IBM Cloud
Technical Whitepaper

IBM Aspera Direct-to-Cloud Storage

A Technical Whitepaper on the State-of-the Art in High Speed Transport Direct-to-Cloud Storage and Support for Third Party Cloud Storage Platforms

Contents:

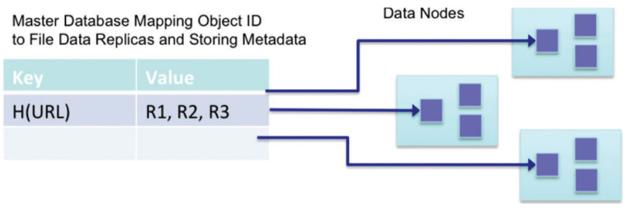
Overview

The Aspera FASP high speed transport platform is enabled to provide high-performance secure WAN transport of files, directories, and other large data sets to, from and between a number of leading third-party cloud storage platforms. The implementation is an enhanced transport stack and virtual file system layer in the Aspera server software that allows for direct-to-object-storage transfer over the WAN using the FASP protocol and the native I/O capabilities of the particular third-party file system. The stack is available in all generally available Aspera server software products and supports interoperable transfer with all generally available Aspera client software.

Aspera continually adds support for new third-party storage platforms as market demand is demonstrated, and in version 3.4 is pleased to currently support all leading cloud storage platforms including, OpenStack Swift (v 1.12) for IBM Cloud and Rackspace, Amazon S3, Windows Azure BLOB, Akamai NetStorage, Google Storage, and Limelight Cloud Storage. This whitepaper overviews the motivation for the platform – the fundamental problem of transporting large data sets to and from cloud environments – details the platform capabilities, and describes the performance and functionality testing that comprises verification of each storage platform.

The problem

The mainstream "Cloud" storage platforms are "object storage" architectures that emanate in design from the early scale out storage systems developed by the leading web search companies such as the Hadoop File System (HDFS), Google File System (GFS), and Amazon Dynamo. The key design principle of these object storage systems is to organize file data and associated metadata such as names, permissions, access times, etc., as an "object" and to store the file data and the metadata referring to it in a decoupled fashion, allowing for extreme scale and throughput. The file data is stored across distributed commodity storage in redundant copies to achieve reliability, and scale is achieved through a single namespace in which master tables store a hash of an object's identifiers and references to the copies of its file data on disk, allowing for fast and universal addressing of individual objects across the distributed platform (see Figure 1).


Master Database Mapping Object ID
to File Data Replicas and Storing Metadata
Data Nodes

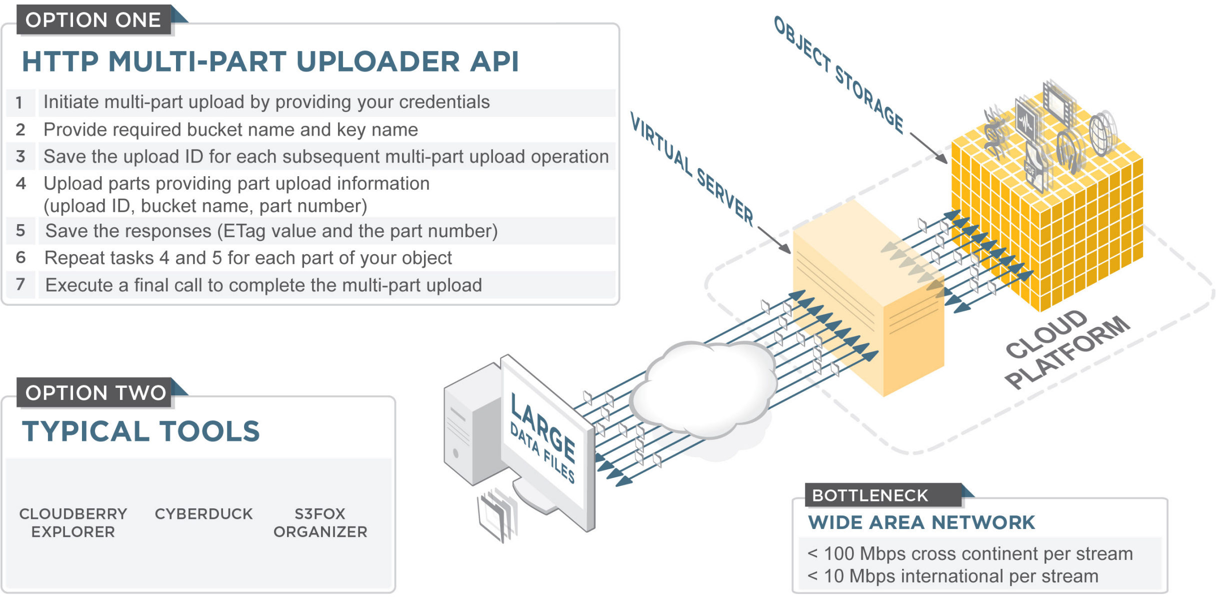
Key	Value
H(URL)	R1, R2, R3

Figure 1: Cloud Object Storage decouples file data from identifying metadata and distributes file data across underlying storage

This approach lends itself extremely well to storage for applications such as indexing for scalable web search, as it allows the application to utilize extremely large data sets, achieve very high aggregate throughput in batch processing, and use inexpensive commodity disks for the underlying storage.

At face it would seem that the scalability of such "object storage" platforms would also be ideal for storing large unstructured data types, such as large files and directories. However, at the core of the object storage design is the assumption that file data is written into the storage system in small "chunks" – typically 64 MB to 128 MB - and stored redundantly across the many physical disks. Each write requires writing multiple redundant copies of each chunk to disk and creating a reference to these copies in the master meta store. Similarly, an object can only be "read" out through a look up of the chunks that comprise it, retrieval from the storage, and reassembly.

An application uploading or downloading any single item greater than the chunk size (e.g., 64 MB) must divide and reassemble the object into appropriate chunks, which is itself tedious and has a bottleneck in transfer speed in the local area unless done in highly parallel fashion. For example, for 64 MB chunks, writing a 1 Terabyte file requires dividing it into more than 10,000 chunks, and throughput in practical implementations tops out at less than 100 Mbps per I/O stream. We refer to this as the local area storage bottleneck (see Figure 2).



OPTION ONE

HTTP MULTI-PART UPLOADER API

1 Initiate multi-part upload by providing your credentials
2 Provide required bucket name and key name
3 Save the upload ID for each subsequent multi-part upload operation
4 Upload parts providing part upload information
(upload ID, bucket name, part number)
5 Save the responses (ETag value and the part number)
6 Repeat tasks 4 and 5 for each part of your object
7 Execute a final call to complete the multi-part upload

OPTION TWO

TYPICAL TOOLS

CLOUDBERRY EXPLORER
CYBERDUCK
S3FOX ORGANIZER

BOTTLENECK

WIDE AREA NETWORK

< 100 Mbps cross continent per stream
< 10 Mbps international per stream

Figure 2: Multi-part HTTP transfer APIs suffer from a local area I/O bottleneck and a wide area transport bottleneck

Because cloud storage platforms are by definition typically at a WAN distance from the application uploading or downloading, this chunk-wise transmission is also limited by the fundamental performance limitations of TCP over the WAN. Specifically the S3-compatible "multi-part" object write and read APIs implemented by the mainstream cloud storage systems use HTTP as the reliable transport mechanism to PUT and GET each object chunk. At a typical cross-country WAN distance, the round-trip latency and packet loss are sufficient to limit the achievable throughput to <100 Mbps, and over international WANs to limit the achievable throughput to <10 Mbps. We refer to this as the WAN transport bottleneck (see Figure 2).

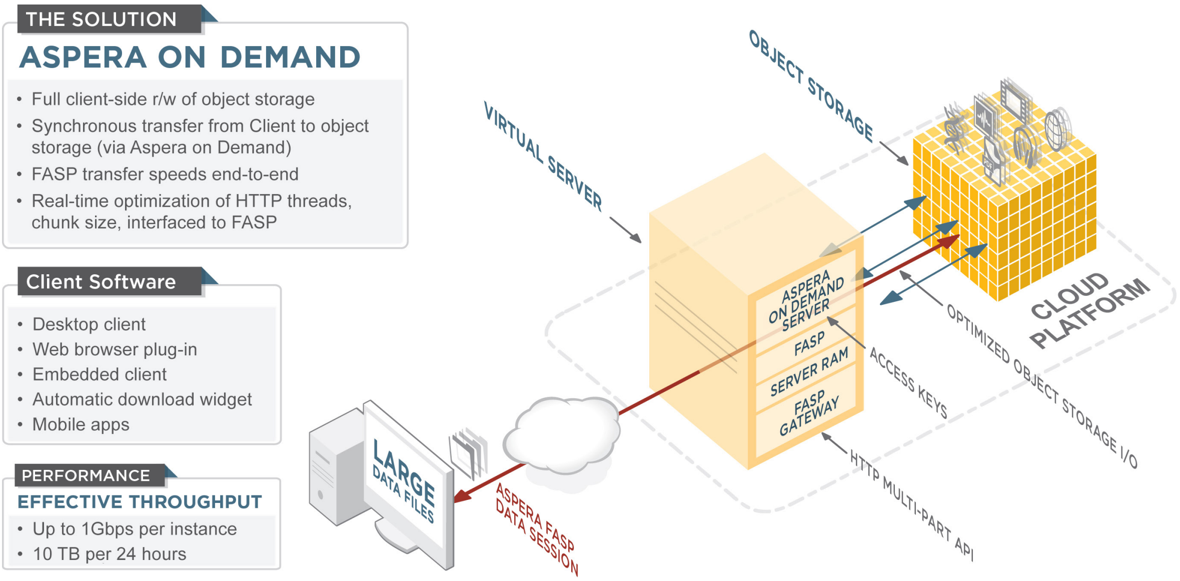
In addition to the storage and transport bottlenecks, the "multi-part" APIs do not support resume of uploads/downloads if an active session is interrupted, leaving this up to the application to manage. And, while HTTPS transmission will secure the transmission of the "chunks" over the wire, most cloud storage has either no option for encryption at rest OR requires the application to use an encryption option in the cloud file system, which can be very slow, inserting yet another bottleneck for high-speed upload or download. Finally, complimentary features such as browsing the object storage to view large files and directories requires building on top of the object storage APIs as there is no familiar file system hierarchy to present to end users.

To work around the programming and deployment challenges of using the multi-part APIs some applications turn to virtual file system drivers, such as "s3fs", a FUSE-based file system backed by Amazon S3, to virtually "mount" the object storage. This has the convenience of making the object storage present to the application as a hierarchical classical file system, but at the cost of extremely slow throughput. Large file read and write rates over s3fs, for example, are limited to less than 100 Megabits per second.

A fundamental solution allowing for large file and directory uploads and downloads direct to the object storage, while maintaining high speed, security, and robustness is needed yet does not exist in the cloud storage platforms on their own. Aspera's Direct-to-Cloud transport capability has been engineered from the ground up as a fundamental solution and has expanded to support all of the major cloud storage platforms in commercial use.

A fundamental solution – IBM Aspera Direct-to-Cloud transport

The Aspera Direct-to-Cloud transport platform is a one-of-a-kind fundamental solution for transfer of file and directory data to, from and between cloud storage. Built on the FASP transport technology deeply integrated with object storage, it brings all of the characteristics of the Aspera transport platform to cloud storage: maximum speed of transfer for upload to cloud, download from cloud and inter-cloud transfers of files and directories regardless of network



THE SOLUTION
ASPERA ON DEMAND
- Full client-side r/w of object storage
- Synchronous transfer from Client to object storage (via Aspera on Demand)
- FASP transfer speeds end-to-end
- Real-time optimization of HTTP threads, chunk size, interfaced to FASP

Client Software
- Desktop client
- Web browser plug-in
- Embedded client
- Automatic download widget
- Mobile apps

PERFORMANCE
EFFECTIVE THROUGHPUT
- Up to 1Gbps per instance
- 10 TB per 24 hours

Figure 3: Aspera Direct-to-Cloud transport, a fundamental solution for large file and directory transfer with cloud object storage, providing native FASP transport capability end-to-end with deep integration to object storage

distance, in a single transport stream – no parallel streaming required, and support for files and directories up to the maximum size allowed by the storage platform1. Transfer rates adapt automatically to the available network bandwidth and storage bandwidth through Aspera's patented dynamic rate control and the aggregate bandwidth of multiple transfers is precisely controllable with Aspera's vlink technology. The platform addresses the fundamental security concerns around data in the cloud with both over-the-wire and at-rest encryption, and provides privacy in multi-tenant storage environments by authenticating all transfer and browsing operations using native storage credentials. Interrupted transfers automatically restart and resume from the point of interruption. Secure file browsing and transfer is supported with all Aspera clients, including browser, desktop, CLI and embedded / SDK modes.

Capability details are highlighted below.

- **Performance at any distance –** Maximum speed single stream transfer, independent of round-trip delay and packet loss (500 ms / 30% packet loss+) up to the I/O limits of the platform.
- **Unlimited throughput in scale out –** Automatic cluster scale out supports aggregate transfer throughputs for single mass uploads/downloads at 10 Gigabits per second and up, capable of 120 Terabytes transferred per day and more, at any global distance.
- **Large file sizes –** Support for files and directory sizes in a single transfer session up to the largest object size supported by the particular platform at a default 64 MB multi-part chunk size, e.g., 0.625 TB per single session on AWS S3. (The most recent software versions have a configurable chunk size extending transfers to the largest object size supported by the platform).
- **Large directories of small files –** Support for directories containing any number of individual files with high-speed, even for very large numbers of very small files (100 Mbps transfers over WAN for file sets of 1-10 KB in size), 500 Mbps+ with new ascp4).
- **Adaptive bandwidth control** – Network and disk based congestion control providing automatic adaptation of transmission speed to available network bandwidth and available I/O throughput to/from storage platform, to avoid congestion and overdrive.
- **Automatic resume –** Automatic retry and checkpoint resume of any transfer (single files and directories) from point of interruption.
- **Built-in encryption and encryption at rest –** Built in over-the-wire encryption and encryption-at-rest (AES 128) with secrets controlled on both client and server side.
- **Secure authentication and access control –** Built-in support for authenticated Aspera docroots implemented using private cloud credentials. Support for configurable read, write, and listing access per user account. Support for platform-specific role based access control including Amazon IAMS and Microsoft Secure SaaS URLs.
- **Seamless, full featured HTTP fallback –** Seamless fallback to HTTP(s) in restricted network environments with full support for encryption, encryption-at-rest and automatic retry and resume.
- **Concurrent transfer support –** Concurrent transfer support scaling up to ~50 concurrent transfers per VM instance on the environment. (Cloud storage platforms vary in their ability to support concurrent sessions depending on the maturity of the platform and the capacity of the particular VM host-to-cloud file system architecture).
- **Preservation of file attributes –** In later versions transfers can be configured to preserve file creation, modification times against AWS S3 and Swift.
- **Complete interoperability with Aspera Clients –** Fully interoperable transfer support with all core Aspera products acting as transfer peers with the cloud storage transfer.
- **Full-featured transfer modes –** Fully interoperable transfer support for all modes of transfer in these products including command line (CLI), interactive GUI point-and-click, browser, hot folder automation, and SDK automation.
- **Comprehensive server capabilities –** Full support for all Aspera server-side features including secure docroots, console configuration of BW, security and file handling policies and reporting to Aspera Console.
- **Support for forward and reverse proxy –** Transfers to/from cloud environments support Aspera proxy on the client side in forward or reverse mode.
- **Comprehensive SDK capabilities –** The server side software supports all of the core Aspera transfer and management SDKs including the Connect JavaScript API, faspmanager, SOAP and REST web services for job initiation, reliable query, aggregate reporting through stats collector, and automatic post-processing scripts.

Transfer Cluster Management with Autoscale

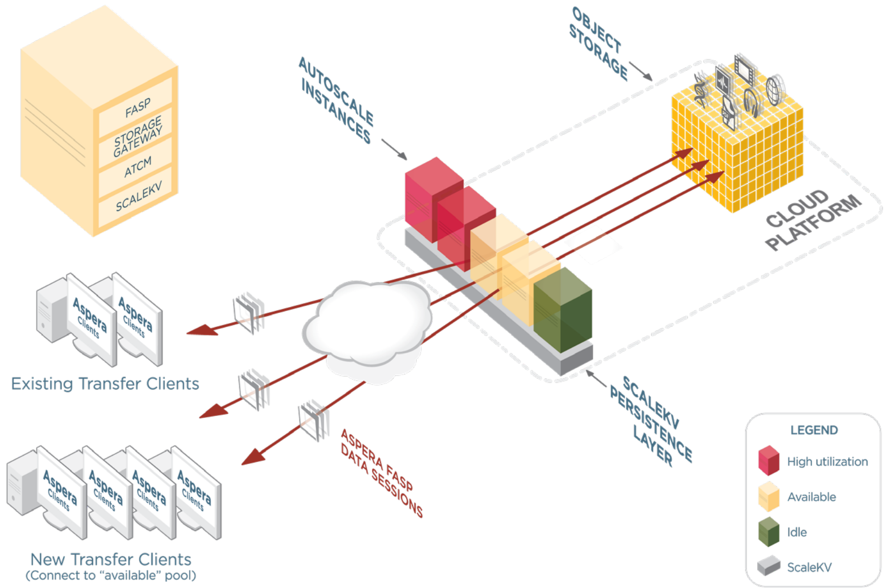
The new Transfer Cluster Manager with Autoscale elastic auto scaling of transfer hosts and client load balancing, cluster-wide reporting, and transfer management, and multi-tenant secure access key system. The service allows for dynamic, real-time scale out of transfer capacity with automatic start/stop of transfer server instances, automatic balancing of client requests across available instances and configurable service levels to manage maximum transfer load per instance, available idle instances for "burst" and automatic decommissioning of unused instances.

The ATCM service includes the following capabilities:

- **Manages Transfer Throughput SLAs and compute/ bandwidth costs with elastic scaling –** The service is part of the Aspera transfer server software stack, and automatically manages the number of server instances needed to support client transfer demands based on user-defined policies and automatically manages the number of nodes in use and booted up in reserve but idle.
- **Provides high availability and load balancing –** As transfer loads increase and decrease, nodes are moved from idle to available for client requests, and from available to highly utilized and back again based on user-defined load metrics such as tolerances for low and high transfer throughput and online burst capacity. If the minimum number of available nodes drops below the user-defined threshold, the cluster manager boots up new nodes automatically, and then brings them back down when they are no longer needed.
- **Provides increased and reliability –** ATCM will monitor the health and availability of Aspera transfer. Any unavailable/ down nodes or services are automatically detected and restarted or replaced if necessary. Subsequent client requests are pointed to healthy nodes via automatic cluster domain name services (DNS) management.
- **Works on all major clouds and in conjunction with Aspera Direct-to-Cloud storage infrastructure independent –** All of the Autoscale capabilities are implemented in the Aspera software and thus are portable across cloud providers including AWS, IBM Cloud, Azure, Google, etc. Works in both public clouds and Virtual Private Cloud (VPC) environments.

All transfer initiation and management scales across the cluster with ScaleKV technology:

- ScaleKV is a new Aspera created scale out data store for distributed, high throughput collection, aggregation and reporting of all transfer performance and file details across the auto-scale cluster, and supports transfer initiation across the cluster.
- A cluster-wide RESTful query API allows the cluster manager and 3rd party applications to query transfer progress and details across the entire cluster. The data structure shares the memory of all the nodes to automatically shard the transfer statistics data across the cluster, and allows gathering of transfer statistics at very high rates, ensuring the collection does not impede the transfer rates.
- The new cluster-wide RESTful transfer API allows third party applications to initiate transfers that use multiple / all nodes in the cluster for throughput beyond the capability of a single node, and automatic failover and fault tolerance.



- New mulit-tenant secure access key system allows Aspera applications such as Faspex and Shares and 3rd party applications to securely support multiple tenants on the same physical cluster with private access to separate cloud storage and private transfer reporting.
- Cluster owners may issue multiple tenant keys, and all application access with the transfer cluster is authenticated using the access key, and restricted to the corresponding cloud storage.
- Cluster REST API allows master node API credentials to query transfer status for all tenants, and individual applications to securely query their transfers only.
- New web-based cluster management UI manages access keys, cluster configuration including Autoscale policy and in memory data store for transfer statistics. The Autoscale platform software is built into the Aspera server software as of version 3.6 and is available for any cloud application using or integrating Aspera transfers.

Validation of third-party cloud storage platforms

In order to bring on support of a new object storage platform, and to verify support for a storage platform in our released software, Aspera carries out a comprehensive suite of automated and manual tests to verify performance with WAN conditions, large file sizes and numbers, file integrity, concurrency, load testing, security including encryption and access control, and backward compatibility between versions. Aspera aims to run the same test sets and conditions across all platforms within the limits of the number, variety and network connectivity of the test hosts the platform provides. The parameters of the test cases and performance capabilities for a single virtual host computer running the Aspera server software, by platform, are detailed in Table 1 on the following page.

Currently certified and supported platforms

As of version 3.7 of the Aspera core product online, Aspera is providing official support for the following cloud storage platforms in general release:

- IBM Cloud
- Amazon AWS S3
- OpenStack Swift version 1.12 and Up
- Microsoft Azure BLOB
- Akamai NetStorage
- Google Storage
- Limelight Object Storage
- HDFS
- HGST
- NetApp Object Storage

Conclusion

The majority of cloud-based storage available in the marketplace today is based on object storage. Key design principles of object storage architectures are the separation of file data and metadata, replication of data across distributed commodity storage, and unified access across distributed nodes and clusters. These principles enable more cost-effective scale-out with greater redundancy and durability then traditional block-based storage.

However, these same attributes create challenges for storing large unstructured data. Large files must be divided into “chunks” and stored independently on “writes”, and reassembled on “reads”. When coupled with the traditional bottleneck of moving large files over long-haul WAN connections with high round-trip time and packet loss, cloud-based object storage becomes impractical for large unstructured data due to the dramatic reduction in transfer speed and throughput, and extended delays in transferring and storing the files.

Aspera FASP high-speed transport platform is enabled to provide high-performance secure WAN transport of files, directories, and other large data sets to, from and between cloud storage. FASP overcomes the WAN data movement bottleneck while also maximizing and stabilizing the throughput to the underlying object storage. The deep integration with the object storage APIs delivers maximum transfer performance, while adding key transfer management features otherwise unavailable such as pause, resume and encryption over the wire and at rest. Through its design model and performance validation testing with leading cloud-based object storage platforms, we can see FASP as an ideal next-generation technology for reliable data transfer to, from, and across cloud storage.

Test Area	Test Type	Test Dimensions	Dimension Values	Platform Limitations
Cloud Storage	Load Test	File Size	400 GB	S3/SL Swift up to 5TB for single sessions Google max session 625 GB (with 10,000 parts" support from Google) Azure max session 1TB (requires page blob and gathering policy set to ‘auto’)
		Bandwidth	10Mbps, 100 Mbps, 500 Mbps, 1Gbps, 10 Gbps (ATCM cluster)	S3 max bandwidth 1 Gbps per instance with m3.xlarge, 2 Gbps per instance, with c3.xlarge, 10 Gbps with ATCM cluster Swift max bandwidth 800 Mbps per instance, 10 Gbps with ATCM cluster Azure max bandwidth 400 Mbps 400Mbps per instance
		Concurrency	2, 6 client; 8, 12, 25, 50 server	S3/SL Swift max 50 concurrent Azure max 10 concurrent Google max concurrent 10
		Encryption	On, Off	
		Direction	Up, Down, Mix	
		Data sets	Small files - 0 Byte to 100KB (420K Files) Medium Files - 1 MB to 100 MB (9K Files) Large Files - 1GB to 100GB (6 Files)	Google max file size 625GB (1TB with new experimental "10,000" parts" support available from Google)
		Transfer Policy	Low, Fair, High, Fixed	
	Stress Test	File Size	0 byte to 1 TB	S3/ SL Swift max file size 5 TB for single sessions Google max session 625 GB (with 10,000 parts" support from Google) Azure max session 1TB (requires page blob and gathering policy set to ‘auto’)
		Bandwidth	500 Mbps to 10 Gbps (ATCM Cluster)	S3 max bandwidth 1 Gbps per instance with m3.xlarge, 2 Gbps per instance, with c3.xlarge, 10 Gbps with ATCM cluster Swift max bandwidth 800 Mbps per instance, 10 Gbps with ATCM cluster Azure max bandwidth 400 Mbps 400Mbps per instance
		Concurrency	12,15,20	Azure max 10 concurrent Google max concurrent 10
		Encryption	On, Off	
		Direction	Up, Down, Mix	
		Data sets	Small files - 0 Byte to 100KB (420K Files) Medium Files - 1 MB to 100 MB (9K Files) Large Files - 1GB to 100GB (6 Files)	Google max file size 62GB (625GB with new experimental "10,000" parts" support available from Google)
		Transfer Policy	Low, Fail, High, Fixed	
	Backward Compatability Test	Product version	ES 3.6.0, ES 3.6.1	
	Soak Test	File Size	0 byte to 10 GB	S3/Swift up to 5TB for single sessions Google max session 62GB Azure max session 200GB
		Bandwidth	10Mbps, 300 Mbps	Azure max bandwidth 400Mbps
		Concurrency	4,6	
		Direction	Up, Down, Mix	
		Transfer Policy	Low, Fail, High, Fixed	
		Duration	100 hours	
	File Integrity Tests	File Size	10 byte, 4MB, 64MB, 100MB, 1 GB, 100 GB, 1 TB	
		Direction	Up,Down	
		Encryption	On, Off	
	System Tests	Products	Faspex, Console, Shares	
		File Size	10 byte to 1 GB (various/real-world)	
		Direction	Up, Down, Mix	

Table 1: Aspera Cloud Storage Verification Testing. Please note: All tests run against Aspera server software (version 3.6.0) on a single virtual machine host in the environment with capabilities comparable to EC2 m3.xlarge AOD with 16 GB Ram and S3 bucket in same region unless otherwise noted.

Test Area	Test Type	Test Dimensions	Dimension Values	Platform Limitations
Concurrency	Load Test, Stress Test, Spike Test, Soak Test	Load	Server: Concurrent sessions [1,10, 25, 35, 50] Client: Concurrent sessions [1, 10, 25]	S3/SL Swift max concurrent 50, Azure max concurrent 10, Google max concurrent 10, HGST max concurrent 25
		Direction	Up, Down, Mix	
		Fileset	3800 @ 0-10MB, 5000 @ 1mb	
		Bandwidth	[25%, 50%, 75%, 100%, 125%, 150%] (1GBps Capacity)	
		Duration	10min, 1hr, 8hr, 2day	
		Packet delay	0, 100ms	
		Encryption	On, Off	
		Resume	none, metadata, sparse, full	
		Operating systems		
		Traffic Spikes	35 - 50 concurrent sessions	

Test Area	Test Type	Test Dimensions	Dimension Values	Platform Limitations
WAN*	Performance Test	Bandwidth	512Kbps, 1Mbps, 10Mbps, 155Mbps, 622Mbps, 1Gbps, 3Gbps, 10Gbps	
		Round trip time	0ms, 2ms, 10ms, 100ms, 300ms, 500ms, 1000ms	
		Packet Loss Rate	0%, 0.1%, 1%, 5%, 10%, 20%	
		Mean File Size (data sets)	1KB, 10KB, 100KB, 1MB, 5MB (small media files), 10MB	
		Concurrency	1, 10 (higher concurrency will be tested in ssh load test)	
		Overdriving	2, 10 and 100	
		Encryption	Enabled and Disabled	
		Block Sizes - Read and Write	16KB, 64KB, 128KB, 256KB, 512KB, 1MB, 4MB	
		Router Buffersize (Queue Depth)	10ms, 100ms, 250ms	
		Direction	Upload, Download	
		Operating Systems	Major operating systems	
Security	Functional Test	Transfer encryption	On, Off (comprehensive use cases)	Executed in controlled lab environment
		EAR	Upload, Download, FASP/HTTP	
		File Checksum	MD5, SHA1, none	
		ssh fingerprint	ascp, HTTP fallback	Executed in controlled lab environment
		HTTP proxy	access control, token	
		DNAT proxy	configuration, concurrency (20), resume, http fallback	Executed in controlled lab environment
		Token authorization	Upload/Download, files, list and pair-list, FASP and HTTP fallback, token ciphers	
		HTTP fallback	token auth, forged requests	

* Not supported on all platforms and object stores.

Glossary of terms

Test Type	Definition
Load Test	Verify product behavior under targeted load conditions
Stress Test	Evaluate product behavior operating under loads significantly higher than targeted levels
Spike Test	Verify product behavior during repeated spikes of stress-level traffic
Soak Test	Verify product behavior operating at target load conditions for sustained periods
System Test	Validate feature functionality in integrated environment of Aspera products
Performance Test	Evaluate and measure file transfer performance with relation to indicated test dimensions in
File integrity test	Verify integrity of transferred file data with relation to indicated test dimensions
Functional Tests	Verify product functional behavior with relation to indicated feature dimensions
Backward Compatibility Test	Verify functional and non-functional product behavior against earlier product releases

About IBM Aspera

IBM Aspera offers next-generation transport technologies that move the world's data at maximum speed regardless of file size, transfer distance and network conditions. Based on its patented, Emmy® award-winning FASP® protocol, Aspera software fully utilizes existing infrastructures to deliver the fastest, most predictable file-transfer experience. Aspera's core technology delivers unprecedented control over bandwidth, complete security and uncompromising reliability. Organizations across a variety of industries on six continents rely on Aspera software for the business-critical transport of their digital assets.

For more information

On IBM Aspera solutions, please visit us at https://www.ibm.com/cloud/high-speed-data-transfer or contact aspera-sales@ibm.com.

IBM Corporation
Route 100
Somers, NY 10589

Produced in the United States of America
December 2018

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on the specific configurations and operating conditions. It is the user's responsibility to evaluate and verify the operation of any other products or programs with IBM product and programs. THE INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT. IBM products are warranted according to the terms and conditions of the agreements under which they are provided.

[1] The maximum object size supported using the default Aspera software configuration depends on the capabilities of the storage platform. Specific platform limitations are given in Table 1.

♻ Please Recycle