Performance benchmarking of Alfresco Content Services (ACS) on Red Hat OpenShift

IBM Power S1024 versus Intel Xeon Platinum 8260

Crest Infosolutions Sdn Bhd

Kuala Lumpur, Malaysia

Table of Contents

Introduction

This white paper describes the comparative performance characteristics of Alfresco® Content Services (ACS) running on Red Hat® OpenShift® Container platform on IBM® Power10 (ppc64le) architecture and Intel® x86_64 architecture.

ACS offers full-featured Enterprise Content Management (ECM) for organizations that require enterprise-grade scalability, performance, and 24x7 support for business-critical content and compliance. It delivers a wide range of use cases, such as content and governance services, contextual search and insight, and the ability to easily integrate with other applications. At the core of content services is a repository supported by a server that persists content, metadata, associations, and full text indexes.

IBM Power® is built for core enterprise applications, and the next wave of digital transformation is fueled by application modernization. IBM Power server's multi-layered approach to security gives you full visibility of your hardware and software. Power10 hardware-accelerated transparent memory encryption, quantum-safe cryptography, and fully homomorphic encryption protects your data with comprehensive end-to-end security at every layer of the stack.

Enable 34.3 times more throughput per core and 48% lower 3-year total cost of ownership (TCO) by running containerized applications and databases on an IBM Power E1080 server, compared to running the same containerized applications on an x86 server.[1]

Running Red Hat OpenShift in a virtual machine adjacent to your AIX®, IBM i, or Linux® virtual machines provides low-latency, reliable communication to your enterprise data with IBM PowerVM® Virtual I/O Server. This provides improved performance due to fewer network hops. It also allows for highly security-enhanced communication between your new cloud-native apps and your enterprise data stores because network traffic never has to leave the physical server.

Objectives

This performance benchmarking study aims to compare the performance of the Alfresco content management system when deployed on systems running on IBM Power10 processor and architecture with those running on X86_64 (Intel).

The test team measured the CPU, memory consumption, and throughput on both systems. The same set of tasks and activities are planned and run on both systems. The time it took to complete each task is recorded and used for comparison.

System architecture

For this exercise, the team deployed ACS version 7.2.0 on OpenShift, running on an IBM Power10 processor-based server, IBM Power S1024 (ppc64le architecture). Alfresco was not supported on the IBM Power platform at the time, so the test team compiled it using the source code provided by Alfresco's parent company, Hyland Software. Alfresco is now supported on IBM Power.

Benchmarking environment

IBM Power S1024 (based on the Power10 processor technology) with a CPU speed of 3.1 GHz was used for the benchmarking. For more details on Power S1024, refer to https://www.ibm.com/products/IBM Power-s1024.

Intel Xeon® Platinum 8260 (Cascade Lake) with a CPU speed of 2.4 GHz was the Intel (X86_64) system used for the benchmarking.

Only the CPU has been taken into account as an important consideration in performance testing. The system is deployed on a container-based platform, and the container platform allocates all the CPU specifications used during deployment as vCPU allocation.

Resource allocation for the critical components:
Repository pods
- Quantity – 2
- CPU – 8 vCPUs each
- Memory – 8 GB each

Solr search pod
- Quantity – 1
- CPU – 4 vCPUs
- Memory – 2 GB

PostgreSQL database
- Quantity – 1
- CPU – 16 vCPUs
- Memory – 32 GB

The layout of OpenShift cluster deployment is depicted in the following figure.

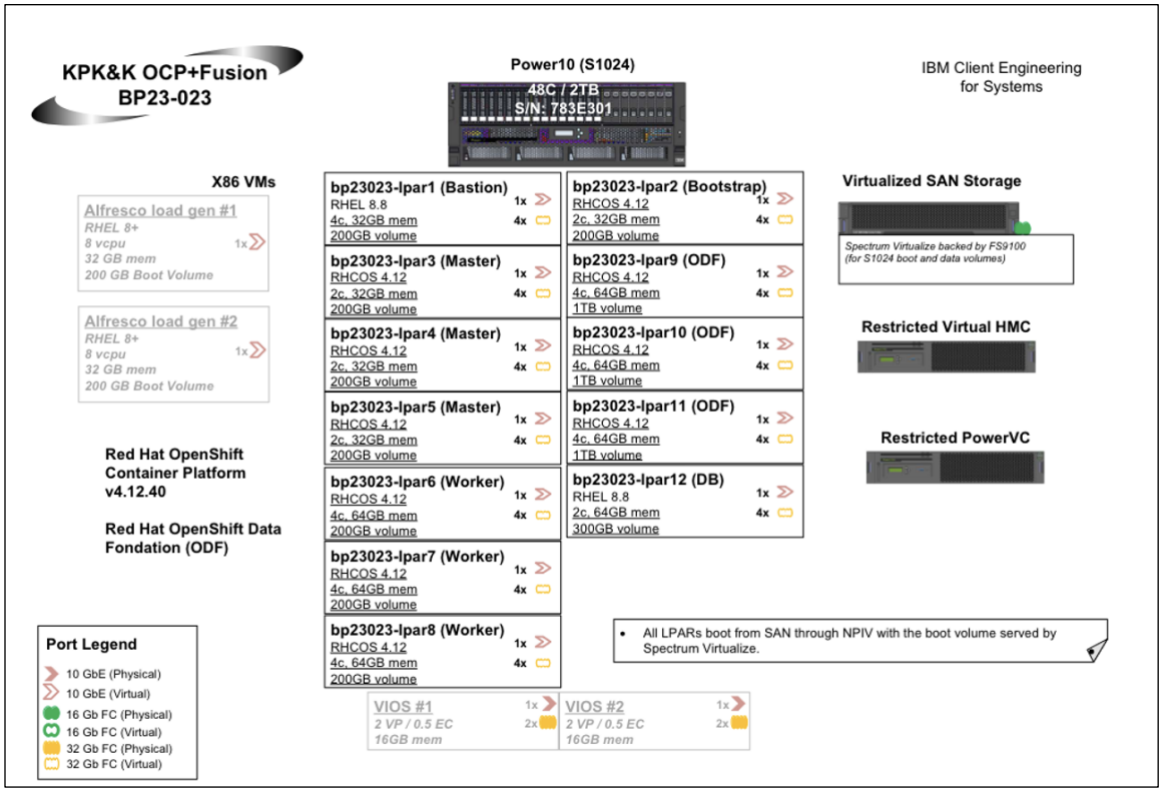


Figure 1. OpenShift cluster deployment layout

Red Hat OpenShift Data Foundation was the storage used for this benchmarking.

IBM Storage Fusion is a data services solution for OpenShift/Kubernetes designed by IBM and Red Hat (specifically for OpenShift). It is an integrated cloud-native application and data services platform that delivers a simplified and consistent experience across all Red Hat OpenShift environments, whether on-premises or cloud.

IBM Storage Fusion, which includes Red Hat OpenShift Data Foundation (also known as Fusion Data Foundation), is by far one of the best data services solutions for OpenShift/Kubernetes.

It provides intelligent data protection, rapid recoverability, security, and resilience, and it is deployed as container-native, easily consumable Kubernetes operators and custom resources.

The following figure depicts the architecture of IBM Storage Fusion on IBM Power.

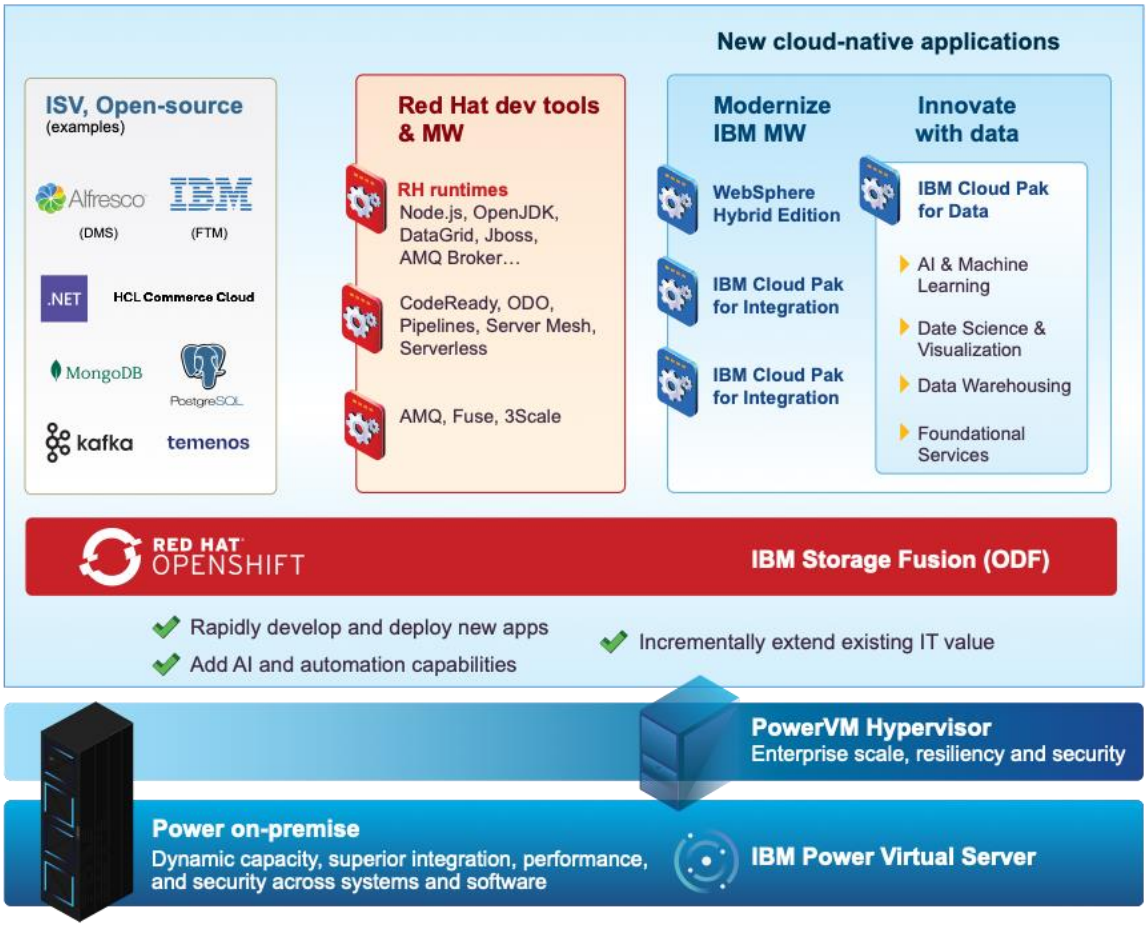


Figure 2. IBM Storage Fusion on IBM Power Architecture

The following section describes how IBM Storage Fusion fits into the architecture with Power10 and the benefits of using container native OpenShift Data Foundation in the architecture solution.

Red Hat OpenShift Data Foundation on IBM Power is supported in two on-premises cloud configurations based on IBM PowerVC and IBM PowerVM. The public cloud implementation is based on IBM Power Virtual Servers in the IBM Cloud®.

Controller nodes provide services that are cluster-wide in scope, including the Kubernetes API server, node configuration management, and more. They manage all the nodes in the cluster and schedule pods to run on those nodes.

Storage nodes aggregate the storage provided by storage worker nodes into storage pools with cluster-wide scope. These nodes offer APIs that allow storage to be consumed by the pods.

Workload nodes are dedicated to application workloads. Containerized applications can run on these nodes and use persistent volume claims and persistent volumes prepared with Ceph block or file storage classes.

IBM Storage Fusion on IBM Power can help optimize infrastructure costs by reducing the number of servers needed. It can also help maximize infrastructure utilization by dynamically allocating resources. Overall, container native OpenShift Data Foundation can provide several benefits when used with IBM Power10. These benefits include improved performance, increased scalability, and improved security.

IBM Storage Fusion is not only about persistent storage. Fusion delivers integrated data services for OpenShift Container Platform on Power, which include:

- Enterprise storage and data services
- Data protection: Backup restore
- Data resiliency: metro disaster recovery (DR) and regional DR
- Data cataloging
- Hybrid cloud integration
- Access data without data movement
- Modernize AI workloads
- Organize and optimize resources

Some benefits of using container native OpenShift Data Foundation in the architecture solution include:

- Ease of deployment and management
- Integration with Kubernetes and OpenShift
- Ease of use and greater efficiency
- Better application deployment
- Support for ReadWriteOnce (RWO) and ReadWriteMany (RWX) access modes
- Support for unified block, file, and object storage types

OpenShift Data Foundation configuration

OpenShift Data Foundation configuration includes three dedicated storage nodes [(IBM Power S1024 logical partition (LPAR)], each with 4 core (8 vCPU), 64 GB memory, and 1 TB disks.

LPAR/VM	OS	CPU Type	CPU (core)	vCPU	Memory (GB)	OpenShift Data Foundation disk
OpenShift Data Foundation Node 1	RHCOS 4.12	Shared	4	32	64	1 TB
OpenShift Data Foundation Node 2	RHCOS 4.12	Shared	4	32	64	1 TB
OpenShift Data Foundation Node 3	RHCOS 4.12	Shared	4	32	64	1 TB

Table 1. OpenShift Data Foundation configuration

Summary OpenShift Data Foundation Performance
OpenShift Data Foundation delivers higher performance than its competitors up to 3 times compared to the performance achieved when using the standard configuration (with default CPU and RAM) and up to 7.5 times higher performance when using the optimized configuration (which utilized more of the available CPU and RAM per node) (see Figure 3).

OpenShift Data Foundation is easy to install, monitor, and manage with the OpenShift Container Platform web console.

OpenShift Data Foundation is highly customizable for higher performance, while competitors have limited customization options.

For more information, refer to the Evaluator Group study comparing several storage solutions for cloud-native applications: https://www.redhat.com/en/resources/evaluator-group-scalable-storage-performance-analyst-material

Comparing performance of OpenShift Data Foundation with other container native storage solutions
Red Hat OpenShift Data Foundation storage showed significant differences compared to other container-native storage solutions. The most notable difference was OpenShift Data Foundation's superior performance scalability and consistent performance over time.

During the actual Sysbench performance testing itself, Red Hat OpenShift Data Foundation outperformed the two competitors in every instance:

The following observations were made when using the standard configuration for Red Hat OpenShift Data Foundation while scaling the workload compared to the competitors:
- OpenShift Data Foundation achieved 1.4x to 3.0x the total transactions per second (TPC) as *Vendor A*.
- OpenShift Data Foundation achieved 1.8x the total TPC as *Vendor B*.

Using an optimized configuration for Red Hat OpenShift Data Foundation while scaling the workload compared to the competitors:
- OpenShift Data Foundation achieved 3.5x to 5.7x the total TPC as *Vendor A*.
- OpenShift Data Foundation achieved 2.1x to 7.5x the total TPC as *Vendor B*.

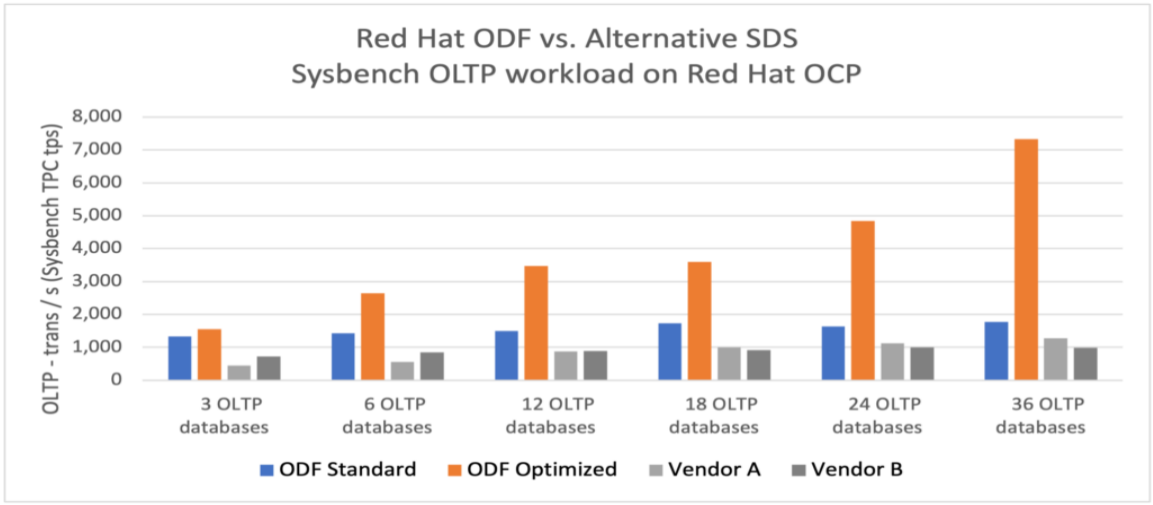
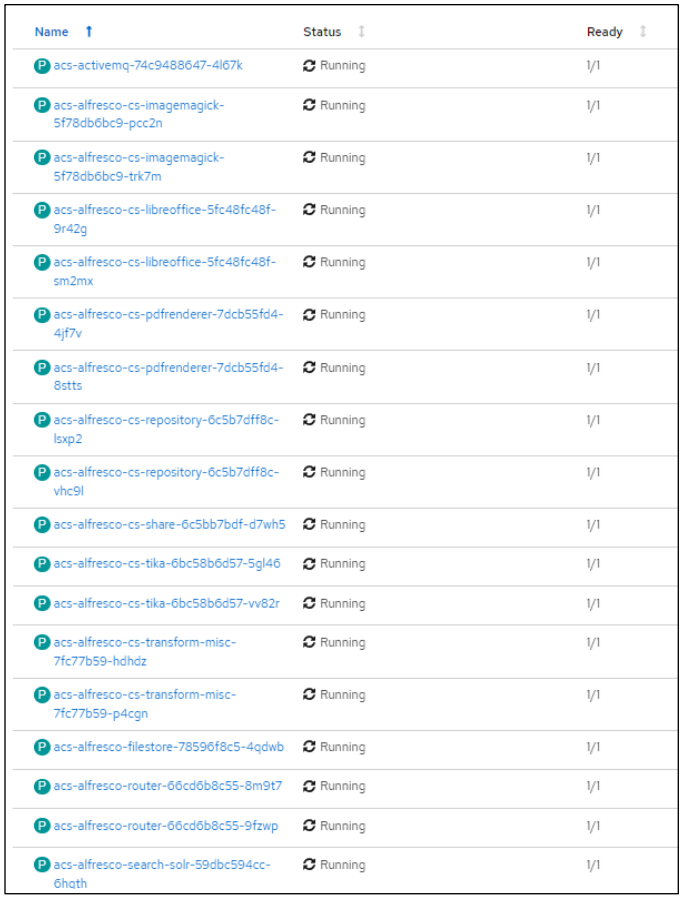


Figure 3. Red Hat OpenShift Data Foundation Storage Performance vs. Alternatives (Source Evaluator Group Testing)

Note: For details on this test report, visit https://www.redhat.com/en/resources/evaluator-group-scalable-storage-performance-analyst-material.

ACS 7.2.0 was deployed using the helm chart provided by the vendor. To improve performance during bulk upload, the test team disabled the transformation services while running the performance testing. Figure 4 shows the components and the number of pods used to conduct the performance testing.



Name	Status	Ready
acs-activemq-74c9488647-4l67k	Running	1/1
acs-alfresco-cs-imagemagick-5f78db6bc9-pcc2n	Running	1/1
acs-alfresco-cs-imagemagick-5f78db6bc9-trk7m	Running	1/1
acs-alfresco-cs-libreoffice-5fc48fc48f-9r42g	Running	1/1
acs-alfresco-cs-libreoffice-5fc48fc48f-sm2mx	Running	1/1
acs-alfresco-cs-pdfrenderer-7dcb55fd4-4jf7v	Running	1/1
acs-alfresco-cs-pdfrenderer-7dcb55fd4-8stts	Running	1/1
acs-alfresco-cs-repository-6c5b7dff8c-lsxp2	Running	1/1
acs-alfresco-cs-repository-6c5b7dff8c-vhc9l	Running	1/1
acs-alfresco-cs-share-6c5bb7bdf-d7wh5	Running	1/1
acs-alfresco-cs-tika-6bc58b6d57-5gl46	Running	1/1
acs-alfresco-cs-tika-6bc58b6d57-vv82r	Running	1/1
acs-alfresco-cs-transform-misc-7fc77b59-hdhdz	Running	1/1
acs-alfresco-cs-transform-misc-7fc77b59-p4cgn	Running	1/1
acs-alfresco-filestore-78596f8c5-4qdwb	Running	1/1
acs-alfresco-router-66cd6b8c55-8m9t7	Running	1/1
acs-alfresco-router-66cd6b8c55-9fzwp	Running	1/1
acs-alfresco-search-solr-59dbc594cc-6hqth	Running	1/1

Figure 4. List of Alfresco Content Services pods running on OpenShift

Benchmarking metrics and methodology

The test team measured the following metrics during the performance testing: CPU usage, Memory usage, Throughput, and Response time.

The details of test scenarios, workload, and specific test cases are explained in the following sections.

Description of tests and test results

The test team ran the following four tests, creating different load characteristics on the Alfresco Content Services system.

Test 1: Create 1000 users.
Test 2: Create 100 sites and add 1000 users to the sites.
Test 3: List the number of folders, create a few random files, and search those files.
Test 4: Upload 5 million files (average size of 50 KB each) of various types and measure the performance.

Test summary of OpenShift on IBM Power S1024

Test 1: Create 10,000 users.
This test performs the following activity:
a. Creates 10,000 users on the Alfresco database and creates 10,000 home folders for those users.

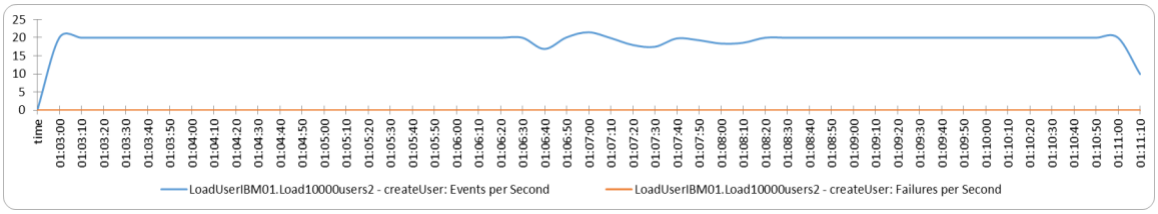


Figure 5. Creating 10,000 users on the Alfresco database

It took 510 seconds (7 mins and 20 sec) to create 10,000 users. On average, the test created 20 users per second.

Test 2: Create 100 sites and add 1000 users to the sites.
This test performs the following activities:
a. Create 100 sites.
b. Add 10 users to each of those sites.

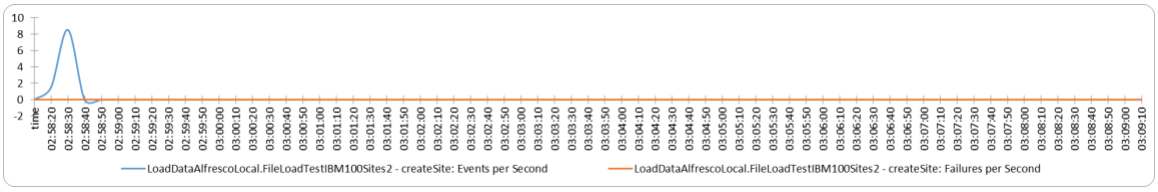


Figure 6a. Creating 100 sites

It took 20 seconds to create 100 sites.

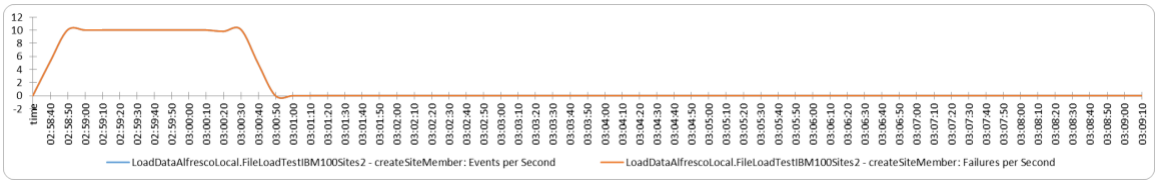


Figure 6b. Adding 10 users to each of the 100 sites

It took 120 seconds (2 minutes) to add the users to the sites.

Test 3: List the number of folders, create a few random files, and search those files.
This test performs the following activities:
a. Get the list of all the folders created during test 1.
b. Create a random number of files in those folders.
c. Search randomly created files to check Alfresco's indexing and search performance.

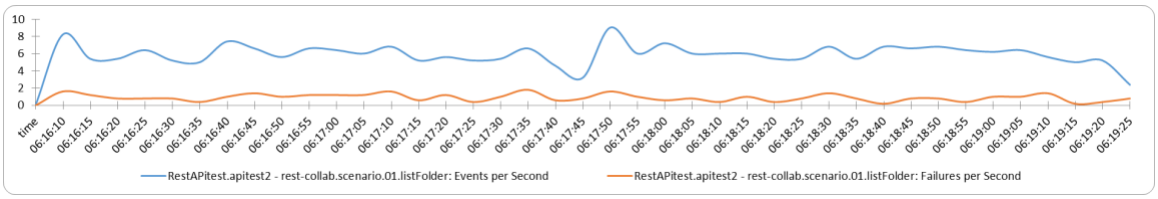


Figure 7a. Gathering the list of all the folders created during Test 1

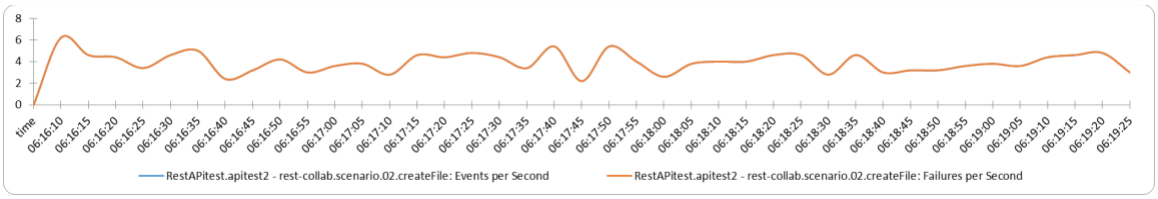


Figure 7b. Creating files in the folders created during Test 1

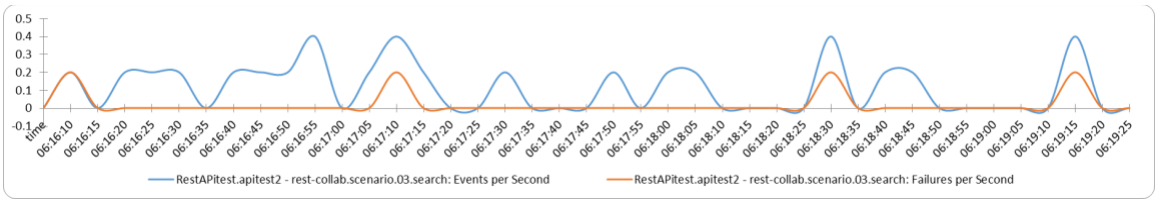


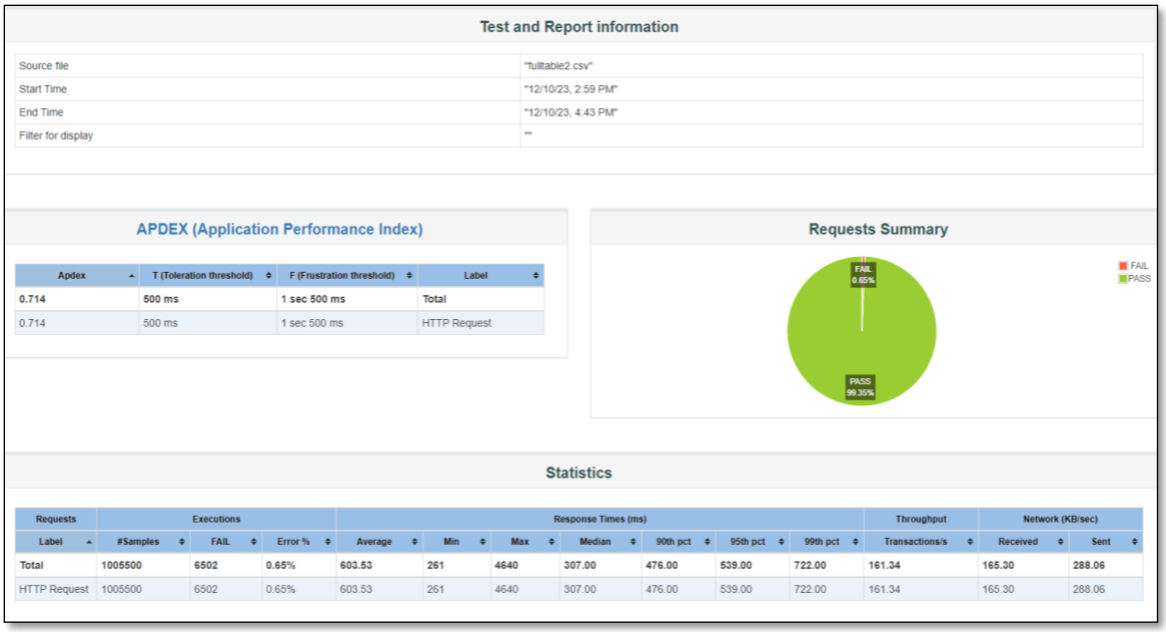
Figure 7c. Searching randomly created files to check Alfresco's indexing and search performance

All three tests ran simultaneously and were completed in 195 seconds (3 minutes and 15 seconds).

Test 4: Upload 5 million files (average size of 50 KB each) of various types and measure the performance.

1. PDF files – 2 million
2. Microsoft Word files – 1 million
3. Microsoft Excel files – 1 million
4. Text files – 1 million

As the test team ran a bulk upload, it was decided to disable the transformation service to eliminate bottlenecks. They observed that the system performance was similar to all the above types of files.



Test and Report information

Source file	"fulltable2.csv"
Start Time	"12/10/23, 2:59 PM"
End Time	"12/10/23, 4:43 PM"
Filter for display	""

APDEX (Application Performance Index)

Apdex	T (Toleration threshold)	F (Frustration threshold)	Label
0.714	500 ms	1 sec 500 ms	Total
0.714	500 ms	1 sec 500 ms	HTTP Request

Requests Summary

Statistics

Requests	Executions			Response Times (ms)							Throughput	Network (KB/sec)	
Label	#Samples	FAIL	Error %	Average	Min	Max	Median	90th pct	95th pct	99th pct	Transactions/s	Received	Sent
Total	1005500	6502	0.65%	603.53	261	4640	307.00	476.00	539.00	722.00	161.34	165.30	288.06
HTTP Request	1005500	6502	0.65%	603.53	261	4640	307.00	476.00	539.00	722.00	161.34	165.30	288.06

Figure 8a. Overall performance

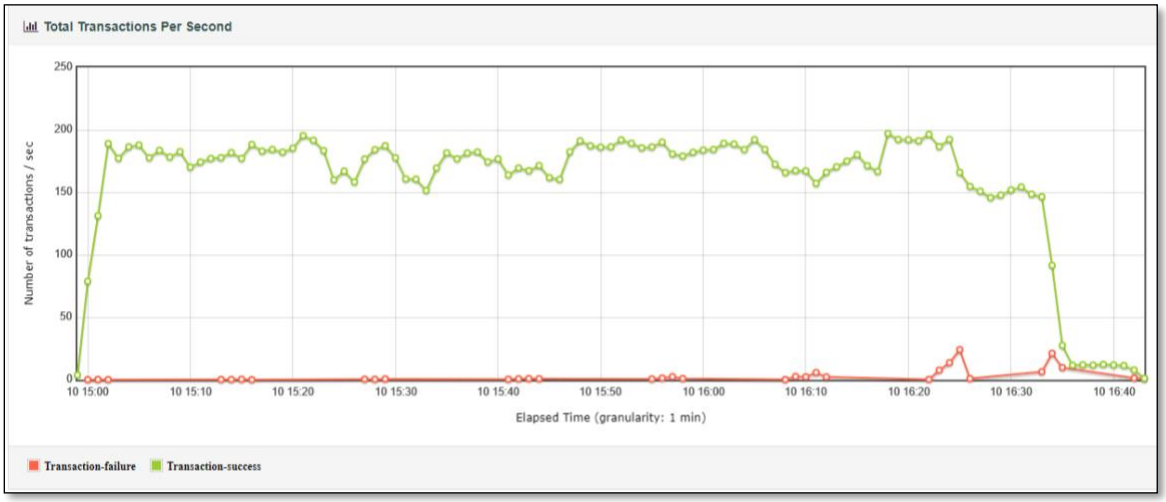


Figure 8b. Total transactions per second

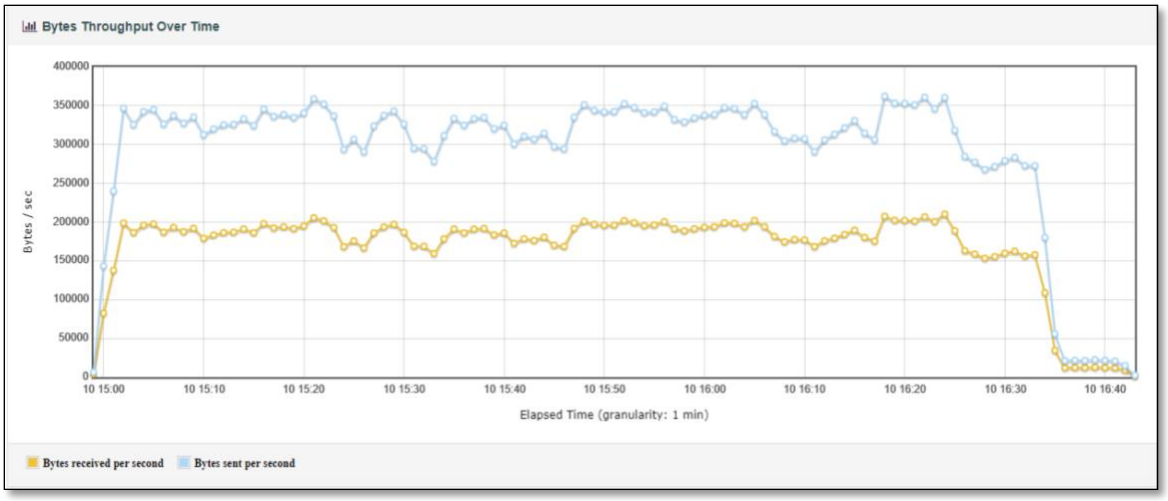


Figure 8c. Bytes throughput over time

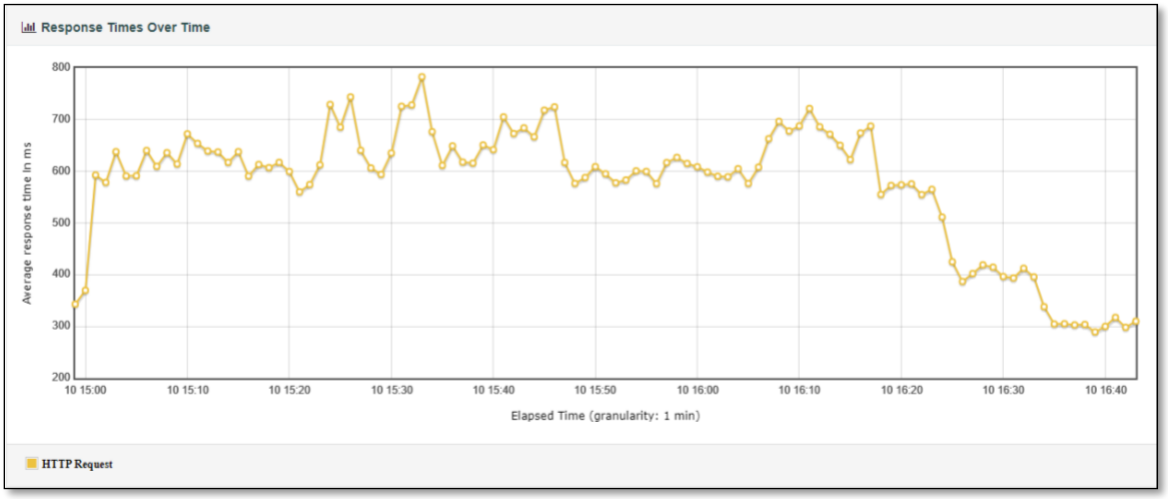


Figure 8d. Response times

Figure 8e. CPU usage (extracted from OpenShift)

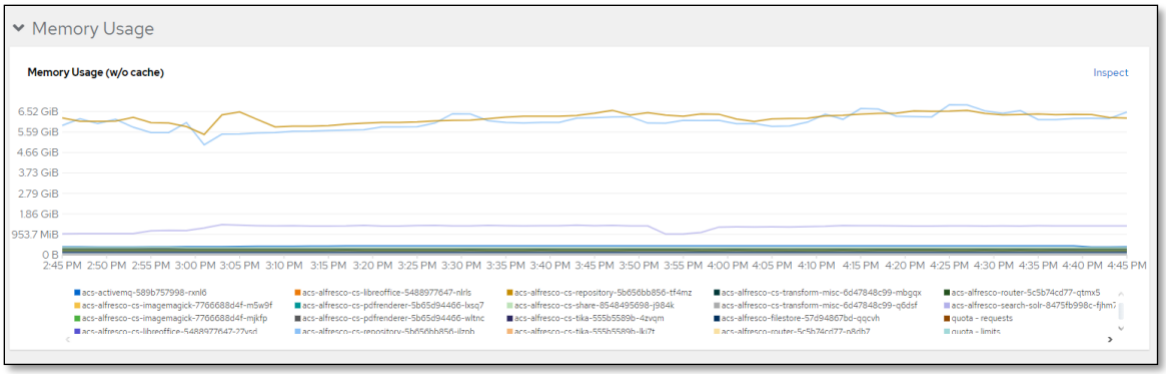


Figure 8f. Memory usage (extracted from OpenShift)

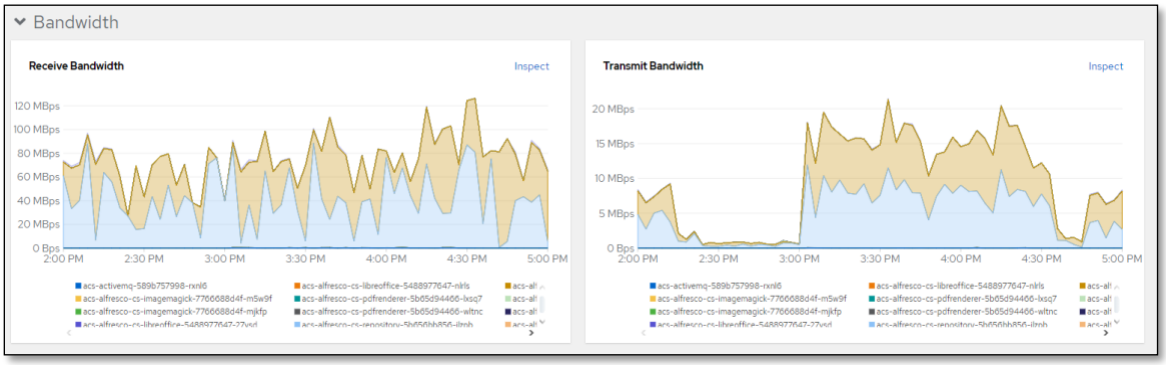


Figure 8g. Bandwidth usage (extracted from OpenShift)

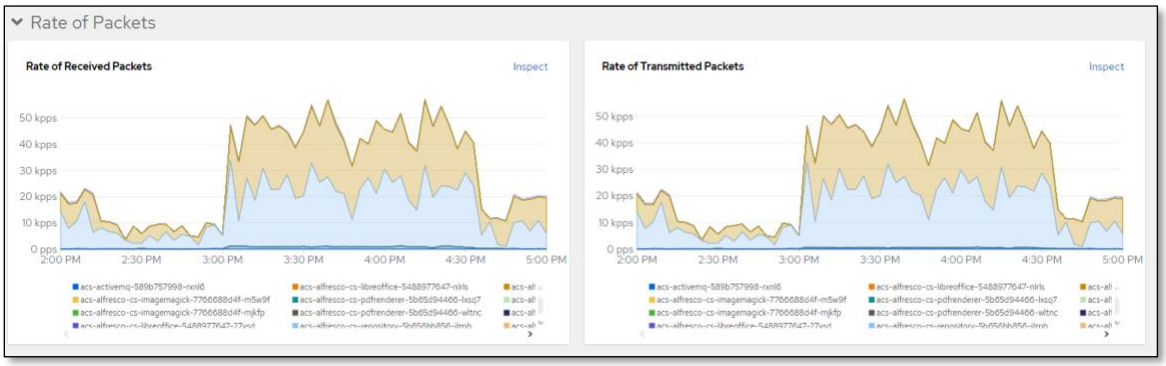


Figure 8h. Rate of packets

The system uploaded 1 million records in 104 minutes (1 hour and 44 minutes). The performance was similar for all types of files, and the charts above depict the upload performance of the PDF files. The end period in the chart shows the ramping down of the test, indicating declining performance.

Test summary of OpenShift on Intel Xeon Platinum 8260

Test 1: Create 1000 users.

This test performs the following activity:

a. Create 10,000 users on the Alfresco database and create 10,000 home folders for those users.

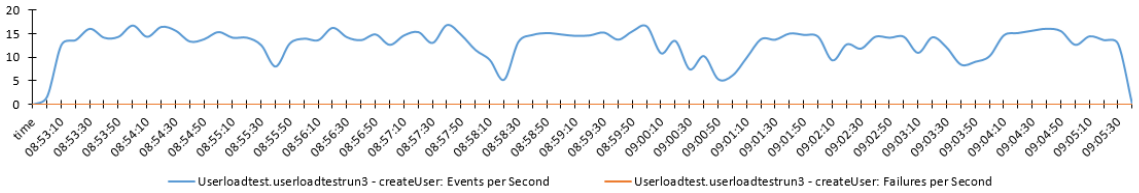


Figure 9. Creating 10,000 users on the Alfresco database

It took 800 seconds (13 minutes and 20 seconds) to create 10,000 users. On average, the test has created 13 users per second.

Test 2: Create 100 sites and add 1000 users to the sites.

This test performs the following activities:

a. Create 100 sites.
b. Add 10 users to each of those sites.

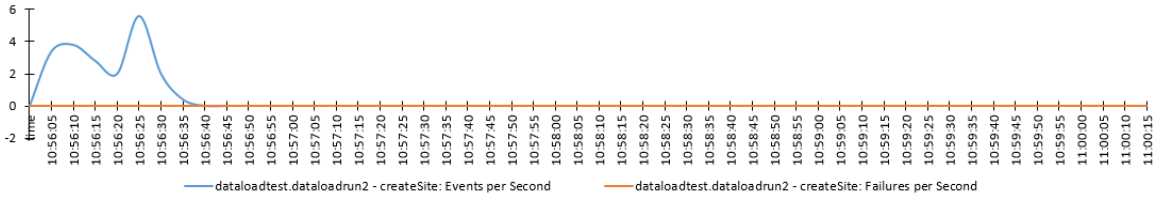


Figure 10a. Creating 100 sites

It took 35 seconds to create 100 sites.

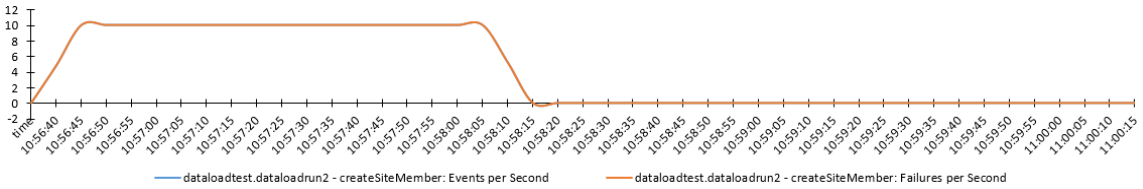


Figure 10b. Adding 10 users to each of the 100 sites

It took 120 seconds (2 minutes) to add the users to the sites.

Test 3: List the number of folders, create a few random files, and search those files.

This test performs the following activities:

a. Get the list of all the folders created during test 1.
b. Create a random number of files in those folders.
c. Search randomly created files to check Alfresco's indexing and search performance.

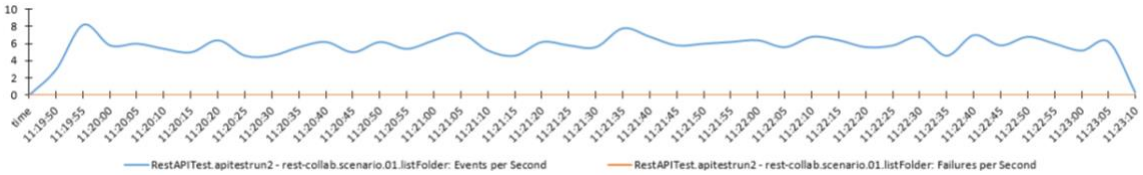


Figure 11a. Gathering the list of all the folders created during Test 1

Figure 11b. Creating files in the folders created during Test 1

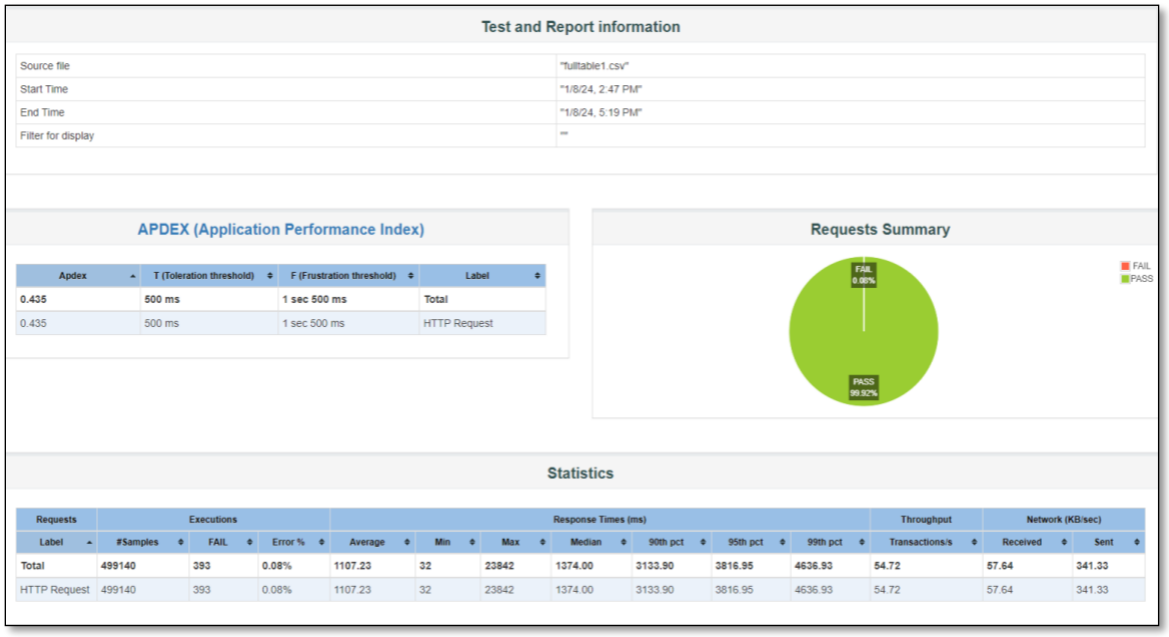
Figure 11c. Searching randomly created files to check Alfresco's indexing and search performance

All three tests ran simultaneously and were completed in 200 seconds (3 minutes 20 seconds).

Test 4: Upload 5 million files (average size of 50 KB each) of various types and measure the performance.

- PDF files – 2 million
- Word files – 1 million
- Excel files – 1 million
- Text files – 1 million

As the test team ran a bulk upload, it was decided to disable the transformation service to eliminate bottlenecks. It was observed that the system performance was similar with all four types of files.



Test and Report information

Source file	"fulltable1.csv"
Start Time	"1/8/24, 2:47 PM"
End Time	"1/8/24, 5:19 PM"
Filter for display	""

APDEX (Application Performance Index)

Apdex	T (Toleration threshold)	F (Frustration threshold)	Label
0.435	500 ms	1 sec 500 ms	Total
0.435	500 ms	1 sec 500 ms	HTTP Request

Requests Summary

Statistics

Requests	Executions			Response Times (ms)							Throughput	Network (KB/sec)	
Label	#Samples	FAIL	Error %	Average	Min	Max	Median	90th pct	95th pct	99th pct	Transactions/s	Received	Sent
Total	499140	393	0.08%	1107.23	32	23842	1374.00	3133.90	3816.95	4636.93	54.72	57.64	341.33
HTTP Request	499140	393	0.08%	1107.23	32	23842	1374.00	3133.90	3816.95	4636.93	54.72	57.64	341.33

Figure 12a. Overall performance

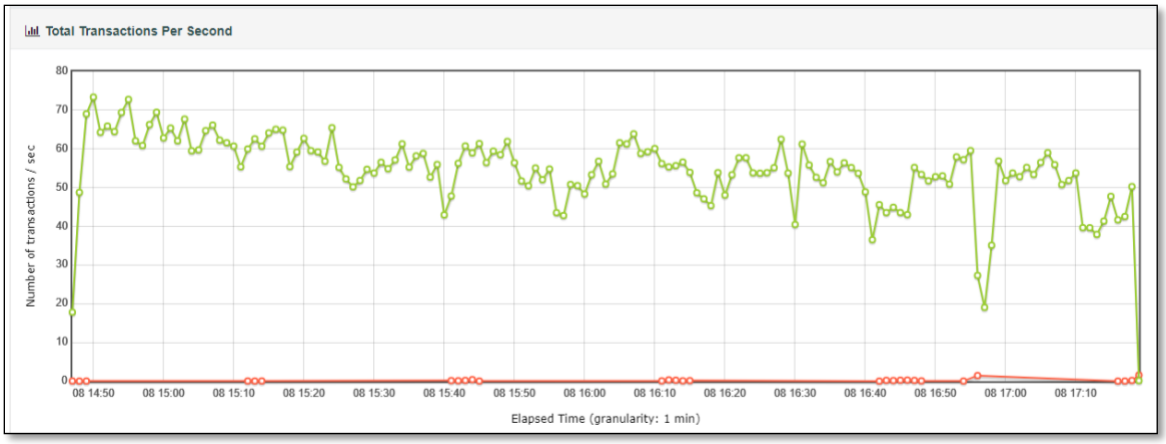


Figure 12b. Transactions per second

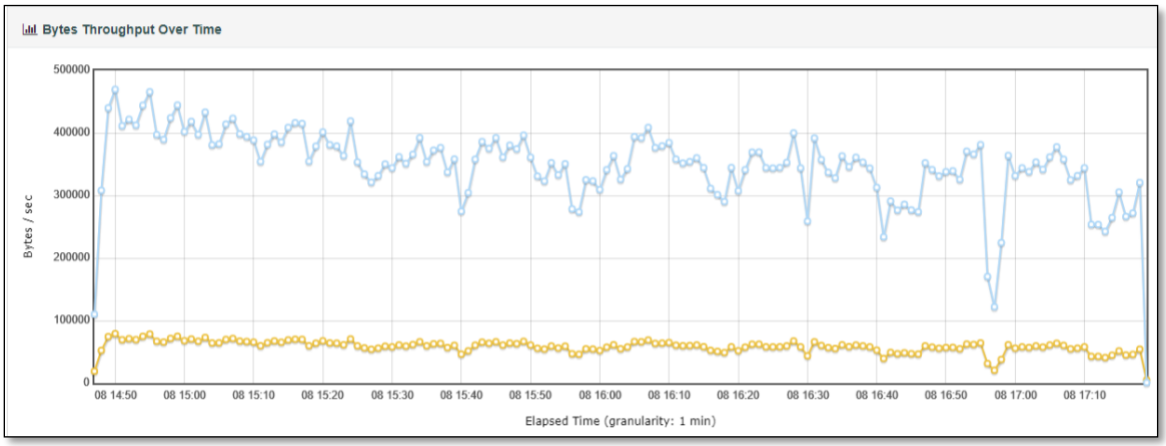


Figure 12c. Bytes throughput

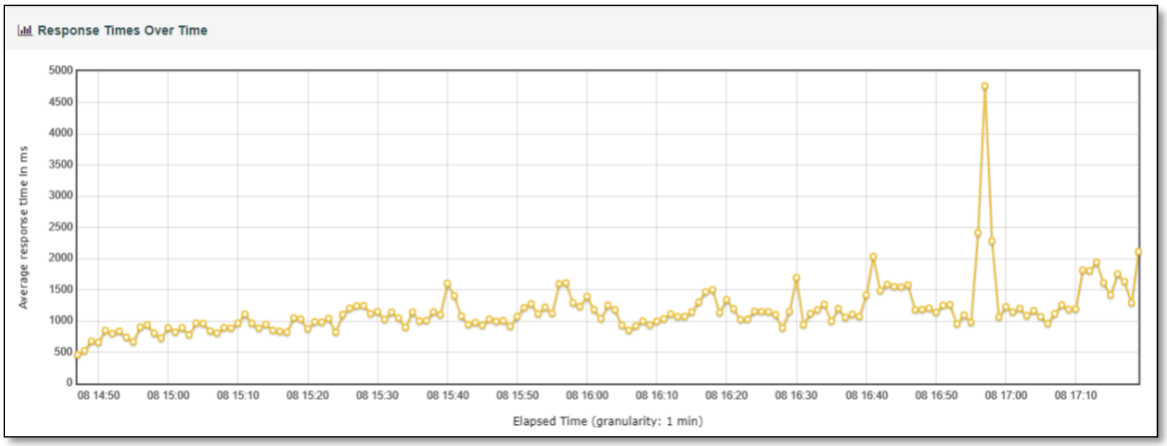


Figure 12d. Response times

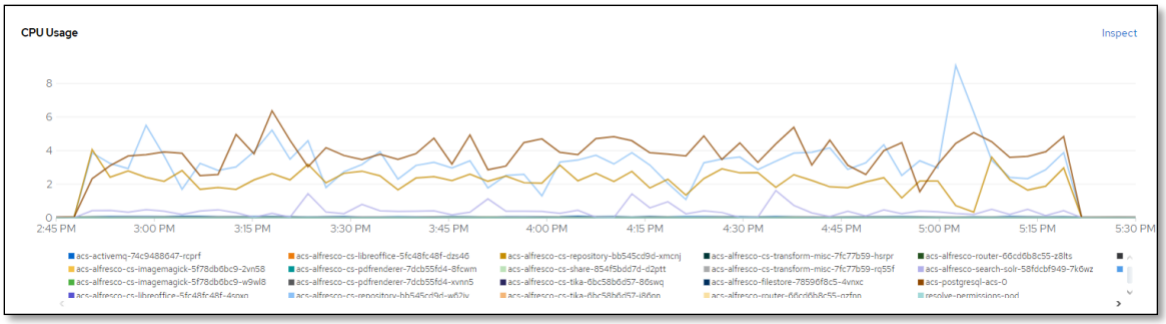


Figure 12e. CPU usage (extracted from OpenShift)

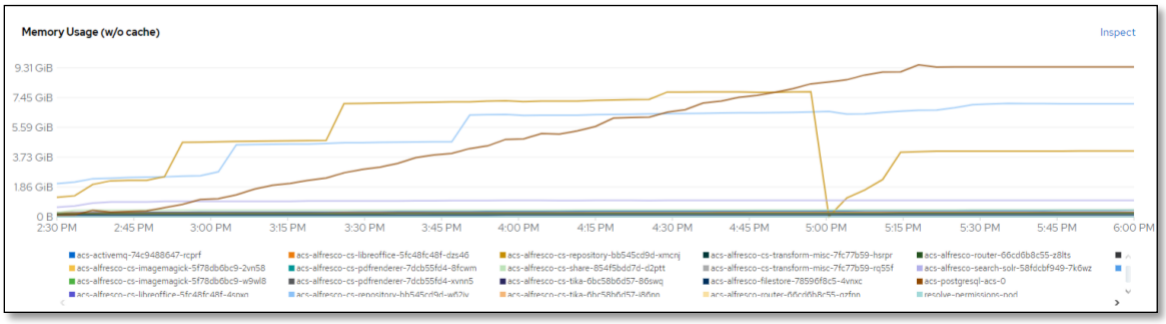


Figure 12f. Memory usage (extracted from OpenShift)

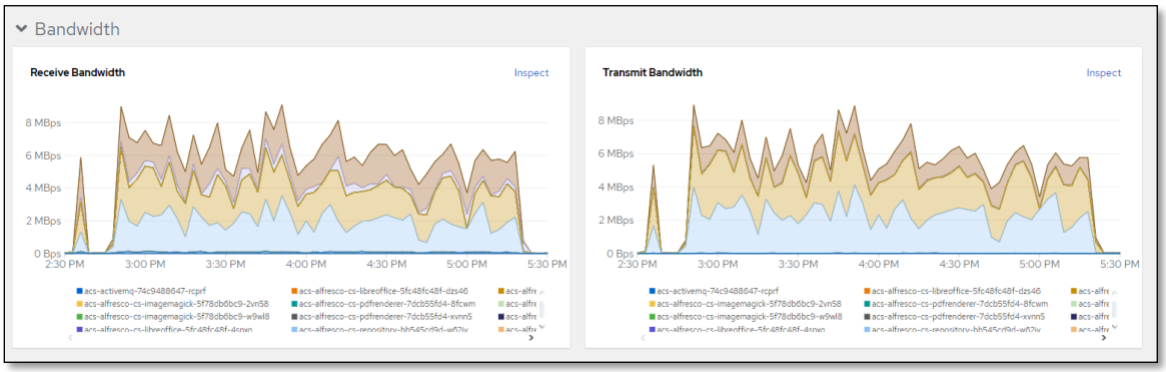


Figure 12g. Bandwidth usage (extracted from OpenShift)

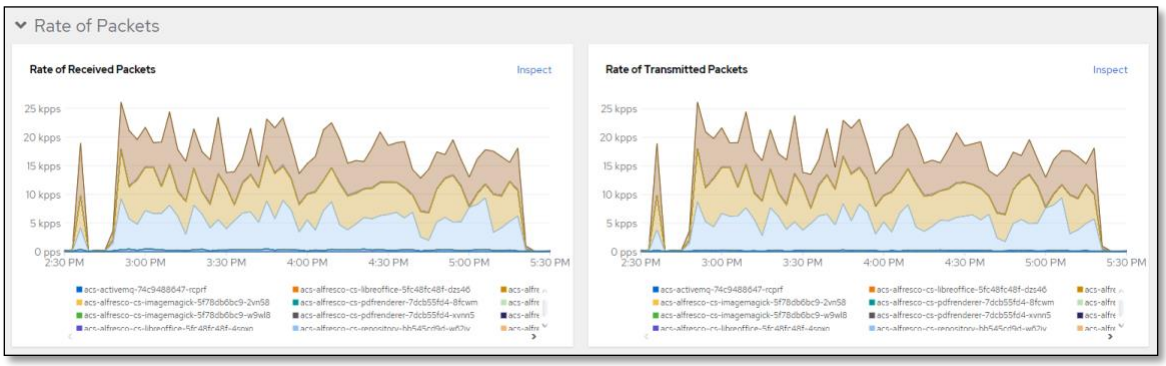


Figure 12h. Rate of packets

The system uploaded 1 million records in 152 minutes (2 hours and 32 minutes). The performance was similar for all types of files, and the charts above depict the upload performance of the PDF files. The end period in the chart shows the ramping down of the test, indicating declining performance.

Comparative analysis (optional)

Perform the following tasks to do a comparative analysis:

1. Compare the performance of IBM Power10 running Alfresco Content Services with x86
2. Highlight the differences and the reasons, for example, scalability, I/O bandwidth, and memory bandwidth
3. Identify any performance bottlenecks or limitations in either case

Conclusion

The key test case of bulk files uploaded in Alfresco showed considerably better results on the Power S1024 server than on the Intel Xeon Platinum 8260 (Cascade Lake). While the Intel Xeon Platinum 8260 showed varied resource usage during tests, the Power S1024 server showed consistent memory, CPU, disk IO performance, and network bandwidth usage. Consistent resource usage and response time on the Power S1024 with OpenShift indicates the stability of the platform and the application and can facilitate horizontal scalability of the Alfresco application to cater to larger workloads, potentially billions of documents ingested and retrieved, without compromising application performance. Because per core performance for the Power S1024 server was observed to be significantly better than the Intel Xeon Platinum 8260, Alfresco customers can benefit from smaller server footprints for even larger workloads, thereby saving space and energy for sustainable deployments.

Key findings

Table 2 depicts the comparison of vCPU (IBM Power S1024) with vCPU (Intel Xeon Platinum 8260). Both systems are configured to use the same number of vCPUs.

Test	Test description	Time taken (Power S1024)	Time taken (Intel Xeon Platinum 8260)	Difference
1	Create 10,000 users	510 seconds	800 seconds	44.27%
2	Create 100 sites	20 seconds	35 seconds	54.55%
2	Add 10 users to each of the 100 sites	120 seconds	120 seconds	0%
3	List the number of folders, create a few random files, and search those files.	195 seconds	200 seconds	2.53%
4	Upload 5 million files of various types and measure the performance	104 minutes	152 minutes	37.5%
Application performance index		0.714	0.435	48.56%

Table 2. Key findings comparing Power10 to Intel x86

- The difference between the two systems is not much when running smaller tests, even though the IBM Power S1024 system always performed slightly better.
- The average CPU and memory usage are similar for both systems during performance testing.
- The IBM Power S1024 system performed better when creating 10,000 users, creating 100 sites, listing the number of folders, creating a few random files, and searching those files, and uploading 5 million files of various types and measuring the performance.
- The Application Performance Index (Apdex) shows that the IBM Power S1024 performance is 51% better.

- Therefore, for ACS running on OpenShift, each IBM Power vCPU performs 50% better than Intel vCPU.
- Taking the above point as a reference, and given the fact that one IBM Power S1024 core is equivalent to eight vCPUs, and one Intel Xeon Platinum 8260 core is equivalent to two vCPUs, the performance of one IBM Power S1024 core can be calculated as follows:

  IBM Power S1024: 8 (threads) x 1.5 (performance) = 12
  Intel Xeon Platinum 8260: 2 (threads) x 1.0 (performance) = 2
  The result: 12 (IBM Power S1024) / 2 (Intel Xeon Platinum 8260) = 6

Figure 13 depicts the performance of one IBM Power S1024 core and one Intel Xeon Platinum 8260 core.

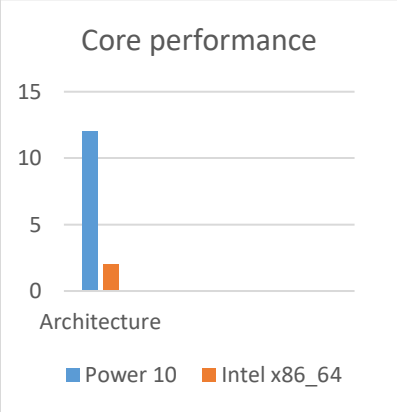


Figure 13. Core Performance

- One IBM Power S1024 core can perform 600% higher than an Intel Xeon Platinum 8260 core.

Contacts

IBM
Sahitya Jain, ISV Technical Engineering, IBM, sahityajain@in.ibm.com

Jason Xin Liu, IBM Power Data and AI Technical Sales Leader, IBM, liuxxin@cn.ibm.com

Vaibhav Shandilya, IBM Client Engineering Systems Solution Architect, IBM, svaibhavs@in.ibm.com

Nijo K J, Client Engineering for Systems, Solution Architect, IBM, nijokj04@in.ibm.com

Murali Neralla, ISV Solutions Architect, IBM, neralla1@us.ibm.com

Crest
Hemant Prasad, CEO, Crest Infosolutions Sdn Bhd, hemant.prasad@crestsolution.com

Rada Kichenin, CTO, Crest Infosolutions Sdn Bhd, rada@crestsolution.com

Endnotes

1. Based on IBM internal testing of Red Hat OpenShift Container Platform 4.8. Two worker nodes running 80 pods each with 10 users using the Daytrader7 workload (https://github.com/WASdev/sample.daytrader7/ releases/tag/v1.4) (link resides outside of ibm.com) accessing AIX Db2 databases. Average CPU utilization for the OCP worker nodes is >95%. Comparison: IBM Power E1080 with collocated OCP and AIX Db2 nodes versus OCP node on Cascade Lake accessing AIX Db2 node on Power E1080. Valid as of 25 August 2021 and conducted under laboratory conditions. Individual results can vary based on workload size, use of storage subsystems and other conditions. TCO is defined as hardware, software and maintenance costs over a period of three years. Power E1080 (40 cores/3.8 GHz/2 TB memory) in maximum performance mode, 25 Gb Ethernet adapter (SRIOV), 1 x 16Gbps FC adapter with PowerVM. Competitive system: Intel Xeon Gold 6248 CPU (Cascade Lake) in performance mode, 40 cores/3.9GHz/512GB memory), 25Gb Ethernet adapter (SRIOV), 1 x 16 Gbps FCA with KVM. Pricing is based on Power E1080 (http://www-03.ibm.com/ systems/power/hardware/linux-lc.html); typical industry standard x86 pricing (https://www.synnexcorp.com/us/govsolv/pricing/); and IBM software pricing for Red Hat OpenShift and IBM WebSphere Hybrid Edition Monthly Subscription.

IBM Corporation
New Orchard Road
Armonk, NY 10504

Produced in the United States of America
July 2024