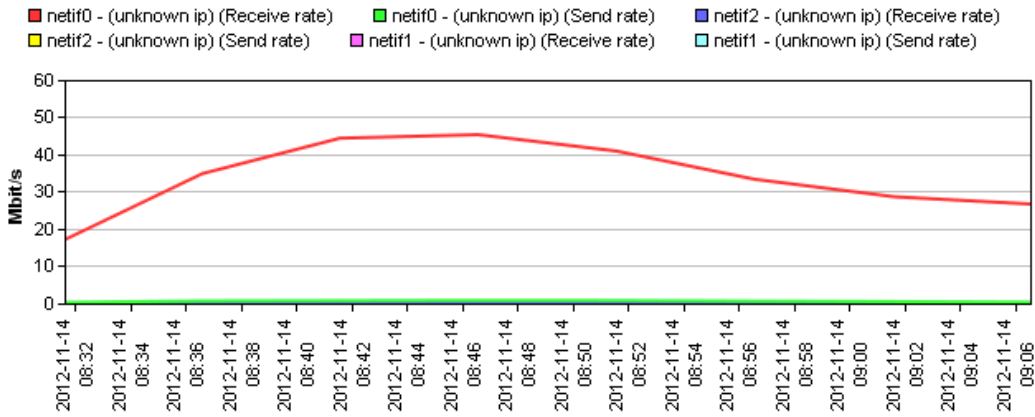
**Free Memory
SQL Back End**



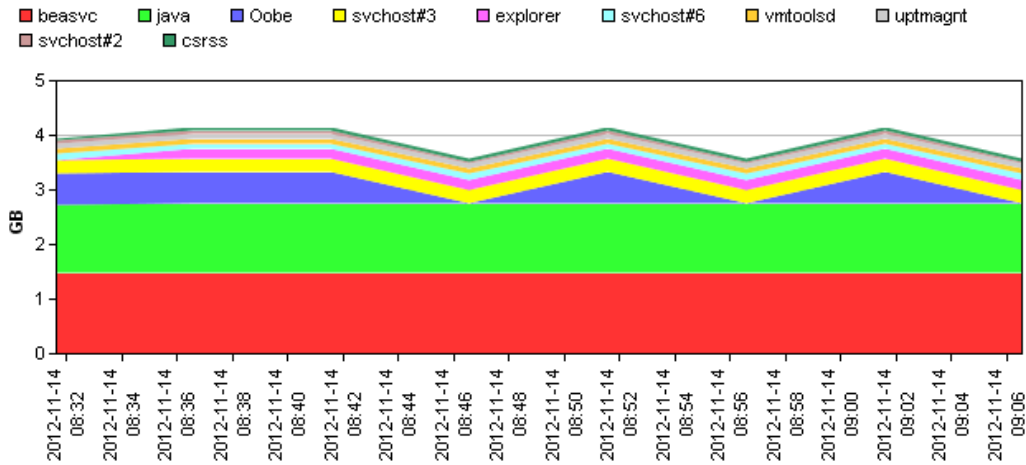
**CPU Usage
Weblogic Front End**



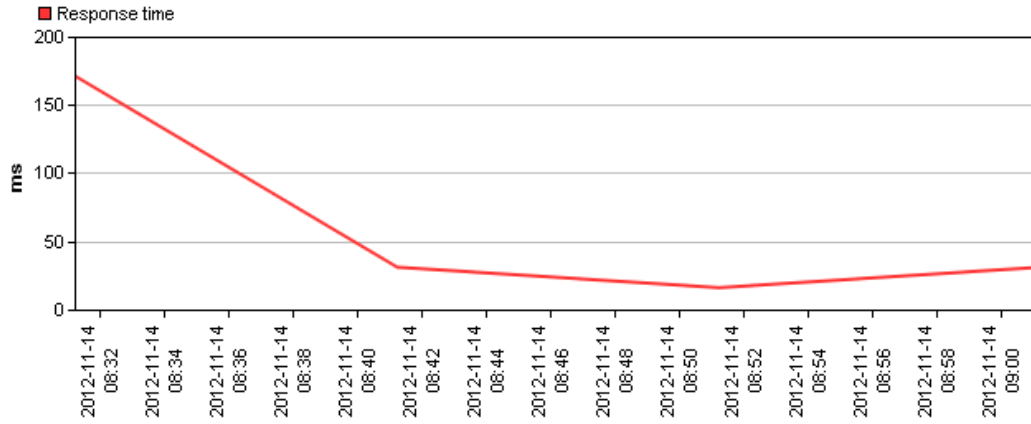
Network Usage
Weblogic Front End



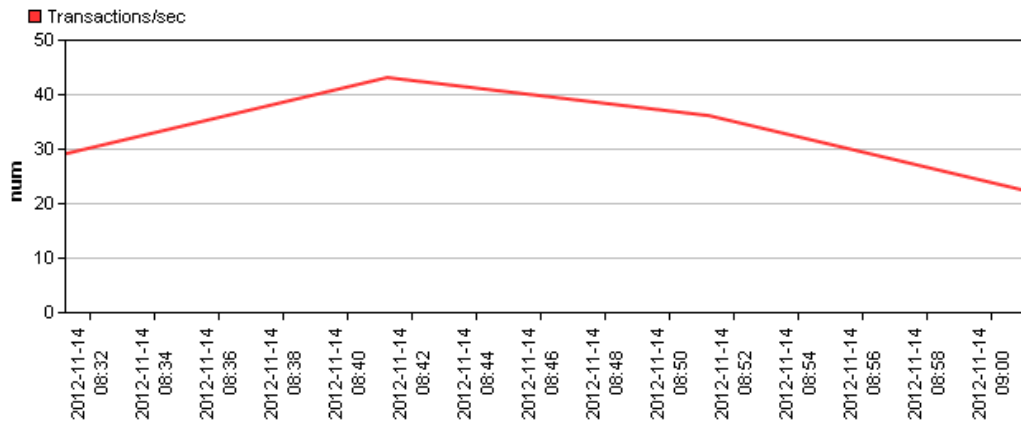
Memory Size Workload Characterization
Weblogic Front End



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Response time



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Transactions/sec



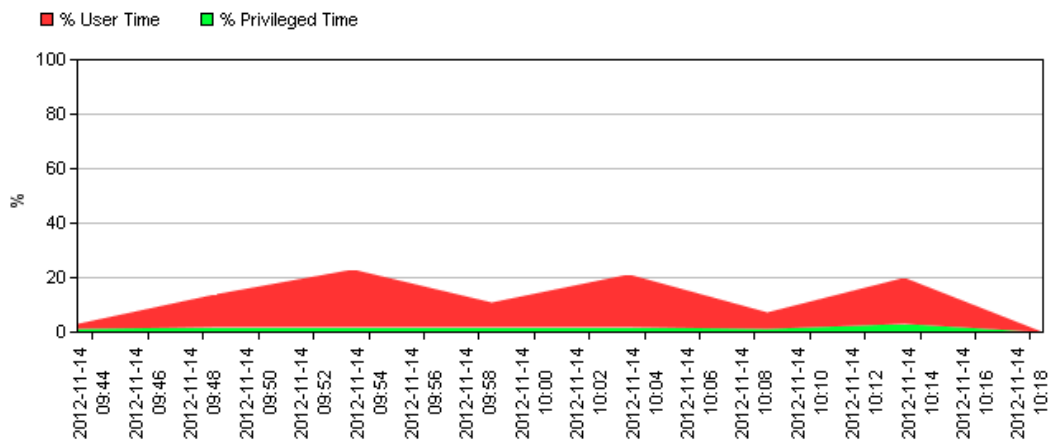
11.5 Test 5

The end user timings were as follows:

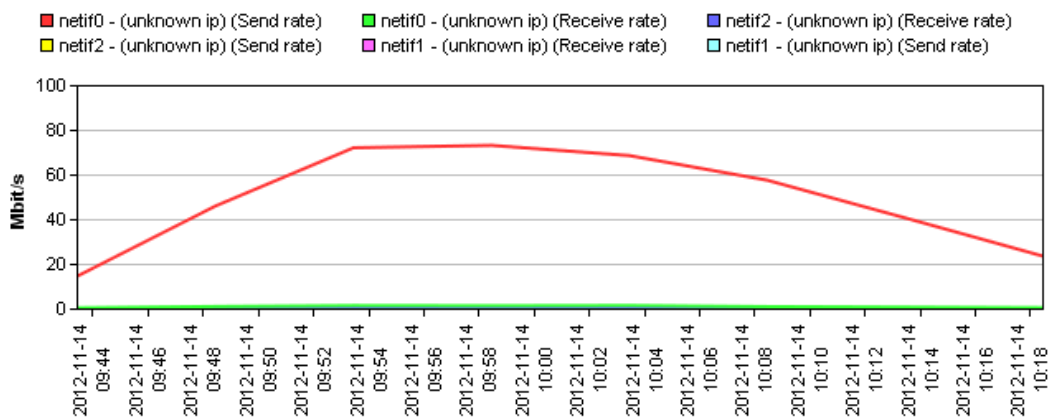
Throughout the stats Minimum, Average and Maximum times in seconds were provided as follows:

Description	Min (Secs)	Average (Secs)	Max (Secs)
CREATE	0.2	2.047	23.723
UNLINK	0.028	0.499	7.027
DELETE	0.32	1.12	10.007
ADD_ATTR	0.1	3.546	22.369
SEARCH	0.192	6.707	52.607

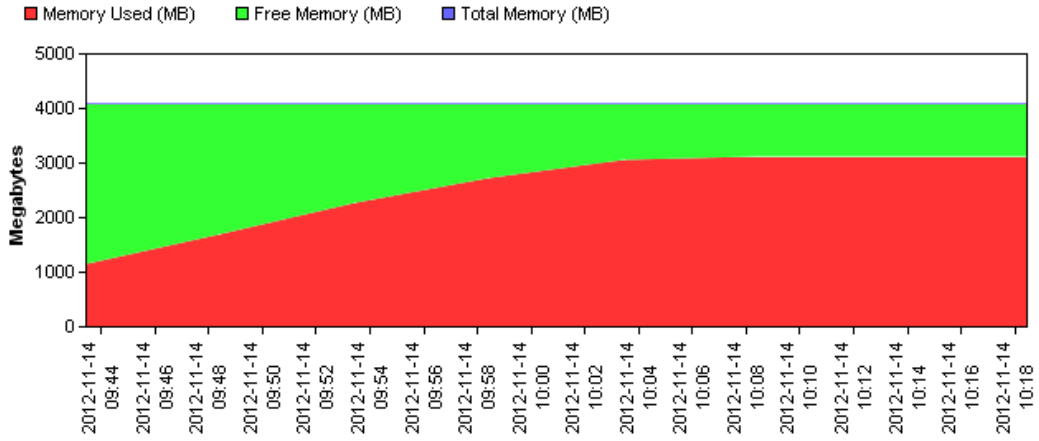
**CPU Usage
SQL Back End**



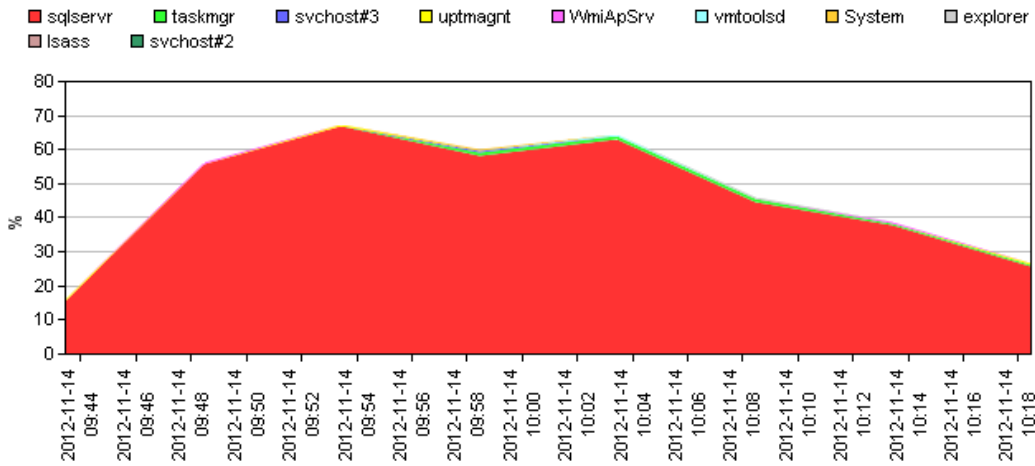
**Network Usage
SQL Back End**



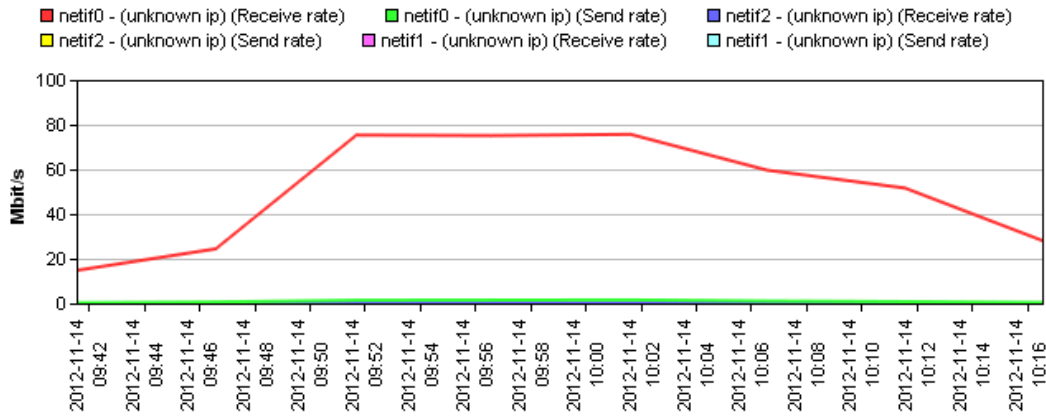
**Free Memory
SQL Back End**



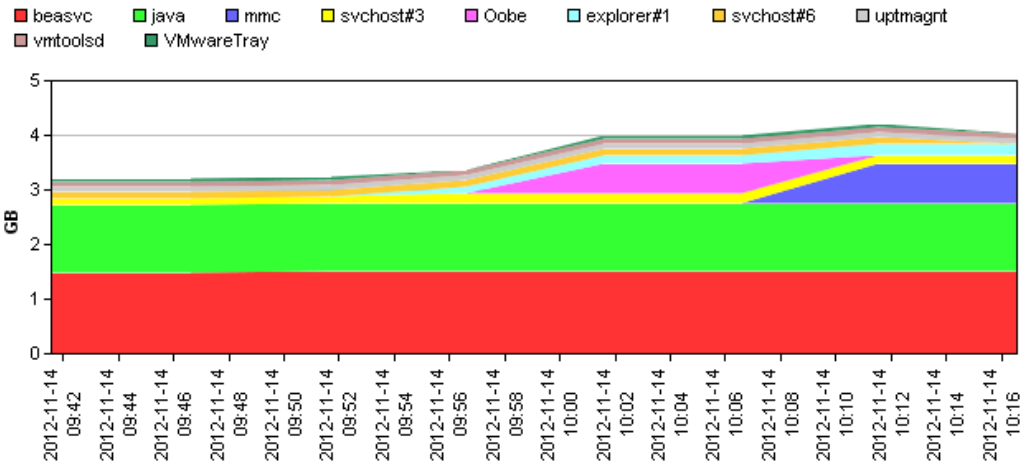
**CPU Workload Characterization
SQL Back End**



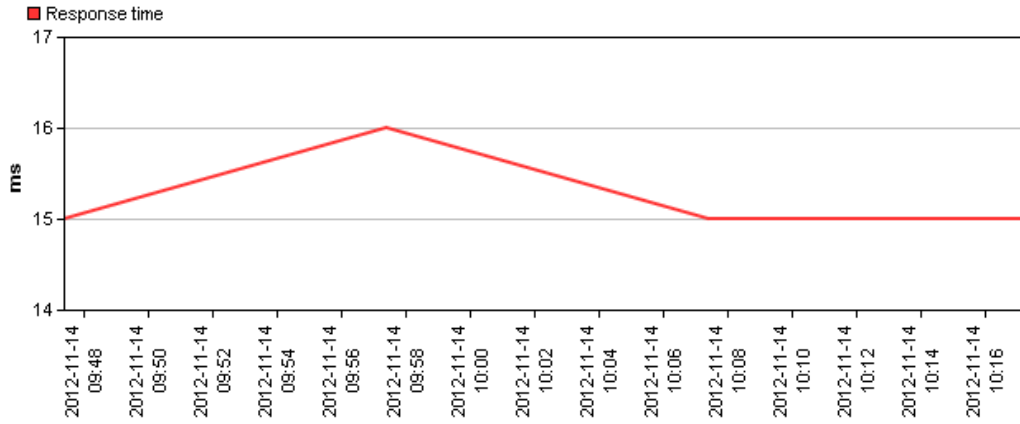
Network Usage
Weblogic Front End



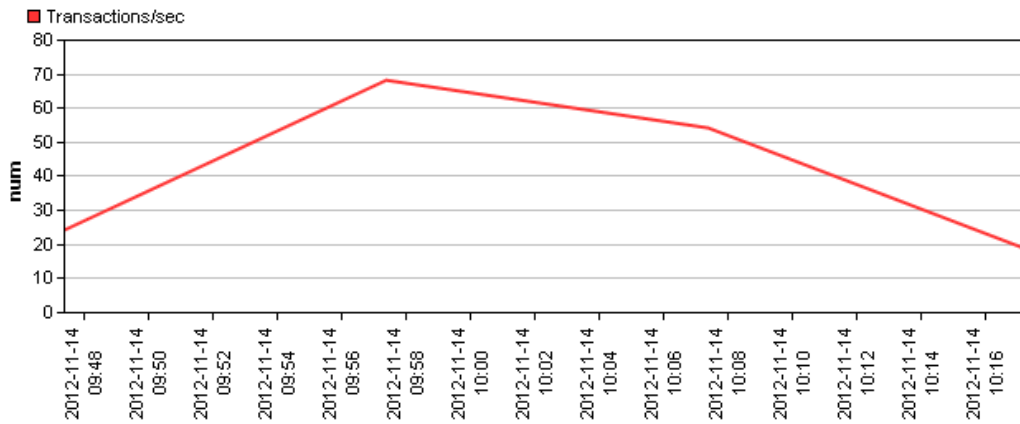
Memory Size Workload Characterization
Weblogic Front End



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Response time



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Transactions/sec



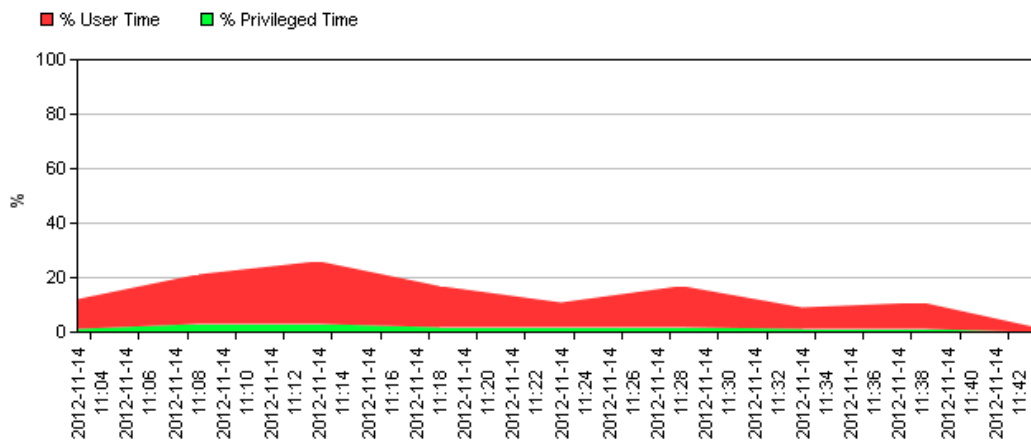
11.6 Test 6

The end user timings were as follows:

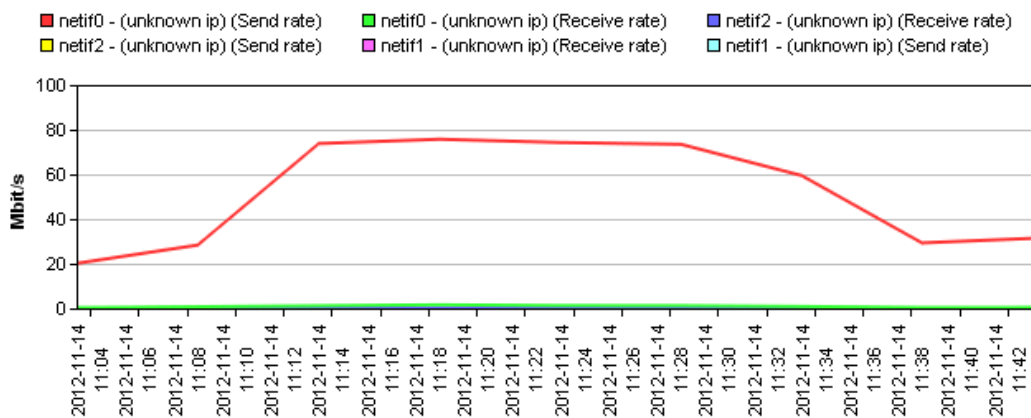
Throughout the stats Minimum, Average and Maximum times in seconds were provided as follows:

Description	Min (Secs)	Average (Secs)	Max (Secs)
CREATE	0.192	3.035	30.225
UNLINK	0.0284	1.17	23.01
DELETE	0.319	2.036	23.424
ADD_ATTR	0.1	7.456	50.61
SEARCH	0.182	11.067	73.92

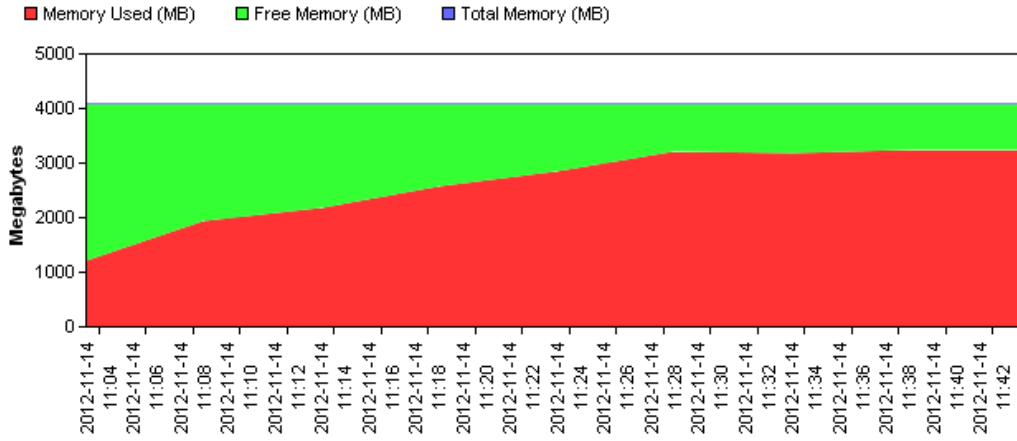
**CPU Usage
SQL Back End**



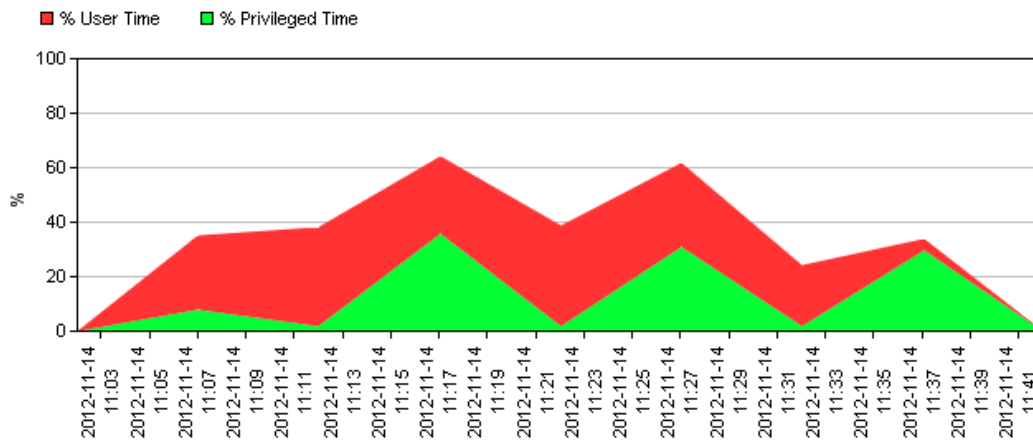
**Network Usage
SQL Back End**



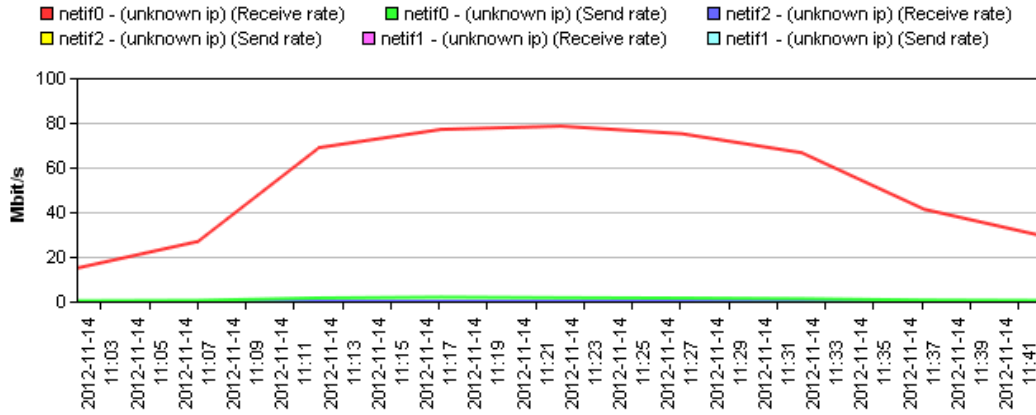
**Free Memory
SQL Back End**



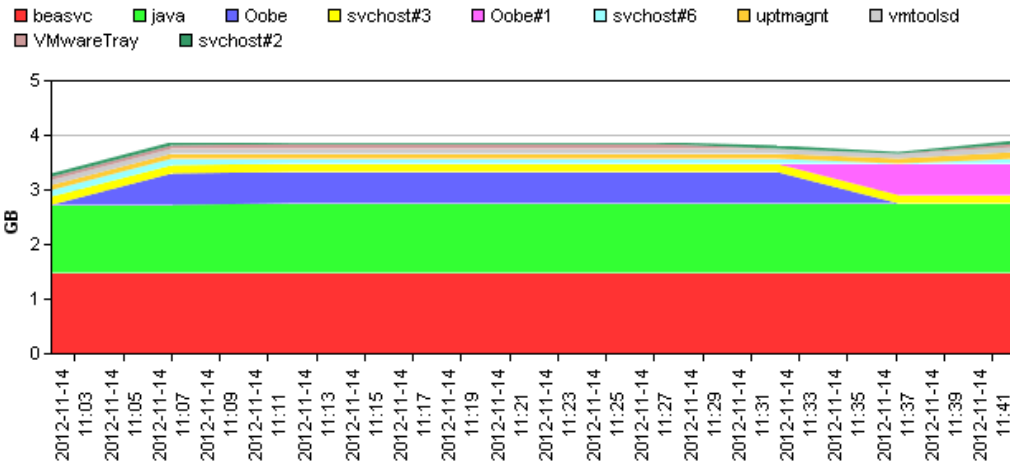
**CPU Usage
Weblogic Front End**



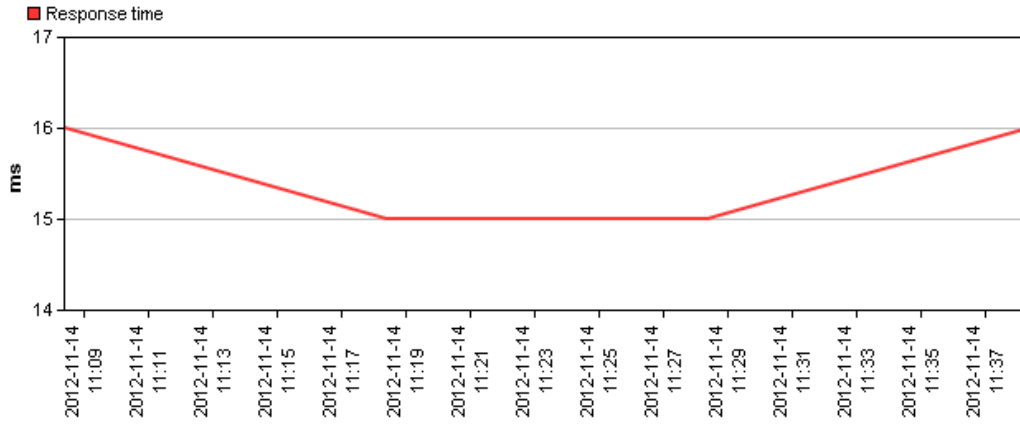
Network Usage
Weblogic Front End



Memory Size Workload Characterization
Weblogic Front End



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Response time



SQL Back End (172.24.4.171)
SQL Service Monitoring - Advanced - Transactions/sec

