Task 3 - Internal storage support

Hongyi Zhang <hongyiz@kth.se> Lida Liu <lidal@kth.se>

I. Access Support to External Storage

In Google Cloud Platform, we can use the Google Cloud Storage FUSE tool to mount a Cloud Storage bucket to your Compute Engine instance. The mounted bucket behaves similarly to a persistent disk even though Cloud Storage buckets are object storage.

Cloud Storage FUSE works by translating object storage names into a file and directory system, interpreting the ”/” character in object names as a directory separator so that objects with the same common prefix are treated as files in the same directory. Applications can interact with the mounted bucket like a simple file system, providing virtually limitless file storage running in the cloud.

While Cloud Storage FUSE has a file system interface, it is not like an NFS or CIFS file system on the backend. Cloud Storage FUSE retains the same fundamental characteristics of Cloud Storage, preserving the scalability of Cloud Storage in terms of size and aggregate performance while maintaining the same latency and single object performance. As with the other access methods, Cloud Storage does not support concurrency and locking. For example, if multiple Cloud Storage FUSE clients are writing to the same file, the last flush wins.

In our VM, we need to set up the environment and install the GCSFuse package. Adding the gcsfuse distribution URL as a package source and import its public key. Then we can update the list of packages available and install gcsfuse.

The example command to mount the storage bucket is followed:

$ gcsfuse example-bucket /path/to/mount

$ ls /path/to/mount

And also another way to have access to the GCS is treating the VM as a regular terminal. In this scenario, you need to set up the API Access environment before you read and write data into the storage bucket.

II. Benchmark Access to External Storage

The code for this section can be found on Github https://github.com/Mr-Hongyi/ID2210-Cloud_Project.git

Similar to task 1, we are also using 10 files to complete the experiment. The whole file size is 557.47 mb in total. During the single transfer, we moving the file in and out from the mounting storage bucket. The following figures show the results of the operation.

We can see that the average Reading throughput for single transfer is 44.17mb/s while Writing throughput is 14.96mb/s. And the maximum for reading is 59.97mb/s and for writing is 20.2mb/s. The reading speed is much faster than writing.

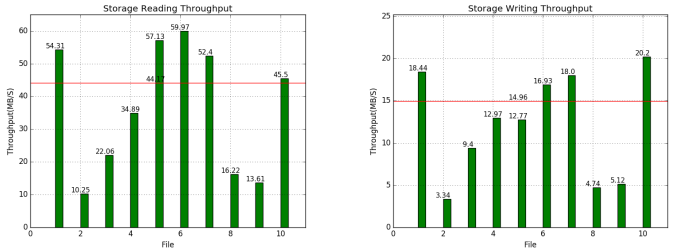

Fig. 1. R/W in Single Transmission

While in the parallel test, we are moving two sets of identical files also 557.47mb for each. We have logged into the VM by using two independent laptop. The following are the result figures.

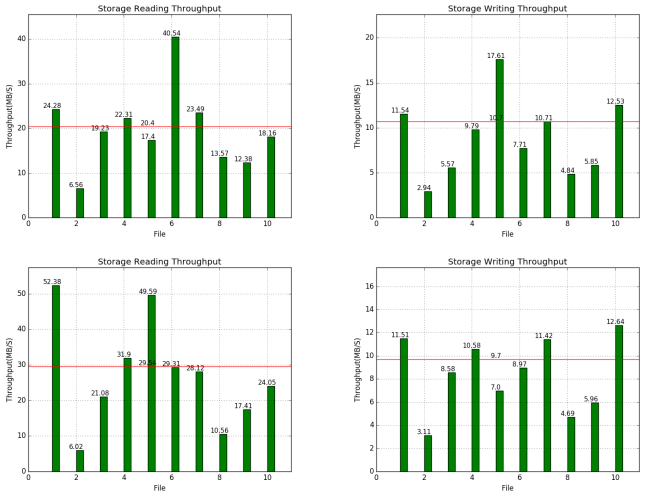

Fig. 2. R/W in Parallel Transmission(Statistics on two independable client)

As we can see the average reading speed for each are 20.4mb/s and 29.54mb/s, while the average writing speed are 10.7mb/s and 9.7mb/s. Addition of two sets of data is almost equal compared with the previous single transfer. In other words, the synchronous transmission will also share the data flow bandwidth just like the experiment in task1.

III. Benchmark Access to Local File System(SSD)

In this section, we have run a simple script which could benchmark the disk performance. The following figure is the result output from that code.

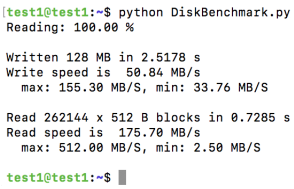

Fig. 3. Benchmark SSD

As we can see, the reading speed of local SSD is almost three times than the writing speed. Besides that, the internal storage R/W speed is almost twice as the external one. That is to say, in terms of processing data, we should choose more internal storage services.

References

[1] Connecting to Cloud Storage buckets, Google Cloud Platform Document [Online]. Available: *https://cloud.google.com/compute/docs/disks/gcs-buckets*

[2] Cloud Storage FUSE, Google Cloud Platform Document [Online]. Available: *https://cloud.google.com/storage/docs/gcs-fuse#using_feat_name*