OpenStack SWIFT Object Storage Tape Library Connector

What is it?

Modern cloud storage architectures like OpenStack SWIFT Object Storage do not natively support Tape Storage Media and Tape Libraries as storage targets.

This SWIFT-TLC software solves this limitation. SWIFT-TLC allows using a tape library as storage target for an SWIFT object storage node. It is possible to extend existing SWIFT cluster by adding tape storage or building new SWIFT tape based object storages as well.

The aim of the TLC is to make all specific behavior of a tape library completely transparent to SWIFT (that's why we call it connector). This ensures that there are no SWIFT modifications necessary to add tape storage to a SWIFT Cluster. The SWIFT TLC storage node uses exactly the same interface (Rest API) as a standard SWIFT storage node. Just the response time to retrieve single objects could be longer (occasionally up 300s).

All SWIFT features like ring controlled data placement, SWIFT policies, replication, auditing are fully supported.

The heart of the SWIFT TLC software builds a special Virtual File System (VFS) which represents the file interface between SWIFT and the tape library. TLC uses the LTFS format to store objects on tape. TLC formats every tape inside a tape library with LTFS and builds a large VFS out of the LTFS files system of each cartridge. For example if you have a tape library with 10 LTO6 tapes (2,5TB uncompressed capacity) TLC will build one large root file system with a capacity of 25TB out of this 10 tapes. This VFS will be mounted to the SWIFT storage node and can be used by SWIFT by adding this target to the SWIFT ring definition files.

Further the TLC software takes care of controlling the complete tape library. TLC controls which data gets written to which tape and controls loading the right tape in to a tape drive to retrieve objects from tape.

For more background information about the architecture of the SWIFT TLC software please refer the separate SWIFT TLC technical overview document.

Although the TLC auditor is completely transparent to SWIFT and uses the same interface to SWIFT as a disk based storage node TLC uses internally a different algorithm to verify objects on tape. Instead of verifying every object on tape separately (like SWIFT auditor does) TLC verifies a complete tape at once, which ensures high speed and optimal efficiency. For more background information about the TLC specific tape auditor please refer the separate SWIFT TLC auditor overview document.

Use Cases

- Grow existing SWIFT cluster by adding price attractive tape storage

  If an existing SWIFT cluster is running out of space and needs to be increased by adding more disk, TLC could be used to increase the object storage capacity by less expensive tape storage instead. This allows significant cost savings for storage hardware and operational costs if the majority of the data growth comes from data which gets accessed infrequently. TLC enables the integration of tape storage totally transparent for SWIFT. TLC does not require any modification on SWIFT. So this can be done very easily.

  The application of SWIFT storage policies allow further to control which data will be stored on disk storage and/or tape storage, which can be totally transparent to the end user as well.

Further there is also a smooth online migration of existing data to/from tape storage possible without any operational interruption. E.g. if there are old project data which don't get accessed frequently anymore they could be easily moved to tape. And if they need more frequent access again they could be moved back to faster storage nodes from tape.

- Tape Object Storage

  SWIFT TLC enables to build easily nonproprietary tape based object storages based on OpenStack SWIFT. By connecting several TLC nodes to one SWIFT cluster you can build reliable and much cost attractive object storage archives which suit very well for very large amount of data (easily up to several PB of data) which tolerate a retrieval time of a few seconds for objects which have not been accessed quite a while

Limitations and Future plans

We believe we developed a good starting point and we did all of the tape library specific SCSI control implementation work. This is where we have many years of development experience in.

Our target for developing this SWIFT TLC software was to make it as easy as possible for the OpenStack SWIFT development community to support tape storage with all the nice existing and upcoming SWIFT features.

This is why decided to make this software Open Source as we hope other will join this project and make it better over time.

Although we learned a lot about SWIFT during the last months SWIFT is still quite new for us and we are no experts for all the SWIFT internals. We believe there are a lot of SWIFT specific enhancements possible to optimize TLC for SWIFT. We hope that some SWIFT specialists gets interested in this subject and want to contribute with their ideas and experience to make TLC better for SWIFT. E.g. one strategic architectural question would be what the optimal capacity of a tape storage node is. Today we build one VFS out of all tapes from the tape library. May be it is better to have several smaller VFS containing of a few tapes only.

Today we don't have a dedicated recovery mechanism for a SWIFT TLC node yet. So if the TLC crashes completely we would setup a new fresh node and let SWIFT rebuild the complete content by replicating object copies from other nodes (which is the standard SWIFT behavior). As the capacity of a tape library could be very big, this could be very time consuming. This replication is actually not necessary as all the data are still available on tape. Each tape has its own LTFS file system which ensures that all data could individually read back without SWIFT or TLC involved. So we plan to write TLC VFS metadata to the LTFS file system on each tape as well and implement a TLC recovery mechanism, to restore the complete VFS metadata based on the recovery information stored on tape. This will allow setting up anywhere a fresh TLC node and the restore of the original VFS can happen just by inserting the original tapes into the tape library. This mechanism could be very valuable also for moving large amount of data between SWIFT regions. Instead of letting the content replicated by SWIFT over the network it would be possible to just send the tapes from one tape library to another region and restore the VFS on a TLC node there. This could save a lot of time.

We did our best to test TLC as we could. Nevertheless we have not had a chance to verify the functionality in very large multi-regional SWIFT clusters with a lot of storage nodes. We hope others become interested and help with running our software in such environments.

The biggest challenge for TLC is to support fast read response times as this is limited by the nature of tape. Tape Storage is a sequential media which requires that the right tape needs to be loaded mechanically into a tape drive and the tape drive need to seek the right position to be able to read the data back. All this takes time and could take worst case a few minutes. Our TLC uses special caching algorithm to mitigate this, but nevertheless whenever data do not reside in cache it will take a while to get them back from tape. This needs to be understood and considered by every solution architect.

Get involved

We hope by offering our SWIFT TLC as Open Source software that many will use it and native nonproprietary tape storage support becomes a part of the OpenStack ecosystem.

Everybody is highly invited to participate in this project and can join the development team.

We hope there will be interested guys who could help with the following subjects e.g.:

- Evaluate this software and provide feedback how it runs in their environment and sharing feedback what they like and what is missing
- SWIFT specialists interested in reviewing with us the TLC internal architecture
- discuss and evaluate potential use cases
- perform code reviews of TLC
- add new features
- help to get the OpenStack community aware of the opportunity using Tape Storage for OpenStack
- help to create case studies

Get started

- Evaluation
    - Download the installer packages (*.gz) from folder /. These packages can be installed on CentOS
    - Follow the QuickStart guide (SWIFT_TLC_QuickStart.docx) to get it running
- Source Code
    - Folder /TLC_ALL/Server contains all source code files
    - Run the script build_package.sh to build the binaries
- More Information
    - If you are interested in more detailed technical information please refer these documents:
        - SWIFT Technical Overview document
        - SWIFT Tape Auditor Overview Document