A world-wide databridge supported by a commercial cloud provider

Kwong Tat Cheung\textsuperscript{1}, Laurence Field\textsuperscript{2}, Fabrizio Furano\textsuperscript{2}

\textsuperscript{1} University of West England
\textsuperscript{1} CERN, Geneva, Switzerland

E-mail: Kwong2.Cheung@live.uwe.ac.uk

Abstract. Volunteer computing has the potential to provide significant additional computing capacity for the LHC experiments. One of the challenges with exploiting volunteer computing is to support a global community of volunteers that provides heterogeneous resources. However, high energy physics applications require more data input and output than the CPU intensive applications that are typically used by other volunteer computing projects. While the so-called databridge has already been successfully proposed as a method to span the untrusted and trusted domains of volunteer computing and Grid computing respective, globally transferring data between potentially poor-performing residential networks and CERN could be unreliable, leading to wasted resources usage. The expectation is that by placing a storage endpoint that is part of a wider, flexible geographical databridge deployment closer to the volunteers, the transfer success rate and the overall performance can be improved. This contribution investigates the provision of a globally distributed databridge implemented upon a commercial cloud provider.

1. Introduction

Volunteering computing \cite{5} enables spare computing capacity from the general public to be donated to science projects \cite{6}. These projects typically require many computationally intensive tasks to be executed to achieve their scientific goals, a standard high-throughput computing use case. While this approach has been proven by projects such as Seti@home \cite{4} and climateprediction.net \cite{7}, other projects require more data intensive tasks to be executed. With the increasing speed, performance a reliability of residential network connections, mainly driven by the streaming video use case, it is now possible to consider sending tasks with more challenging input/output requirements. But how big can files become before transfers start failing using an average residential network?

The bandwidth of residential networks will differ and in many cases the upload bandwidth is lower than the download bandwidth. How much data a volunteer can transfer within an acceptable time frame essentially creates a performance limitation. In addition, volunteers are from all over the world and hence can potentially be thousands of kilometres from the projects storage location. While loss due to interruptions during the computation phase can be mitigated to some extent by checkpointing \cite{3}, interruptions during a file transfer would most likely mean that the entire transfer has to be restarted, wasting valuable bandwidth and processing time.

The aim of this paper is to explore the possibility of optimizing file transfers in volunteer computing projects by offering a local upload endpoint. An asynchronous eventually consistent method can then be used to migrate those files to project servers or an alternative data
centre, where the validation and merge operations can take place. The paper is outlined as follows. Section 2 provides an overview of the limiting factors for file transfer performance and investigates the relationship between physical distance and file transfer performance. This includes a case study on a recent volunteer computing project, CMS@home [8], developed at CERN, the European Organization for Nuclear Research. These findings are then used to design a novel distributed storage infrastructure for volunteer computing, which is described in Section 3 along with an implementation based upon a commercial cloud provider. An evaluation is provided in Section 4 along with some concluding remarks in Section 5.

2. The effect of round-trip time on file transfers over HTTP

As the volunteers are dispersed worldwide, the physical distance between the project’s servers and a volunteer can potentially be very large (i.e. span half the globe). For volunteers far away from the servers, network latency cloud have an impact on transfer performance and reliability. Larger distances between the transport endpoints results in a higher round-trip times (RRT) and an increased chance of packet loss during transmission, thus lowering the transfer performance. For projects which require large input or output files, an average low upload speed and high latency will imposes a limit on the file size that can be adequately handled. Given a constant transfer speed, the larger the file, the longer it takes to complete the transfer, thus increasing the probability of interruptions.

To measure the effect of RTT on file transfers, seven identical virtual machines were provisioned at different global locations (North Virginia, North California, Oregon, Dublin, Tokyo, Sydney and Singapore). An Amazon Web Service (AWS) [1] S3 bucket in the Frankfurt region was used as the destination. Each of the instances performed sequential file transfers over HTTP with file size from 50MB to 300MB in 50MB increments. The results are shown in Figure 1, which compares the average transfer time to the average RTT.

![Figure 1. Average file transfer time compared with the average RTT or different file sizes](image)

It is evident that even for the same file size, transfers from instances where a higher latency was recorded took a considerably longer time to complete. This issue is further compounded by
the asymmetrical bandwidth offered by ISPs in residential networks, where upload bandwidth is often many times lower than download bandwidth, as well as poor network infrastructure in some areas.

The combined effect of these factors are challenging for volunteer computing projects that produce large outputs. One such project is CMS@home, which runs event simulations for the CMS experiment based at CERN. The effect of large distances on file transfer performance and the stage-out failures experienced in CMS@home were analysed to understand if there is a correlation between distance and file transfer failure. The log file of the upload server records an exit status code for the transfer and the IP address of the client initiating the transfer. This log file was analysed to understand the reliability of stage-out transfers for CMS@home during a two week period. Figure 2 shows each exit code as a percent for recorded for upload attempts from volunteers against the CEPH [2] S3 storage instance at CERN, grouped by their physical distance from CERN.

![Figure 2. Transfer exit codes from the CMS@home storage endpoint](image)

It can be seen that the network related error codes (0, 400, 403) occurred more frequently as the physical distance from CERN increases. This suggests that in a globally distributed volunteer computing infrastructure with large output file sizes, the distance between volunteers and the storage endpoint could be a significant factor that affects the success rate of file transfers.

3. Using a commercial cloud to provide local upload endpoints
The objective of this design is to optimise the global transfer of files by offering the volunteers a closer upload endpoint to an asynchronous eventually consistent method for migrating those files to CERN, or to an alternative data centre where the validation and merge operations can take place. This has the effect of reducing latency between the volunteers machine and the upload endpoint and hence increases file transfer performance and reliability. The location of AWS regions are shown in Figure 3 and offers a good match with areas identified where there are a high number of volunteers. In practice, storage systems from other providers could also be federated into the system.

A core component of the design is the DataBridge [8] which federates the upload endpoints, and redirects an upload request to an endpoint that is closest to the volunteer. The Uniform
Generic Redirector of the Dynafed [9], which is used to implement the DataBridge, provides the redirection for HTTP GET/PUT requests. This enables the administration credentials for the S3 service to be hidden from the volunteer by providing a redirect to a signed URL. The redirection can consider the geographical location of the request based on the IP address and hence select the nearest upload endpoint. A high-level view of the design is shown in Figure 4, with the flow of events summarised as follows:

- The volunteer uploads the file via a HTTP PUT request to the DataBridge
- The DataBridge redirects upload requests to the S3 bucket closest to the volunteer
- Files uploaded to S3 are replicated to the AWS region closest to the project data storage
- Files are replicated to the project's data storage from this close data store

Figure 3. AWS regions

The advantages of this design is that the latency between the volunteer's machine and the upload endpoint can be reduced, as well as transferring scalability and operational responsibility of the upload endpoints to the commercial provider. The approach is also designed to be highly extensible. As providers expand their global footprint, more locations can be federated into the system.

4. Evaluation
To evaluate the new design, a DataBridge service was provisioned in Frankfurt and configured to federate ten S3 buckets (one in each AWS region). A virtual machine was provisioned in Tokyo to act as a volunteer which attempted to transfer files from 50MB to 300MB in 50MB increments over HTTP, which is representative of the CMS@home use case. In the first test, the transfers were targeted directly at an S3 bucket in the Frankfurt region. In the second transfer, transfers were targeted at the DataBridge, with the requests redirected towards an S3 bucket closest to Tokyo (in this case, a bucket in the Seoul region). The result is shown in Figure 5, which compared the average file transfer time to file size.

The results show a significant improvement in transfer time when the transfers were redirected towards an endpoint closer to the Tokyo instance.
Figure 4. High-level overview of the distributed storage infrastructure

Figure 5. Average transfer time compared to file size

To investigate the reliability of file transfers, the system was also tested under poor network conditions. Squid proxy servers were set up close to the target S3 buckets in Sydney and Ireland to simulate this. They were configured to randomly drop 2% of incoming TCP packets using iptables. File transfers were then initiated with file sizes between 50MB to 200MB in
Figure 6. Impact of packet loss on transfer performance (direct)

Figure 7. Impact of packet loss on transfer performance (redirected)

50MB increments, with upload bandwidth capped at 5Mbps. The comparison of the average file transfer time and file size for 0% and 2% packet loss are shown for the direct and redirected scenarios in Figures 6 and 7 respectively.

The results show that with no packet loss, the file transfer time between the two endpoints are comparable. However when a 2% packet loss was introduced, uploads targeted at the Ireland
S3 bucket only experienced an increase in average transfer time of a few percent, whilst the file uploads targeted at the Sydney S3 bucket doubled for some file sizes. This suggests that the system is more resilient to performance penalties on the file uploads caused by poor network conditions and that it should provide more reliable transfers for volunteers in areas that have potentially poor-performing residential networks.

5. Conclusion
In this paper the consequences of expanding volunteer computing to more data intensive tasks was considered. An investigation into the effect of round-trip time on file transfers showed that transfer errors do increase with network latency and hence physical distance between the volunteer and the project’s storage system. Building upon existing solutions for data federations, it was suggested that a distributed storage federation that provided geographical load balancing could be used to provide the volunteers with upload endpoint that were close to them. A prototype of this design was implemented using the AWS to simulate a scenario where the project’s storage was located in Frankfurt and a volunteer based in Tokyo. It was shown that providing a close upload endpoint does improve the transfer time and makes them more resilient to failure.

References
[1] Amazon Web Services (AWS). url = https://aws.amazon.com/.
[2] Ceph. url = http://ceph.com/.
[3] David Anderson, Carl Christensen, and Bruce Allen. Designing a Runtime System for Volunteer Computing. pages 33–33. IEEE, November 2006.
[4] David P. Anderson, Jeff Cobb, Eric Korpela, Matt Lebofsky, and Dan Werthimer. SETI@home: an experiment in public-resource computing. Communications of the ACM, 45(11):56–61, November 2002.
[5] D.P. Anderson and G. Fedak. The Computational and Storage Potential of Volunteer Computing. pages 73–80. IEEE, 2006.
[6] D.P. Anderson and G. Fedak. The Computational and Storage Potential of Volunteer Computing. pages 73–80. IEEE, 2006.
[7] C. Christensen, T. Aina, and D. Stainforth. The Challenge of Volunteer Computing with Lengthy Climate Model Simulations. pages 8–15. IEEE, 2005.
[8] L Field, H Borras, D Spiga, and H Riahi. CMS@home: Enabling Volunteer Computing Usage for CMS. Journal of Physics: Conference Series, 664(2):022017, December 2015.
[9] Fabrizio Furano, Oliver Keeble, and Laurence Field. Dynamic federation of grid and cloud storage. Physics of Particles and Nuclei Letters, 13(5):629–633, September 2016.