Global Profiling of Open Science Data Federation Services

using the Pelican Platform

XRootD Workshop 26

Introduction

- Contemporary research relies heavily on computational resources and storage, with data sharing serving as a critical element. Data access remains a central challenge.

- The Open Science Data Federation (OSDF) project aims to establish a global scientific data distribution network by leveraging the Pelican Platform and the National Research Platform (NRP)

Introduction

- OSDF is based on the XRootD and Pelican projects.

- The object is to create a set of test to check how is the performance “now” of the OSDF, creating a dataset enabling research to plan the job and resource for the OSPool and create a system to check this performance periodically (PerfHist).

Introduction - OSDF cache and origins

Introduction - OSG Sites

Methodology

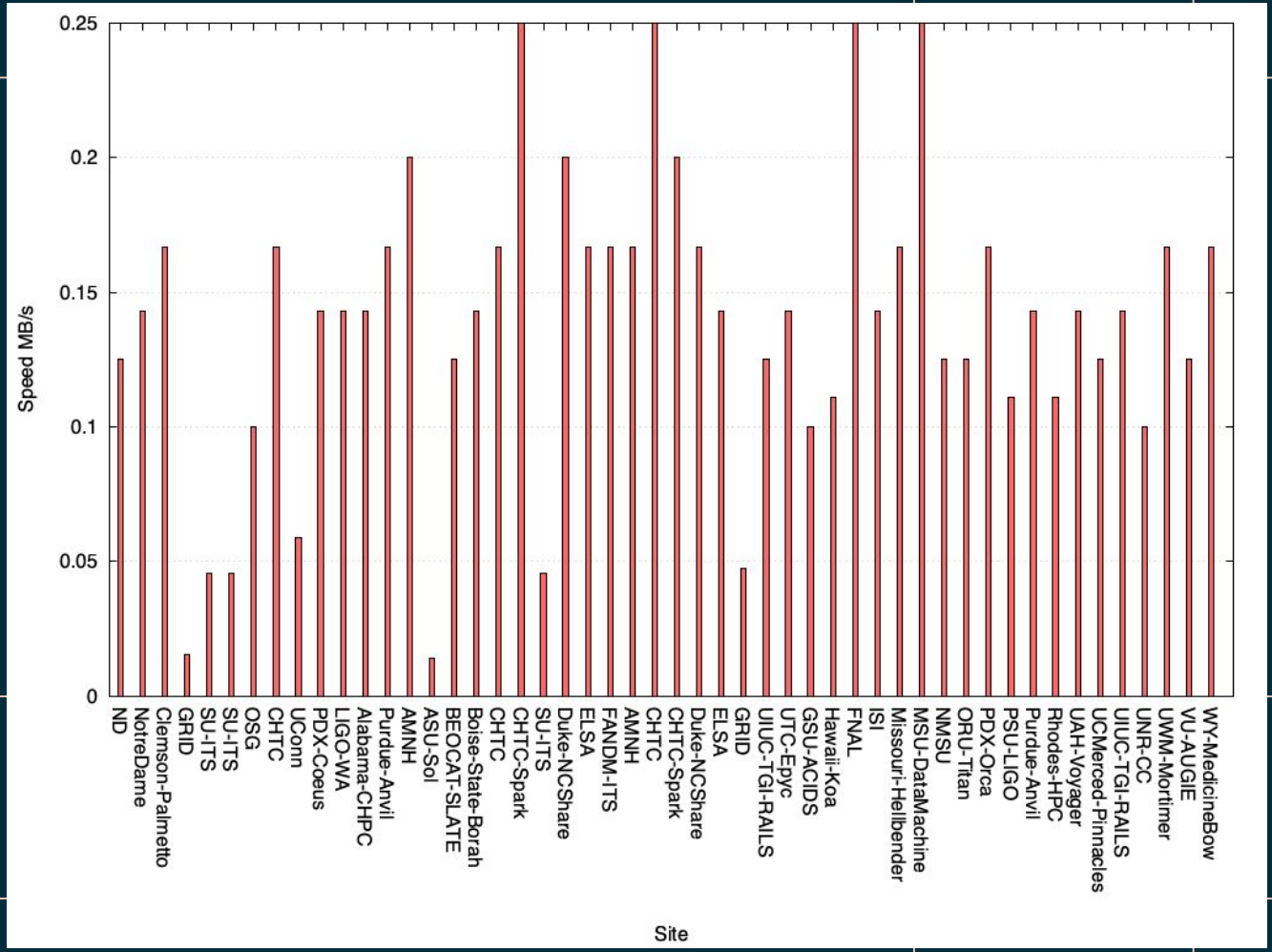
- List and check what sites are available in the OSPool;
- Run jobs copying files 1MB, 1GB, 10GB and 100GB:
  - Benchmarking the Open Science Data Federation services to develop XRootD best practices
    Fabio Andrijauskas, Igor Sfiligoi, Frank Würthwein
    EPJ Web Conf. 337 01256 (2025)
    DOI: 10.1051/epjconf/202533701256
- Use the pelican client and HTTP cliente (e.g. wget);
- Keep a history of the performance - Project PerfHist - https://github.com/biozit/PerfHistXrootD
-
- Track any error or messages;

https://github.com/biozit/PerfHistXrootD

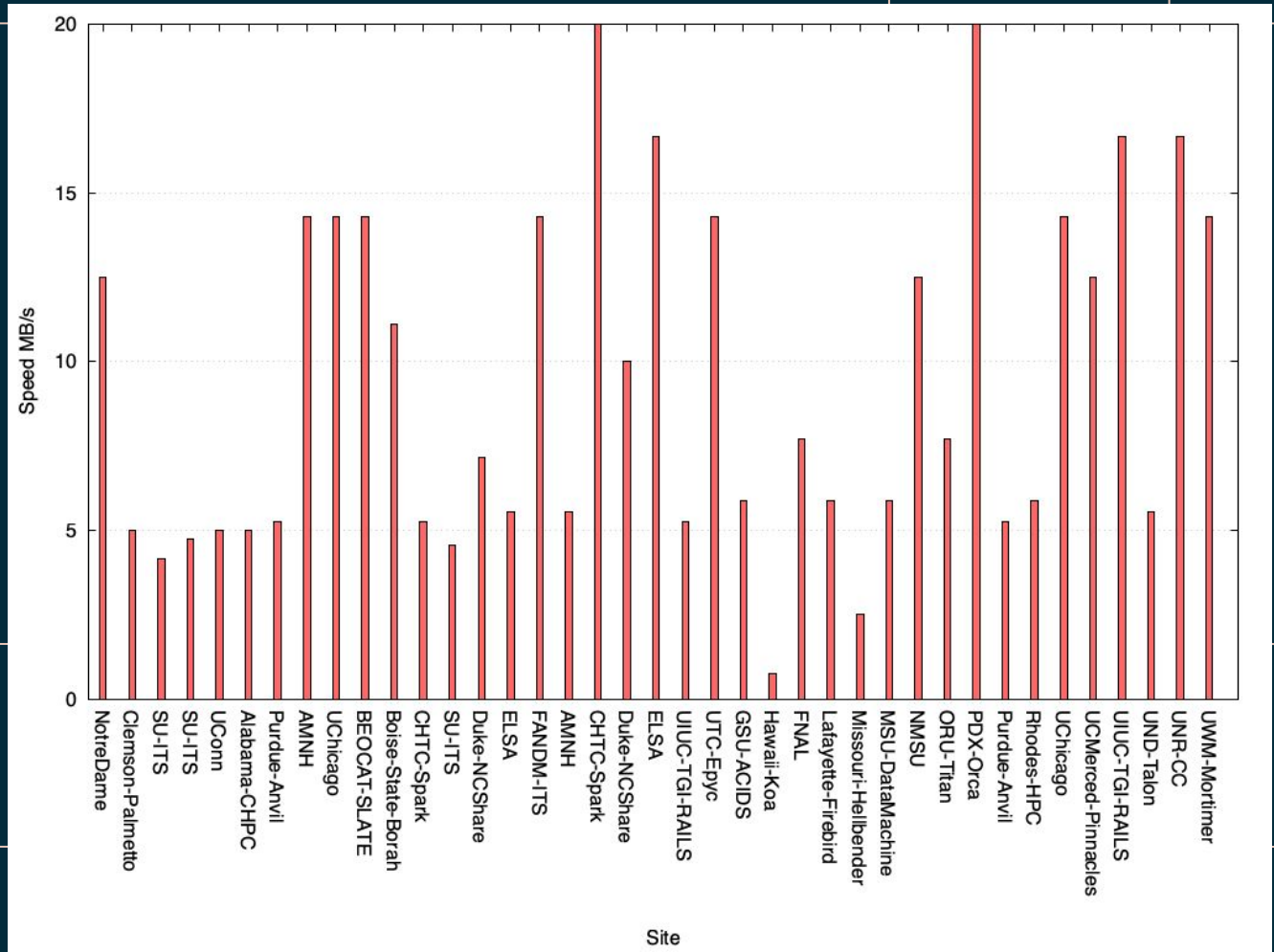
The performance test must follow these requirements:

- The host and client must have:
  - 4 cores using a server-type processor with a minimum clock speed of 2.5 GHz.
  - 4 GB of RAM per core.
  - Network of 10GB/s - max RTT of 2
- The tests must be performed using:
  - 10 times, using file sizes of 1MB, 10MB, 100MB, 1GB, and 10GB.
  - Using 1, 5, and 10 parallel requests.
- The results must be reported in CSV format.
- HTTP, HTTPS and xrootd protocols must be used

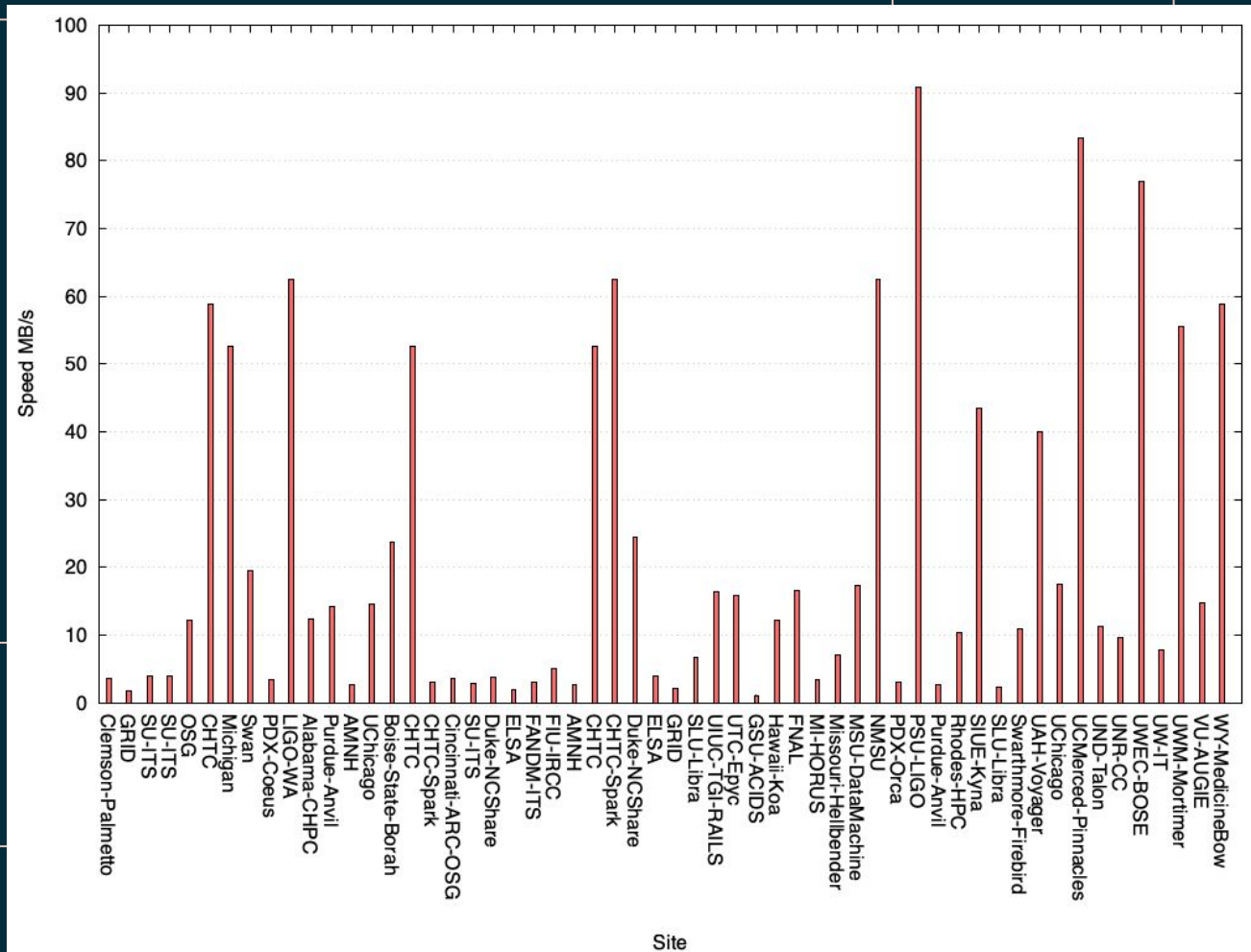
1MB files - OSPool



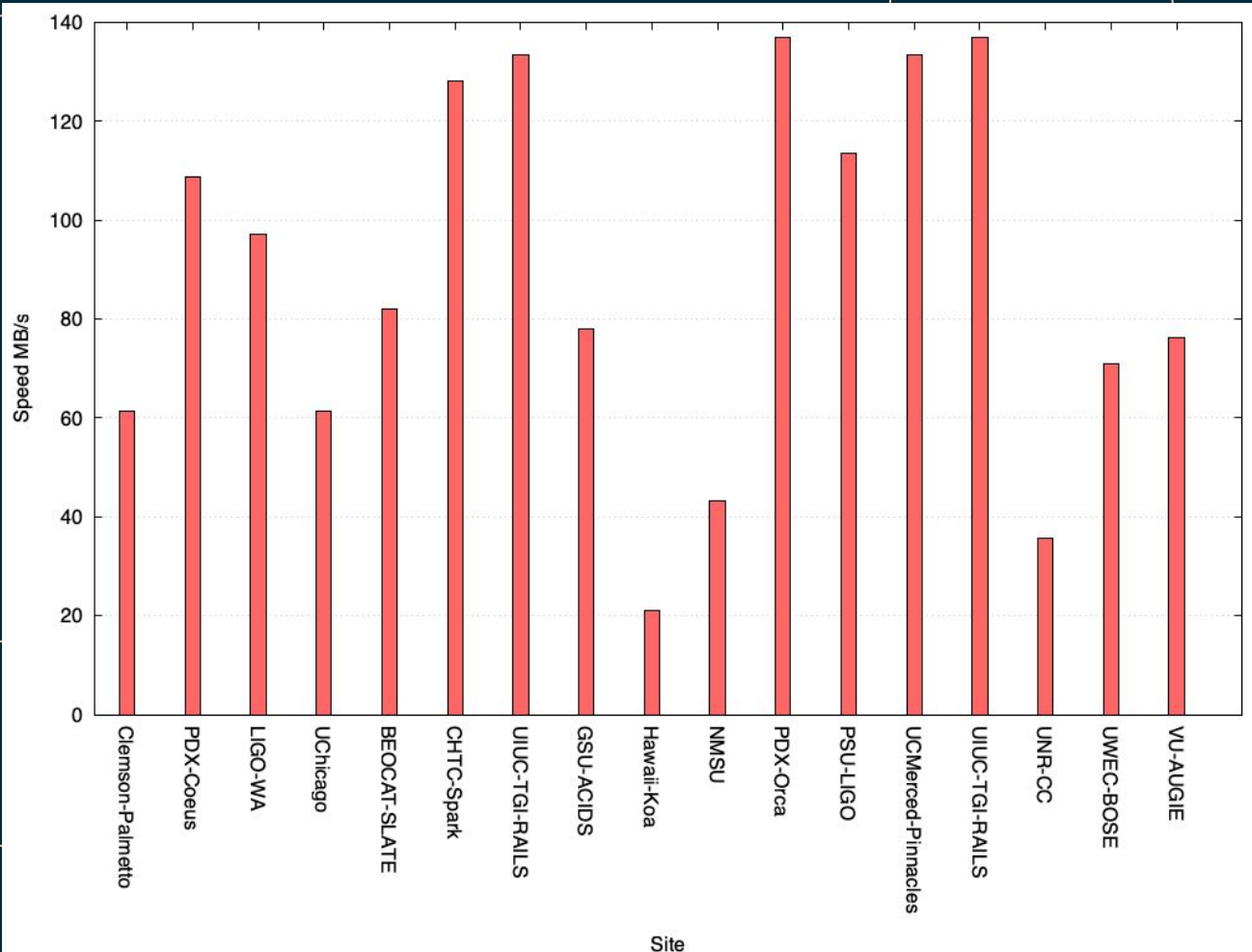
100MB files - OSPool



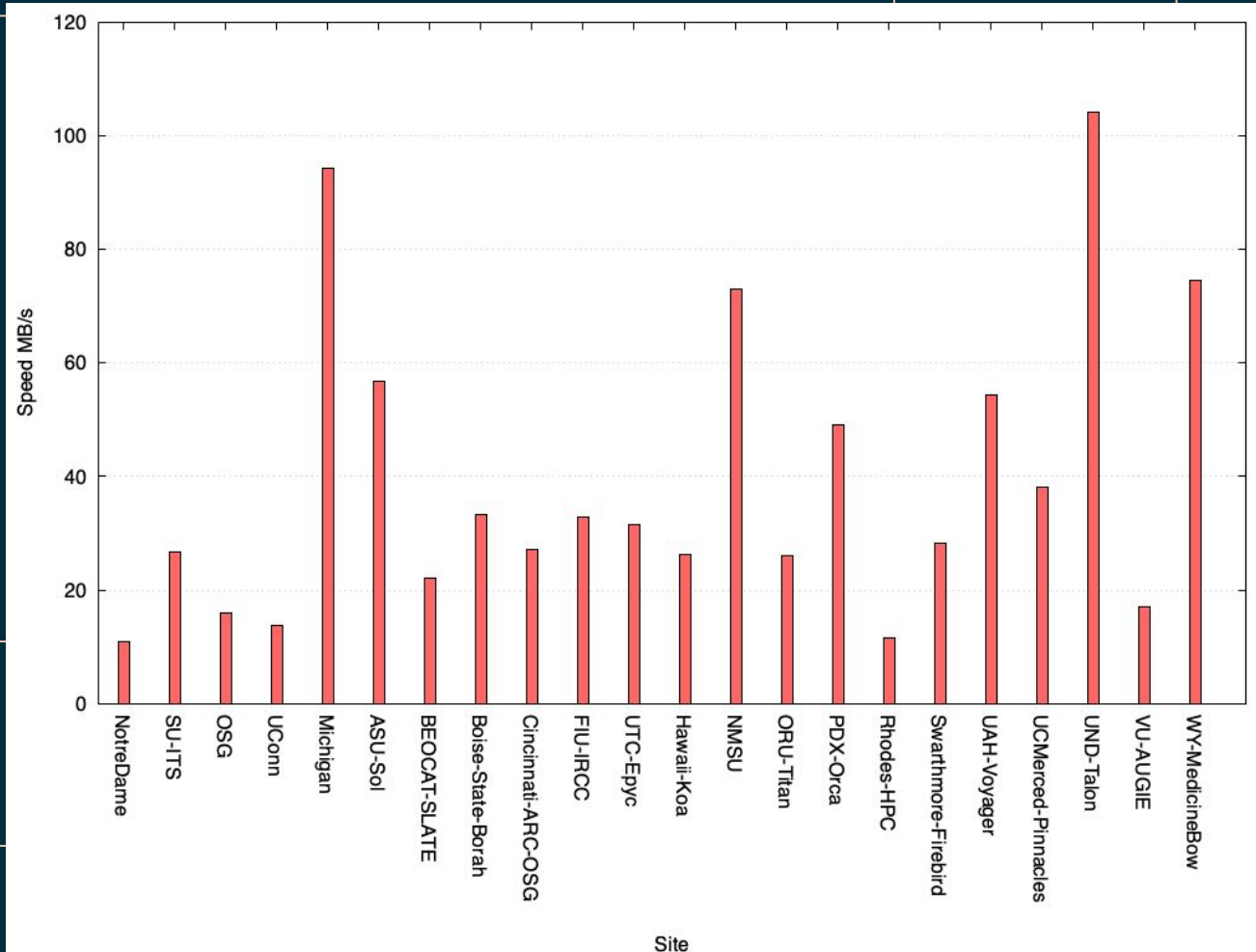
1GB files - OSPool

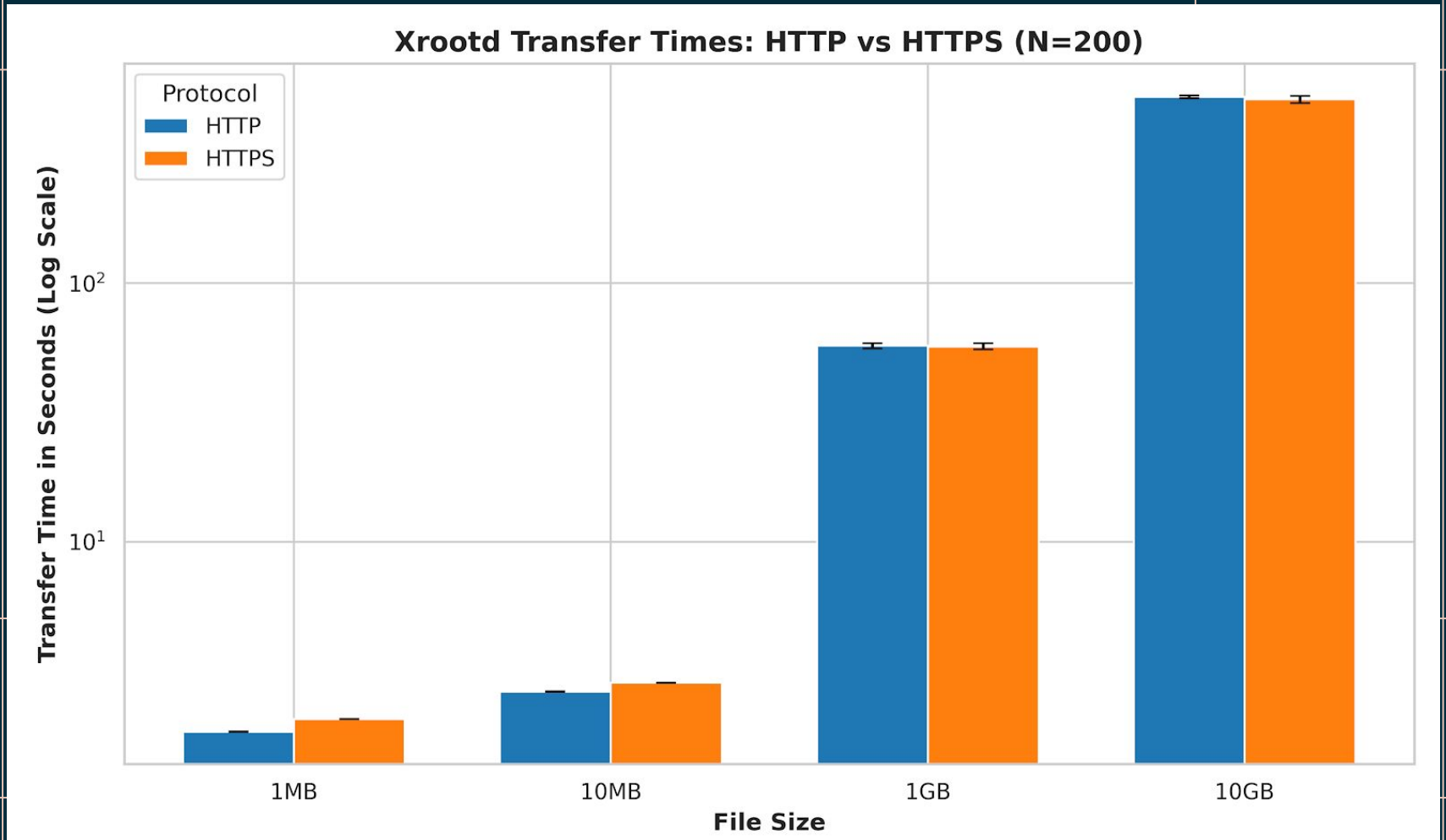


10GB files - OSPool



100GB files - OSPool




Xrootd Transfer Times: HTTP vs HTTPS (N=200)
Protocol
HTTP
HTTPS
Transfer Time in Seconds (Log Scale)
10^2
10^1
1MB
10MB
1GB
10GB
File Size

Small File Sizes (1MB, 10MB): HTTP shows a slight speed advantage over HTTPS, likely due to the absence of SSL/TLS handshake overhead.

Large File Sizes (1GB, 10GB): The performance difference between HTTP and HTTPS narrows significantly, with transfer times remaining nearly identical as bandwidth utilization dominates overhead costs

Conclusion and Recommendations

* Each OSPool site could already have the closest cache set, which could help accelerate cache selection in scenarios with multiple files.

* A cache could be set to “timeout” if performance or load exceeds a threshold.

* The timeout transfer of a file could be the maximum time the job can run.

* Some cache could be optimized to “larger” and “small” files

Conclusion and Recommendations

* If the file is not found, the job should fail, not be put on hold

* The cache selection worked without problem.

* The S3 plugin was tested too, and there are no improvements in speed; access to S3Plugin is still ~3x slower than a “direct” access.;

Conclusion and Recommendations

* Create a set of tests for speed and functionality before sending a version to production.

* The OSDF is still showing a better performance for larger files.