



white paper

Benchmark Results: Scalability of Alfresco Content Management System in a Unisys ES7000/one Enterprise Server Environment

results

Table of Contents

Executive Summary	4
Overview	4
Requirements Definition	4
Performance	4
Scope	4
Assumptions	4
Goals	5
Metrics	5
Baseline Establishment	5
Key Metrics	5
Other Measurements	5
Hardware Required	5
System Software Required	5
Application Software Required	5
Application Overview	6
Test Methodology	7
Methods	7
Benchmark Business Process 1 – Load 10 Million Documents	7
Benchmark Business Process 2 - Load 100 Million Documents	7
Benchmark Business Process 3 – Demonstrate Adequate Response	7
Benchmark Business Process 4 - Test Reindex Performance (optional)	8
Benchmark Business Process 5 – Keep Loading	8
Results	8
Key Metrics	8
Other Measurements	8
Discussion	9
Achievement of Goals	9
Observations	9
Performance Analysis	9
System Statistics	9
JVM	9
Conclusions	10
Next Steps	10

Test System Configuration	10
Hardware Required	10
Database/Application Tier	10
Driver Systems	10
Storage	10
Software Required	11
Database Tier	11
Application Tier	11
Driver Systems	11
Network Required	11
Configuration Diagram	11
Storage Configuration	12
LUN Assignments	12
Meta-LUN Assignments	12
Storage Groups	13
Server Configuration	13
ES7000 Partition Assignments	13
Commodity Server Assignment	14
IP Address Assignments	14
Appendix - Test Documents used for load and retrieval	15

Executive Summary

In October 2007, Unisys provided Alfresco with a Unisys ES7000/one Enterprise Server, EMC2 CX3-80 storage Subsystem, and related equipment to benchmark the scalability of Alfresco's Content Management System (CMS) on enterprise-class equipment.

The goals of this benchmark were to load at least 100 million documents into a single repository, achieve a load rate of at least 100 documents loaded per second, and demonstrate acceptable response time for single-document retrieval with a repository this large.

All three of the goals were met. A total of 107 million documents were loaded into the repository at a rate of 140 documents per second. After some minor fixes to the Alfresco code, all document retrieval operations were demonstrated to take the same amount of time as in a clean system.

During the test, the ES7000/one Enterprise Server was underutilized – processor idle time was approximately 80 percent. Since the goals were met “out of the box” and due to the short testing timeframe, no attempts were made to optimize the application for this scale hardware. It is recommended that a follow-on test be scheduled where this can be addressed. The goal of such a test would be to fully utilize the ES7000/one Server and maximize the documents loaded per second.

Overview

Alfresco is the leading Open Source alternative for enterprise content management. It couples the innovation of Open Source with the stability of a true enterprise-class platform. The Open Source model allows Alfresco to use best-of-breed Open Source technologies and contributions from the Open Source community to get higher quality software produced more quickly at much lower cost.

Requirements Definition

- Load 100 million documents into a single Alfresco repository
- Demonstrate adequate response times for various document-retrieval operations
- Demonstrate scaling using clustered servers (if necessary due to ceiling on the scale-up architecture)

Performance

Demonstrating load performance was a secondary requirement to demonstrating the ability to support large-scale storage. However, the latter is not possible without the former. Read performance is critical only in that it must be “acceptable” (acceptable response times are defined in the section Test Methodology).

Scope

This test plan addresses the validation and performance characterization using the existing application release only. Only the tests described in the Requirements Definition were covered in this test. Alfresco has shown interest in loading one billion documents into the repository, however, at the rate of 140 documents per second that was achieved during the test, this would take a continuous run of approximately three months. The EPC will be happy to work with Alfresco to achieve this goal in a future engagement. We believe that working to increase the document processing rate would bring the total run time down to a more reasonable number.

Assumptions

- Successful installation of RHEL 4 U4 and associated components
- Successful installation and performance of the Alfresco application, which is a development version and is different from the version used in the previous benchmark

Goals

1. Load as many documents as possible in the time allocated, theoretically 100 million documents
2. Achieve a load rate of 100 documents per second
3. Document single-document retrieval time using various levels of concurrency

Metrics

Baseline Establishment

The baseline for this test was established during a previous benchmark in 2006 using Alfresco version 1.4. This benchmark was conducted using a dual-processor, dual-core 2.6 MHz Opteron-based system with 12 GB memory. The following results were obtained:

- Documents loaded: 10 million
- Documents loaded per second – 64
- Read content – 0.34961 seconds
- Read property – 0.41976 seconds
- Create content – 0.68788 seconds (not critical now)
- Create folder – 0.54419 seconds (not critical)

In so far as this is considered our “baseline,” the goal of the current benchmark is to determine how much the previous results can be exceeded using Alfresco version 2.2 (still under development) running on an enterprise-class infrastructure. The primary goal of the current test is to exceed this previous benchmark by a factor of 10.

The need to demonstrate scaling by way of using clustered servers was not required.

Key Metrics

- Total number of documents stored
- Documents loaded per second
- Response time for document operations
- Processor utilization

Other Measurements

- Context switches
- Memory utilization
- Disk statistics (queue length, response time)
- Size of Lucene indexes
- Size of database

Hardware Required

- ES7000/one Server as a database/application server
- EMC2 CX3-80 storage subsystem for database, indexes, and image files
- 2.7 GHz commodity four-processor servers for driver systems (quantity TBD)

System Software Required

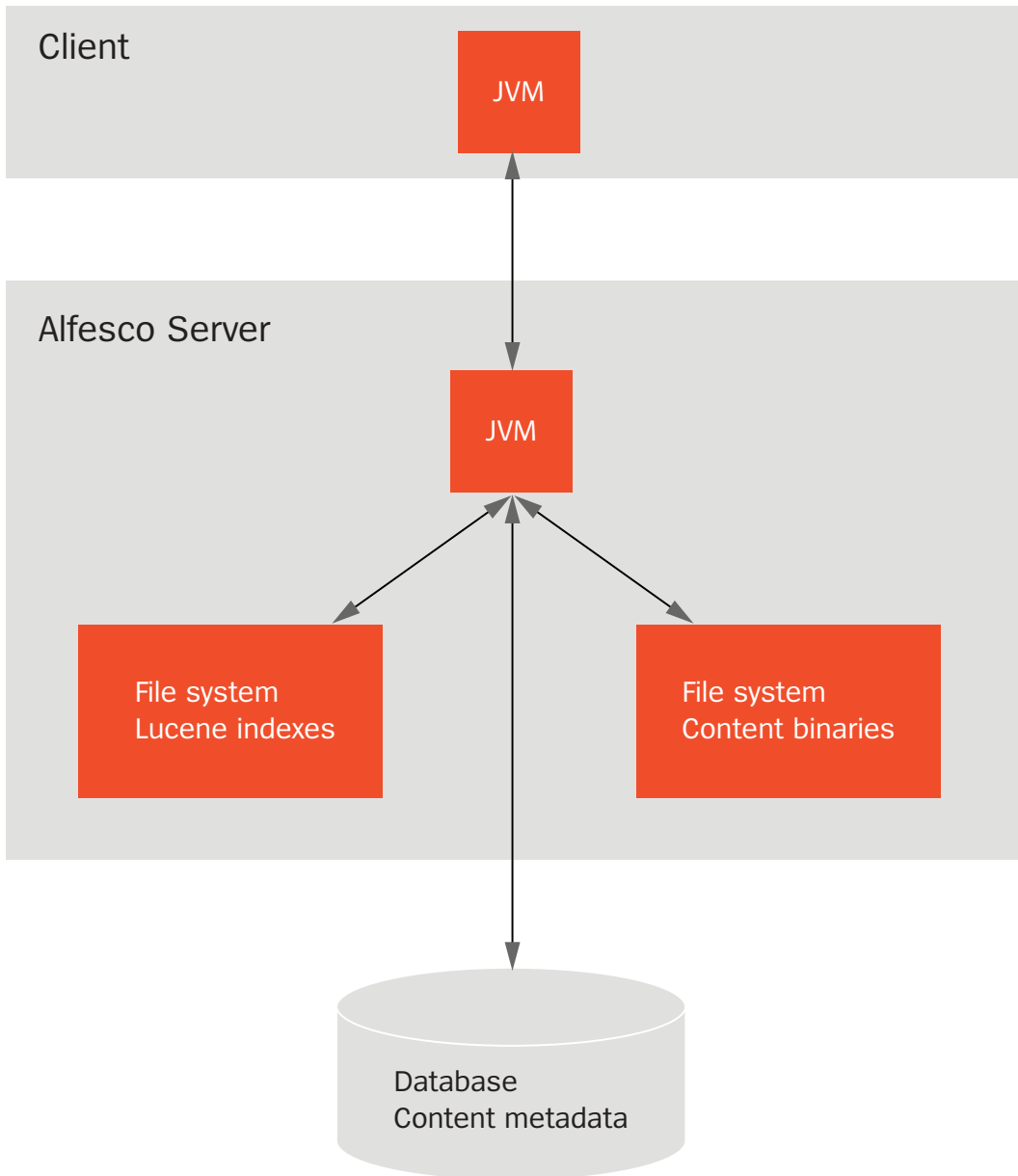
- Red Hat Enterprise Linux, RHEL 4 Update 4
- MySQL 4.1.21 Enterprise (or Oracle 10g if needed)
- JDBC driver for MySQL 4.1.21 (or Oracle 10g if needed due to performance issues with MySQL)
- Unisys JVM 5.0.10 (or Sun JVM 6.0 if needed due to performance issues with Java SE 5.0- based JVM)
- Firefox web browser
- FTP client
- Apache Tomcat 5.5.20

Application Software Required

- Alfresco 2.2 (development version)

Application Overview

The following diagram shows the architectural overview of the application.



The Alfresco Server can either run an application server such as Tomcat, or as a standalone application. Tomcat was used in order to simplify the installation and configuration of the server. This also allowed the repository to be accessed via the user interface, and in particular via the Node Browser web application.

The client loader program communicates with the Alfresco server via RMI.

The document structure and metadata is stored in the database while the actual content binaries are stored on a file system that is locally accessible to the server VM. Indexes of the document structure and documents' text are maintained on the file system in the form of Lucene indexes.

Test Methodology

Methods

Most clients of the Alfresco repository (*the Repository*) use a common API for managing documents. The API will be exported via RMI and made available to a remote benchmark client (*the Benchmark Application, BMLoader*), written specifically for these tests.

The document structure will be as follows:

- 10 stores (partitionable units)
- Each store has 10 million documents spread across 10 thousand folders
- Trimmed-down configuration settings to remove unused functionality
- A list of the documents used for all of the tests is provided in the Appendix

The Benchmark Loader or BMLoader is an application written by Alfresco for use in benchmarks. The BMLoader simulates a production environment and input to the Alfresco repository without the necessity of installing the external third-party applications that interface with the Alfresco. It will be responsible for creating the document structure. For additional information on this process, please refer to the Alfresco website at: http://wiki.alfresco.com/wiki/Server_Benchmarks.

Performance metrics will be recorded by both the BMLoader and by a separate process running within the Repository itself.

Client responsiveness can be measured using an FTP client and performing common operations such as navigation and uploading and downloading of files. Sub-second client response times are adequate for these tests to be considered successful.

Benchmark Business Process 1 – Load 10 Million Documents

The first task will be to load 10 million documents into each of the logical stores. Analysis of performance and hardware utilization during the phase will take place. Hardware configurations and their effects will be measured in order to:

- Demonstrate scaling as concurrency is increased
- Demonstrate scaling as system resources are added
- Demonstrate scaling as the server is clustered. This only needs to be done if further performance improvements are *required* or if there is enough time available in this phase.

Records of the resource usage must be kept. In particular, the size of the Lucene indexes and databases.

From this, the best available configuration can be set up. The predicted outcome of the next phase will be more certain, which will be to get up to 100 million documents.

Benchmark Business Process 2 - Load 100 Million Documents

With the available hardware and software optimized, the next phase will be to ramp up to the first target milestone. 100 million documents in the repository is the target number of documents that will be considered a success. The process will be automated, with the BMLoader and Repository working 24 hours a day until the target is reached. Intervention will only be required in the event of unexpected problems or a significant reduction in the document load rate.

Records of the resource usage must be kept. In particular, the size of the Lucene indexes and databases.

Benchmark Business Process 3 – Demonstrate Adequate Response

The BMLoader application will perform a set of read and write operations against the Repository. Since the BMLoader will be using a fully functional API shared by the other Alfresco clients, response times can be considered similar to those that will be achieved by the clients themselves.

Tests using FTP and the Web Client will also be done.

Although results may be measured, they need only satisfy the general *user adequacy* requirement.

Benchmark Business Process 4 - Test Reindex Performance (optional)

This optional phase would be useful, but should only be done if:

- the Lucene indexes can be safely moved away and brought back later (backed up and restored)
- there is time available

The aim of this phase is to

- measure the re-index speed, which will be a measure of the overall read performance
- see if the size of the Lucene indexes differs significantly from the incrementally built indexes

Benchmark Business Process 5 – Keep Loading

This is just like phase 2, but continues until the last possible moment. If further hardware is required and available, this can be added in. Enough time must just be available to repeat phase 3.

Results

Key Metrics

- Total number of documents stored
 - **107 million**
- Documents loaded per second
 - **140 documents per second**
- Response time for document operations
 - **Acceptable (see Discussion)**
- Processor utilization
 - **average of approximately 20 percent on the application server and 15 percent on the database server**

Other Measurements

- Context switches
 - **Not recorded**
- Memory utilization
 - **Application server – 30 GB (JVM)**
- Disk statistics (queue length, response time)
 - **The only disk with a high queue depth was the content store**
- Size of Lucene indexes
 - **4 GB each for each of the 10 stores**
- Size of database
 - **~800 GB**

Discussion

Achievement of Goals

All goals of this test plan were met. 107 million documents were loaded (target was 100 million). The average load rate was 140 documents per second, 40 percent higher than the target of 100 documents per second and more than double what was achieved during previous tests. The response times for the read and write operations against the fully loaded store were less than one second which is more than adequate.

Observations

1. A problem was detected during server startup. As part of the consistency checks, a “select max(id)” query was performed which was observed to take up to 10 minutes to execute. The check was switched off and Alfresco has corrected the issue.
2. The UI response times for most operations were not noticeably different from the clean system. Browsing was fast and creation and modification of folders showed no sign of being impaired by the data volume or even by having a benchmark run concurrently. Content upload speeds were good, but slightly affected by running a concurrent benchmark.
3. Deletion speeds were affected by the data volume. A particular query that had to get executed N+1 times (where N is the depth of the hierarchy being deleted) was taking minutes to execute. This rendered delete unusable. Alfresco has fixed the query.
4. The Alfresco Node Browser was used to perform searches against both the standard Spaces Store and the benchmark partitioned stores. The observed results were normal. In other words, the usual large resultset limitations that are currently present in the system were not exacerbated by the data volumes. The Lucene indexes showed no signs of degradation at volume.
5. The FTP client displayed the same good response as the Web Client, with the exception of deletes as mentioned above.

6. A full reindex was kicked off to see how long it would take. The single process took approximately 24 hours to reindex 10 percent of the complete set. Alfresco is addressing this.
7. Full scalability for this test was achieved on a single application server, therefore a clustered environment was not necessary.

Performance Analysis

System Statistics

Processor utilization was low (~20 percent) on both the database (four CPU) and application (eight CPU) servers. The JVM was allowed to use all of the available memory but probably did not need all of that to produce these levels of performance. The only disk that showed significant queuing was sdl on the application server which contained the content store.

JVM

Testing was begun with the Sun JVM, v1.6 (aka 6.0). During the test, the Unisys JVM was swapped for the Sun JVM and a decline in performance was noted. It was determined that this was due to the Unisys JVM being based on the Sun JVM v1.5 (Java SE 5.0). The Alfresco team had noticed a significant improvement in going from Sun v1.5 to Sun v1.6. Unisys has a development version based on v1.6 which was tried later in the test. It appeared to provide roughly the same level of performance as was obtained with the Sun JVM.

Conclusions

The test goals presented in this test plan were all met with very little tuning. The goals were aimed at determining whether the Alfresco system could provide acceptable performance with a large content store (greater than 100 million documents) rather than optimizing either the document load speed or retrieval times. The ES7000/one Server and related test components were vastly underutilized (with the exception of the content store disk). Because of the test goals, the limited time available during this test, and the fact that acceptable performance was achieved essentially “out of the box,” minimal optimizations were made and minimal performance statistics were captured. The Unisys EPC team recommended that a future engagement be scheduled which will focus on tuning the document load process. One of the desired goals of Alfresco is to test with one billion documents in the content store. This would only be feasible with a significant increase to the document load speed. At the observed rate of 140 documents per second, the system would need to run continuously for 83 days in order to achieve one billion documents.

Next Steps

First and foremost, all of the fixes discovered during this test will be incorporated into the 2.2 release of the Alfresco Repository. They will also be included in the community code base as well.

Second, Alfresco and Unisys should seriously consider a second set of tests to be run sometime in early 2008 to address actual performance tuning on the ES7000/one Server. From the metrics observed during this test, it is quite feasible that the rate of document insertion into the repository can be significantly increased. Along with faster processing times, the increase in load speed would make it feasible to aim for a higher total number of documents inserted into the repository, making Alfresco and Unisys the leader in document management.

Test System Configuration

Hardware Required

Database/Application Tier

ES7000/one Server with eight cells, each cell containing:

- Four processors, Intel® Dual-Core Xeon® 7140M, 3.33 GHz with 16 MB L3-cache
- 32 GB memory
- Two LP11002-M4 4 Gb dual-port fibre-channel HBAs
- Two Intel® PRO/1000 MT dual-port gigabit NICs
- Two 146 GB internal disks, SCSI 320, mirrored

Driver Systems

Unisys ES3040L Servers, four required, each containing:

- Four Intel® Xeon® MP processors, 2.7GHz with 2MB cache
- Eight GB memory
- Two Broadcom NetExtreme gigabit NICs
- Two 36 GB internal disks, mirrored

Storage

One EMC2 CX3-80 Clariion Storage subsystem

100M documents, 50K average size, 5000 GB for content storage

- 16 GB shared cache
- 8 4-Gb fibre-channel host ports
- 120 73 GB, 15K RPM, 4-Gb fibre-channel disks
- 60 146 GB, 15K RPM, 4-Gb fibre-channel disks

Software Required

Database Tier

- Red Hat Enterprise Linux AS 4 Update 4 for AMD64/Intel64; or
- MySQL 4.1.21 Enterprise (or Oracle 10G, Enterprise Edition)

Application Tier

- Red Hat Enterprise Linux AS 4 Update 4 for AMD64/Intel64
- JDBC driver for MySQL 4.1.21 or Oracle 10g
- Unisys JVM 5.0.10 x64 (or Sun JVM 6.0 x64)
- Alfresco 2.2 (development version) (supplied by client)

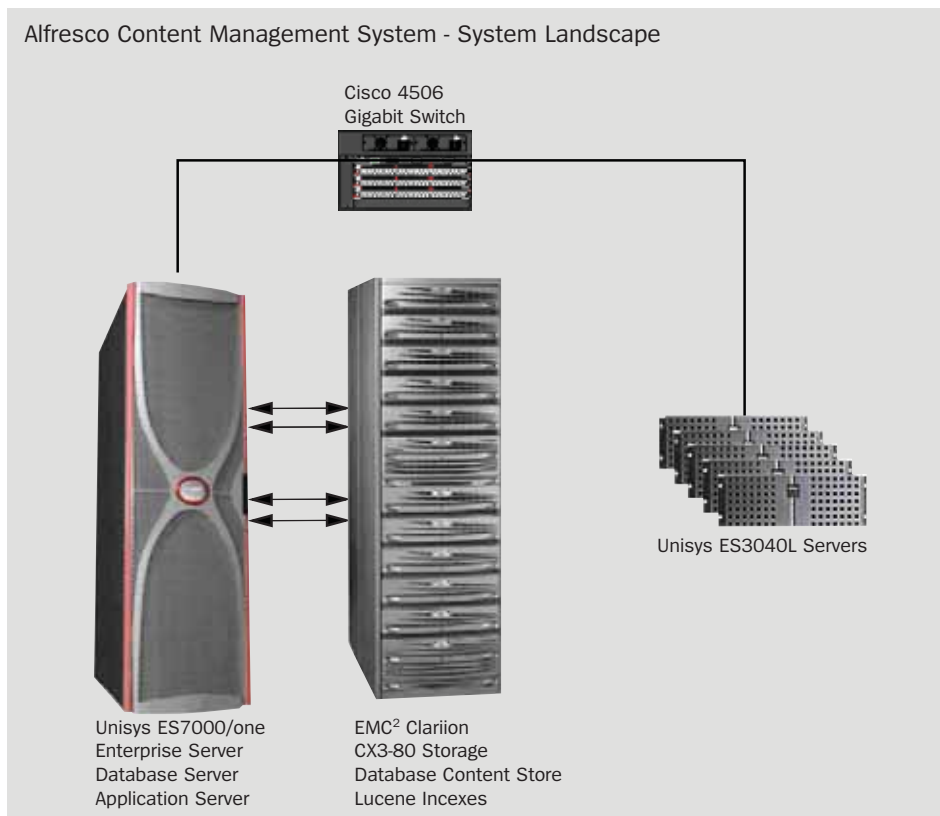
Driver Systems

- Red Hat Enterprise Linux AS 4 Update 4
- Firefox
- FTP client
- Unisys JVM 5.0.10 i386

Network Required

- Public LAN for communications between servers and remote administration

Configuration Diagram



Storage Configuration

LUN Assignments

Array	Name	LUN ID	RAID Type	Number of Drives	Physical Drive Size (GB)	LUN Size (GB)	Storage Processor	Meta-LUN Component
1	LUN 0	6138	5	8	133	936	A	Yes
1	LUN 1	6143	5	8	133	936	A	Yes
1	LUN 2	6142	5	8	133	936	A	Yes
1	LUN 3	6141	5	8	133	936	A	Yes
1	LUN 4	6140	5	8	133	936	A	Yes
1	LUN 5	6139	5	8	133	936	A	Yes
1	LUN 1	1	1+0	10	67	333	B	No
1	LUN 2	2	1+0	10	67	333	A	No
1	LUN 3	3	1+0	10	67	333	B	No
1	LUN 4	4	1+0	10	67	333	A	No
1	LUN 5	5	1+0	10	67	333	B	No
1	LUN 6	6	1+0	10	67	333	A	No
1	LUN 7	7	1+0	10	67	333	B	No
1	LUN 8	8	1+0	10	67	333	A	No
1	LUN 9	9	1+0	10	67	333	B	No
1	LUN 10	10	1+0	10	67	333	A	No
1	LUN 11	11	1+0	10	67	333	B	No

Meta-LUN Assignments

Array	Name	LUN ID	Type	LUNs (ID)	Meta-LUN Size (GB)	Storage Processor
1	LUN 0	0	Stripe	6138, 6143, 6142, 6141, 6140, 6139	5.5TB	A

Storage Groups

Array	Storage Group	LUN (ID)	Host	Host Drive	Name/Function
1	Alfresco DB	1	usmv-epce2p0	sdb	MySQL Database
1	Alfresco App	0	usmv-epce2p4	sdb	Lucene Index
1	Alfresco App	2	usmv-epce2p4	Sdc	Lucene Index
1	Alfresco App	3	usmv-epce2p4	Sdd	Lucene Index
1	Alfresco App	4	usmv-epce2p4	Sde	Lucene Index
1	Alfresco App	5	usmv-epce2p4	Sdf	Lucene Index
1	Alfresco App	6	usmv-epce2p4	Sdg	Lucene Index
1	Alfresco App	7	usmv-epce2p4	Sdh	Lucene Index
1	Alfresco App	8	usmv-epce2p4	Sdi	Lucene Index
1	Alfresco App	9	usmv-epce2p4	Sdj	Lucene Index
1	Alfresco App	10	usmv-epce2p4	sdk	Lucene Index
1	Alfresco App	11	usmv-epce2p4	sdl	Content Store

Server Configuration

ES7000 Partition Assignments

Name	CPUs	CPU Speed	Cache (MB)	Memory (GB)	NICs	FC Adapters	Function
usmv-epce2p0	4	3.4	16	32	4	4	Database server
usmv-epce2p4	4	3.4	16	32	4	4	Application server

Commodity Server Assignments

Name	CPUs	CPU Speed	Cache (MB)	Memory (GB)	NICs	Storage Size (GB)	Storage Type	Function
usmv-epcc2-1	4	2.7	2	8	2	36	RAID	Client
usmv-epcc2-2	4	2.7	2	8	2	36	RAID	Not used
usmv-epcc2-3	4	2.7	2	8	2	36	RAID	Not used
usmv-epcc2-4	4	2.7	2	8	2	36	RAID	Not used
usmv-epcc2-5	4	2.7	2	8	2	36	RAID	Not used
usmv-epcc2-6	4	2.7	2	8	2	36	RAID	Not used
usmv-epcc2-7	4	2.7	2	8	2	36	RAID	Not used
usmv-epcc2-8	4	2.7	2	8	2	36	RAID	Not used
usmv-epcc2-9	4	2.7	2	8	2	36	RAID	Not used
usmv-epcc2-10	4	2.7	2	8	2	36	RAID	Not used

IP Address Assignments

Name	IP Address (Primary)	IP Address (Secondary)	Subnet Mask	DNS Server (Primary)	DNS Server (Secondary)
usmv-epce2p0	172.26.102.0	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epce2p4	172.26.102.4	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epcc2-1	172.26.142.1	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epcc2-2	172.26.142.2	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epcc2-3	172.26.142.3	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epcc2-4	172.26.142.4	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epcc2-5	172.26.142.5	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epcc2-6	172.26.142.6	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epcc2-7	172.26.142.7	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epcc2-8	172.26.142.8	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epcc2-9	172.26.142.9	N/A	255.255.0.0	172.26.252.1	172.26.252.2
usmv-epcc2-10	172.26.142.10	N/A	255.255.0.0	172.26.252.1	172.26.252.2

Appendix – Test Documents used for load and retrieval

<i>File Name/Type</i>	<i>Size (Bytes)</i>
1768.jpg	25,817
1784.jpg	23,926
2634.jpg	25,638
2897.jpg	27,608
3517.jpg	26,232
3531.jpg	26,236
5099.jpg	25,902
5102.jpg	27,562
512.jpg	24,371
5132.jpg	24,178
5135.jpg	27,321
6016.jpg	24,044
653.jpg	25,231
656.jpg	26,736
659.jpg	25,532
6979.jpg	27,011
6985.jpg	23,467
8227.jpg	24,367
8230.jpg	24,999
8329.jpg	26,990
8350.jpg	27,511
alfresco.log	3,238
Break Even Analysis.xls	36,352
Cashflow Forecast.xls	23,552
crud.txt	7,583
Employee Interview Questionnaire.doc	44,544
Marketing Activity Analysis Sheet.doc	29,696
ml-permissions.txt	271
New Client Needs Analysis Form.doc	62,976
Project Briefing Paper For Contractors.doc	49,664
QA component list Final.ods	53,957
Site Development Process.doc	52,224
sitepoint_logo.jpg	69,448
srs_template.doc	55,296
Update.pdf	49,733
User-Authentication-HOWTO.pdf	51,350
Valgrind-HOWTO.pdf	56,827
VCR-HOWTO.pdf	58,144
VMailMgr-HOWTO.pdf	44,336
Web Development Seminar.doc	43,520
Web development tips- Ten (or a few more) files every Web site needs.pdf	83,446
Windows-Newsreaders-under-Linux-HOWTO.pdf	50,427
WordPerfect.pdf	51,693
XFree86-R200.pdf	48,470

For more information, contact your Unisys representative.

In a hurry to learn more, Visit:
<http://www.unisys.com>

Specifications are subject to change without notice.

© Unisys Corporation. All rights reserved.

Unisys is a registered trademark of Unisys Corporation. Intel and Xeon are registered trademarks of Intel Corporation. Microsoft and Windows are registered trademarks of Microsoft Corporation. Red Hat is a registered trademark of Red Hat, Inc. Linux is a registered trademark of Linus Torvalds. All other brands and products referenced herein are acknowledged to be trademarks or registered trademarks of their respective holders.

