# IBM TS7700 R5.4 DS8000 Object Store User's Guide

Dave Brettell

Rin Fujiwara

Lourie Goodall

**Cloud**

**Storage**

IBM

**Red**paper

IBM Redbooks

**IBM TS7700 R5.4 DS8000 Object Store User's Guide**

October 2024

**Note:** Before using this information and the product it supports, read the information in "Notices" on page vii.

**Fourth Edition (October 2024)**

This edition applies to IBM TS7700 Version 5, Release 4, and the current DS8000 supported with this solution.

This document was created or updated on October 3, 2024.

# Contents

# Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:
*IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

| | | |
|---|---|---|
| Db2® | IBM® | Redbooks (logo) ® |
| DS8000® | IBM Cloud® | z/OS® |
| FICON® | IBM Z® | z15® |
| FlashCopy® | Power9® | |
| HyperSwap® | Redbooks® | |

Other company, product, or service names may be trademarks or service marks of others.

# Preface

The IBM TS7700 features a functional enhancement that allows for the TS7700 to act as an object store for IBM DS8000® Transparent Cloud Tiering with IBM DS8000, DFSMShsm (HSM), and native DFSMSdss (DSS). This function can be used to move data sets directly from DS8000 to TS7700. This IBM Redpaper publication provides a functional overview of the features, provides client value information, and walks through DFSMS, DS8000, and TS7700 set up steps.

This IBM Redpaper publication contains the following chapters:

# Authors

This paper was produced by TS7700 Development working with the IBM Redbooks, Tucson Center.

**Dave Brettell** is an Advisory Engineer with IBM's Storage Infrastructure, based out of the IBM laboratory in Tucson, Arizona, US. He is a Test Engineer for the IBM TS7700 Virtualization Engine. David has worked in mainframe back-end tape and virtual tape storage for over 20 years. He has held various rolls while under the product umbrella, including, Technical Writer, Hardware Microcode Integration Engineer, and Early Verification Test Engineer.

**Rin Fujiwara** works in the technical support team for DFSMS and high-end tape products in Japan. She has 15 years of experience in the IBM Z® area as a Subject Matter Expert. Rin has supported DFSMS and TS7700 for 5 years.

**Lourie Goodall** is a Senior Technical Staff Member (STSM) with IBM's Storage Infrastructure, based out of the IBM laboratory in Tucson, Arizona, US. She is the IBM Enterprise TCT Architect for the IBM TS7700 Virtualization Engine and DS8000 storage products. Lourie has worked in mainframe virtual tape and cloud storage for over 21 years and has authored or co-authored hundreds of design documents and numerous technical papers. Lourie is one of the original architects of the DS8000-TS7700 TCT feature.

Thanks to the following people for their contributions to this project:

# Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an IBM Redbooks residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

**ibm.com**/redbooks/residencies.html

# Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this paper or other IBM Redbooks publications in one of the following ways:

► Use the online **Contact us** review Redbooks form found at:

   **ibm.com**/redbooks

► Send your comments in an email to:

   redbooks@us.ibm.com

► Mail your comments to:

   IBM Corporation, IBM Redbooks
   Dept. HYTD Mail Station P099
   2455 South Road
   Poughkeepsie, NY 12601-5400

# Stay connected to IBM Redbooks

- Find us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm

- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

# Summary of changes

This section describes the technical changes in this edition of the paper and in previous editions. This edition might also include minor corrections and editorial changes that are not identified.

Summary of Changes for IBM TS7700 R5.4 DS8000 Object Store User's Guide
as created or updated on October 3, 2024.

## October 2024, Fourth Edition

This revision includes the following new information.

### New information
- ► Expand on network configuration rules and requirements.
- ► Add information about how **mkcloudserver** distributes TS7700 IP Addresses between the two DS8000 CECs.

### Changed information

General improvements in consistency, organization, and flow of the content.

## August 2023, Third Edition

For the Third Edition, the information in this revision of the Redpaper was reorganized from sections in previous editions to chapters. These changes have resulted in a reorganization of the information presented. Significant changes were made to bring the document up to date with current features, product names and terminology changes.

### TS7700 Advanced Object Store
This revision primarily covers the support for the TS7700 Advanced Object Store feature. Information about this new enhancement is summarized in the following list:

- ► Introduction to TS7700 Advanced Object Store (FC 5283)
- ► Hardware requirements
- ► Object Store policy management including synchronous and asynchronous replication of objects
- ► Cloud name to object policy association
- ► Grid awareness of objects
- ► Configuration and setup changes
- ► Management interface Dual Control support for object policy and object store settings
- ► DS8000 to TS7700 connection changes
- ► Understanding objects, containers and object tokens
- ► Object copy processing

- ► Throttling of object workload
- ► Library Request Command changes
- ► BVIR data additions
- ► Events and notifications
- ► Removing FC 5282
- ► FC 5282 to FC 5283 migration services
- ► Object migration services by copy refresh

### DS8000 Enhancements
- ► DS8000 Multi-Cloud support

# March 2021, Second Edition

This revision includes the following new information.

### New information
- ► IBM DS8000 Transparent Cloud Tiering object encryption support for the TS7700
- ► DS8000 Transparent Cloud Tiering compression support for the TS7700
- ► DFSMSdss full volume dump support for DS8000 Transparent Cloud Tiering

# Overview

This chapter provides an overview of the DS8000 Transparent Cloud Tiering solution and an introduction to the TS7700 Object Store and TS7700 Advanced Object Store Feature Code (FC) 5283 solutions.

The information that is provided throughout this document assumes that the reader is familiar with the TS7700, DS8000, and IBM Z host systems.

This chapter includes the following sections:

# 1.1 Introduction to transparent cloud tiering

Today, an IBM z/OS host works with many different applications to offload data from primary storage. Two predominant applications are IBM DFSMShsm and DFSMSdss, which are primarily used with IBM DS8000 disk storage as the primary storage system.

After the host writes to its primary disk cache system, it can choose to migrate or back up the data to a long-term storage system. This storage system can be a physical tape library or a virtual tape library, such as the IBM TS7700. Backing up to tape or virtual tape provides cost benefits greater than those provided by regular disk-based storage systems and is part of an efficient long-term storage strategy.

Since its origin, the TS7700 was designed to present itself as an image to a z/OS® host as one or more virtual tape libraries, each of which contain a range of virtual tape volumes and virtual drives. These virtual volumes are files that are contained in a disk cache file system. A virtual volume consists of customer data, header data, and metadata, which represent an entire image of a real tape volume. The customer data, header data, and metadata in this volume consist of hundreds to hundreds of thousands of objects.

A cost exists for z/OS to coordinate and manage back-end storage as it can require a large amount of the mainframe's processing power to package, move, and then manage the data continuously. This resource usage is measured in Millions of Instructions Per Second (MIPS).

It might be beneficial if this process can be reduced or eliminated. This IBM Redpaper publication discusses the ability to move some of this data from primary disk storage directly to an IBM TS7700 in the form of objects. The TS7700 not only retains its identity as a Virtual Tape Server, but it also acts as a private cloud storage device.

In 2017, z/OS DFSMS and DS8000 introduced transparent cloud tiering. When DFSMS and DS8000 Transparent Cloud Tiering manage a portion of the back-end storage workload, then z/OS can reduce the number of MIPS it uses for back-end storage processing and reserve those MIPS for other tasks. The transparent cloud tiering offering is focused on processing DFSMShsm and native DFSMSdss data to take advantage of these savings when offloading data.

The processing of back-end tape storage can be complex. A cloud solution allows for a simplified approach to managing data, as shown in Figure 1-1.
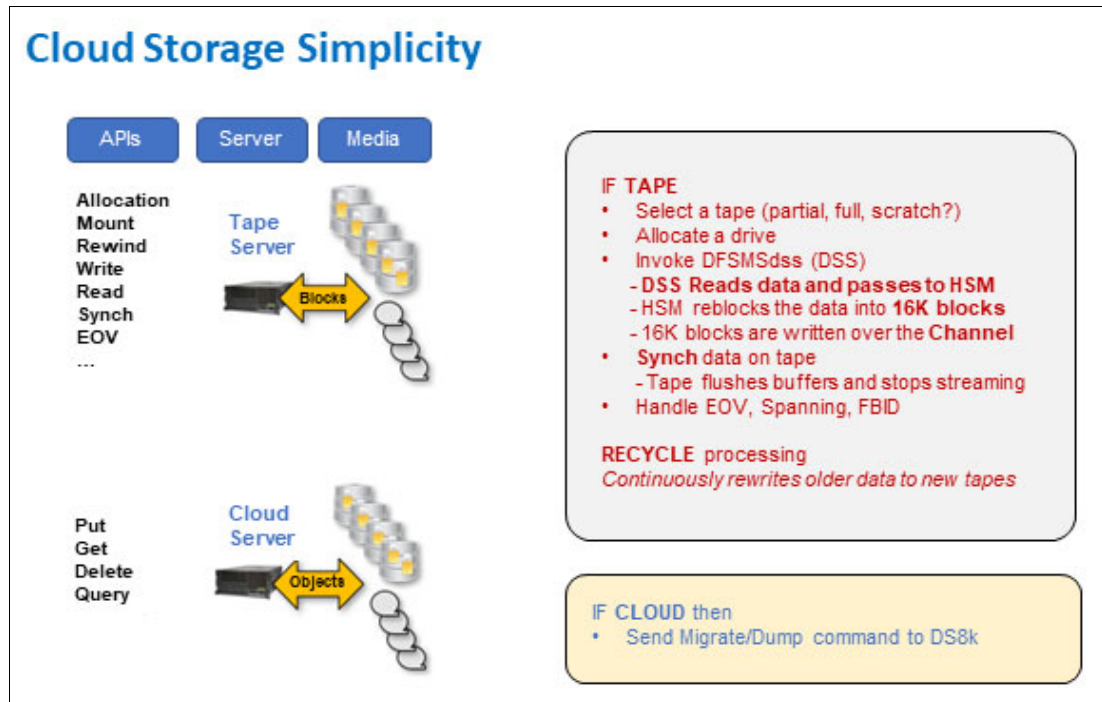


*Figure 1-1   Cloud storage simplicity*

Recalling the data by using tape services is also more complex and requires the data to move through multiple steps before it is available for use, as shown in Figure 1-2.
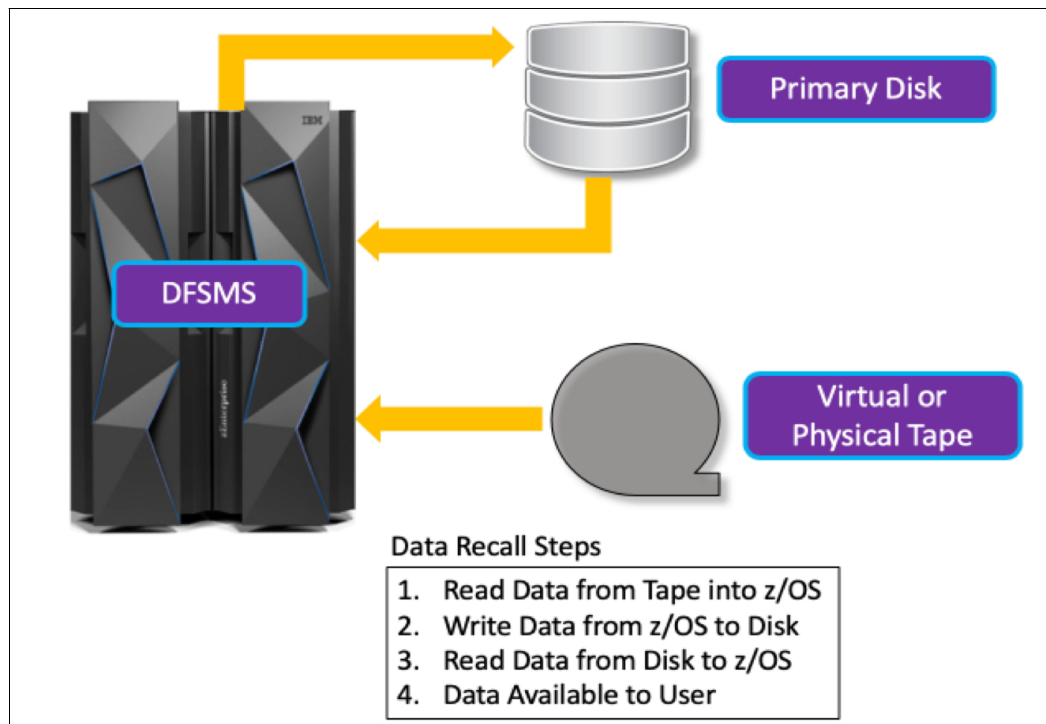


*Figure 1-2   Data recall steps from tape-based storage*

The process of migrating and recalling data when using transparent cloud tiering provides a simplified method for accessing stored data. Recalling data is reduced to a single step, as shown in Figure 1-3.
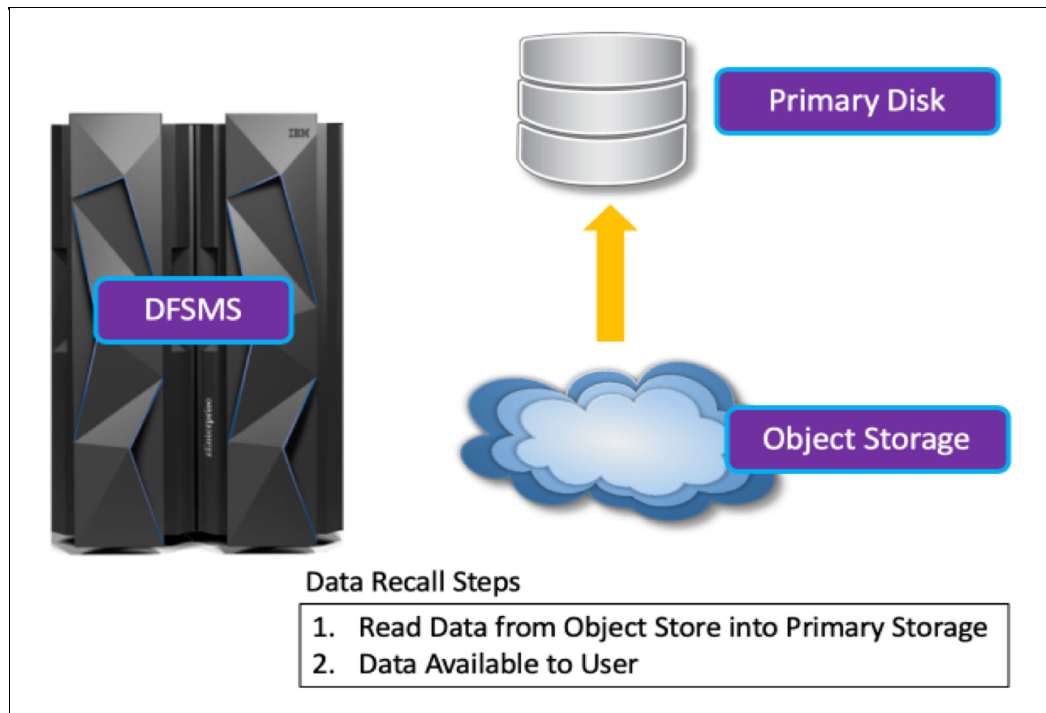


*Figure 1-3   Data recall steps for cloud object storage data*

Using an object store off the DS8000 allows for the elimination of HSM's main constraints and resource drains:

► Reduces a large amount of CPU resources needed to process data.
► Data passes through fewer layers of software.
► Allows data to be co-located.
► Allows parallel access to data.
► Eliminates the need for continuous recycles.
► Block size is not limited to 16 K.
► Removes impact on <control data set> REORGs, Audits, and backups.

z/OS DFSMS and DS8000 Transparent Cloud Tiering features enable the ability to move HSM MIGRATE data and DSS full volume dump, and other services, through an IBM Storage DS8000 to a cloud storage device. That cloud storage device can be an IBM TS7700 Virtualization Engine through one of two TS7700 offerings: DS8000 Object Store (FC 5282) or TS7700 Advanced Object Store (FC 5283).

## 1.2  DS8000 Object Store (FC 5282)

Beginning with microcode level 8.50.x.x, the TS7700 can store customer data, header data, and metadata in the form of objects within the same cache as virtual tape volumes through the Feature Code 5282 DS8000 Object Store offering.

This DS8000 Transparent Cloud Tiering solution not only avoids the HSM limitations that were mentioned previously, but also enables the ability to use existing TS7700 hardware to store the same hierarchical storage management (HSM) and DSS data.

This data is stored in the form of objects instead of tape logical volumes. Tape logical volume data and DS8000 Transparent Cloud Tiering object data can co-exist in the same cache storage space. This document describes the following DS8000 Object Store functions:

► Co-existence of logical volume data and object data within the same TS7700 cache system

► Logical cache partitions for object data

► Replication of object data by way of a forking mechanism to up to two TS7700 clusters in the same grid for FC 5282 only

► Encrypting object data between the DS8000 and TS7700

► Compressing object data at the DS8000 before sending it to the TS7700

► Error recovery mechanism for forked setups for FC 5282 only

► Display cache capacity and cache use on the TS7700 Management Interface (MI) for object data

► Display historical statistics for object partitions on the MI

► Library Request Commands:
    – To show object cache partition usage
    – To show throughput data for object workload
    – To set object cache partition alerts

► DS8000 Object Store enablement and setup information Restrictions and limitations

**Note:** FC 5282 is no longer available for distribution and is replaced by Feature Code 5283 TS7700 Advanced Object Store, which is described in the next section. At the time of writing, FC 5282 is still supported, but it is recommended to move to FC 5283 to benefit from the new functions and any future enhancements. For more information on upgrading FC 5282 to FC 5283, contact your IBM Representative.

## 1.3  TS7700 Advanced Object Store (FC 5283)

Beginning with microcode level 8.52.200.x, the TS7700 introduced Feature Code 5283 Advanced Object Store. This new feature further enhances the original offering by including complete grid awareness of objects, object policy management, and other functions that are covered in this document.

In microcode level 8.53.x.x, a copy refresh migration service for objects was also added as an enhancement to FC 5283. FC 5282 to FC 5283 migration capability is also available in this release level. These migration services are discussed in more detail in Chapter 10, "Migration services" on page 77.

Feature Code 5283 includes the same functions as FC 5282 described in 1.2, "DS8000 Object Store (FC 5282)" on page 4 except where noted. This document describes the following additional functions are supported by the FC 5283 Advanced Object Store feature:

► Grid awareness of objects
► Synchronous or asynchronous replication of object data to any TS7700 cluster in the same grid
► Object policy management
► DFSMS Cloud Network Connection Construct name association within TS7700
► Deferred copy throttling for objects
► Bulk handling of objects
► TS7700 object consistency model
► DS8000 Multi-Cloud Support (up to 8 object stores)
► Additional TS7700 Management Interface (MI) pages for object data
► MI Dual Control support for sensitive object store settings
► Additional Library Request Commands:

   – Modified object cache partition alerting
   – Summary of object incoming copy queue

► BVIR commands to dump the following object information:

   – OBJECT STATUS
   – OBJECT COPY AUDIT
   – OBJECT COPY QUEUE
   – HISTORICAL SUMMARY

► DS8000 to TS7700 connectivity and workload balancing
► DS8000 to TS7700 link monitoring and failover
► TS7700 handling of objects and containers
► Object Copy Refresh migration services
► FC 5282 to FC 5283 migration services
► FC 8083 prerequisite
► Memory requirements for Advanced Object Store
► Network considerations
► Removing FC 5282

### 1.3.1  Grid Awareness

Since its origination, the IBM Storage TS7700 is known for its grid technology. A TS7700 Grid consists of two or more clusters with each cluster being an active participant in the grid. There is no concept of primary or secondary clusters as each cluster is aware of all tape logical volumes in the grid.

Beginning in microcode level 8.52.200.x with FC 5283 TS7700 Advanced Object Store, each TS7700 cluster is aware of all object data and all tape logical volume data in the grid. Object data within the grid can be accessed through any FC 5283 feature-enabled cluster regardless of whether that cluster has a copy of the object.

If the z/OS host issues a `GET` (read) command to a TS7700 cluster that does not have a copy of an object, the TS7700 reads the object data from another cluster in the grid that does have a copy. Figure 1-4 is an example of how grid awareness works in the TS7700:
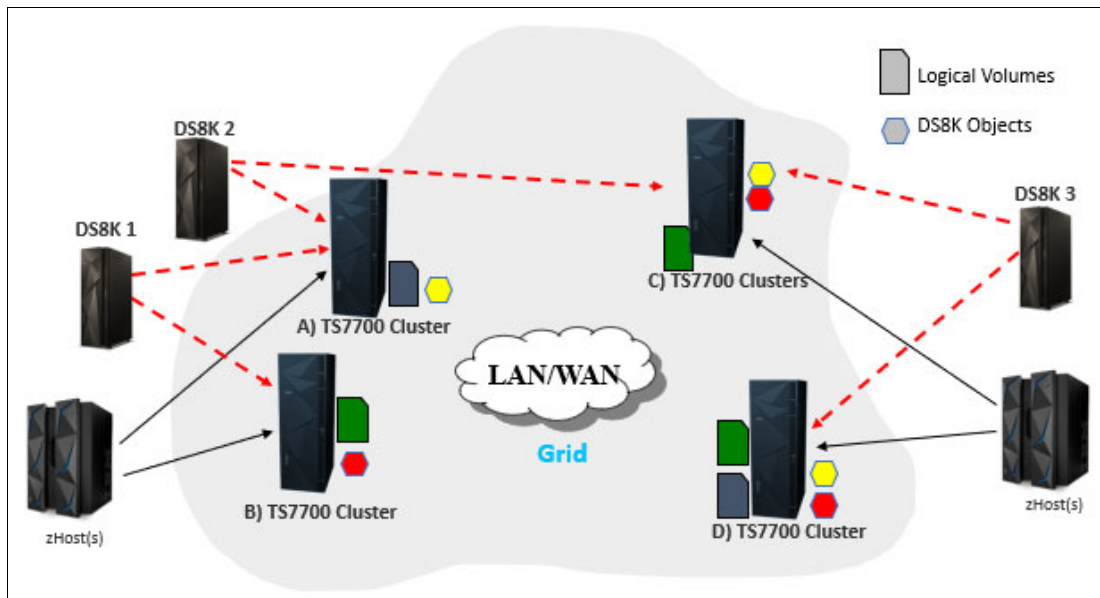


*Figure 1-4   Grid awareness example*

Figure 1-4 shows a 4-way TS7700 grid with clusters *A*, *B*, *C, and D.* DS8K1 is connected to clusters *A* and *B*, DS8k2 is connected to clusters *A* and *C* and DS8k3 is connected to clusters *C* and *D*.

DS8k1 does a PUT operation (on behalf of a z/OS host application) for the red object to cluster *B*. The red object is replicated to clusters *C* and *D* through a replication policy setting within the TS7700. If cluster *B* is unavailable for any reason, DS8k1 can still access the red object through cluster *A* even though cluster *A* does not have a copy of the data. The data is remotely read from either cluster *C* or cluster *D*. The TS7700 automatically locates the most optimal cluster in the grid to read the data from.

DS8k2 does a PUT operation for the yellow object to cluster *A*. The yellow object is replicated to clusters *C* and *D* through a replication policy setting. When cluster *B* becomes available again, it is notified of all changes that happened in the TS7700 grid automatically. If cluster *B* finds that it requires a copy of any data that was written while it was unavailable, it queues the copies and retrieves them. No user intervention is required as this is done as part of the grid self-healing functionality.

### 1.3.2  Object stores and object policy management

The TS7700 Advanced Object Store feature also introduces object policy management. A user can create, modify, or delete object policies from the TS7700 Management Interface Object Policy page. This page can be found under the **Object Store** navigational icon after FC 5283 is enabled.

When you create an object policy, you can choose a **synchronous**, **deferred** or **no copy** option for each FC 5283 enabled Advanced Object Store cluster within the grid. After policies are created, you can access the Object Store page that is found under the same navigational icon. From there, you can create an object store record and assign a policy to it.

The entry in the **Cloud Name** field in the Object Store page must match the DFSMS **Cloud Network Connection Construct** name (Cloud Name) created in the ISMF Cloud page (option S).

> **Note:** The TS7700 stores and displays a Cloud Name in lowercase, and z/OS might display this name in all uppercase.

This Cloud Name originates from the z/OS host and is sent to the DS8000 in every command to identify the target of a command. This Cloud Name is passed to the TS7700 with each object and container transaction including `GET`, `PUT`, `DELETE`, `LIST`, and `HEAD`. The TS7700 matches the Cloud Name that is associated with the object to the assigned policy in the Object Store page to determine which replication policy is applied. This flow can be seen in Figure 1-5.
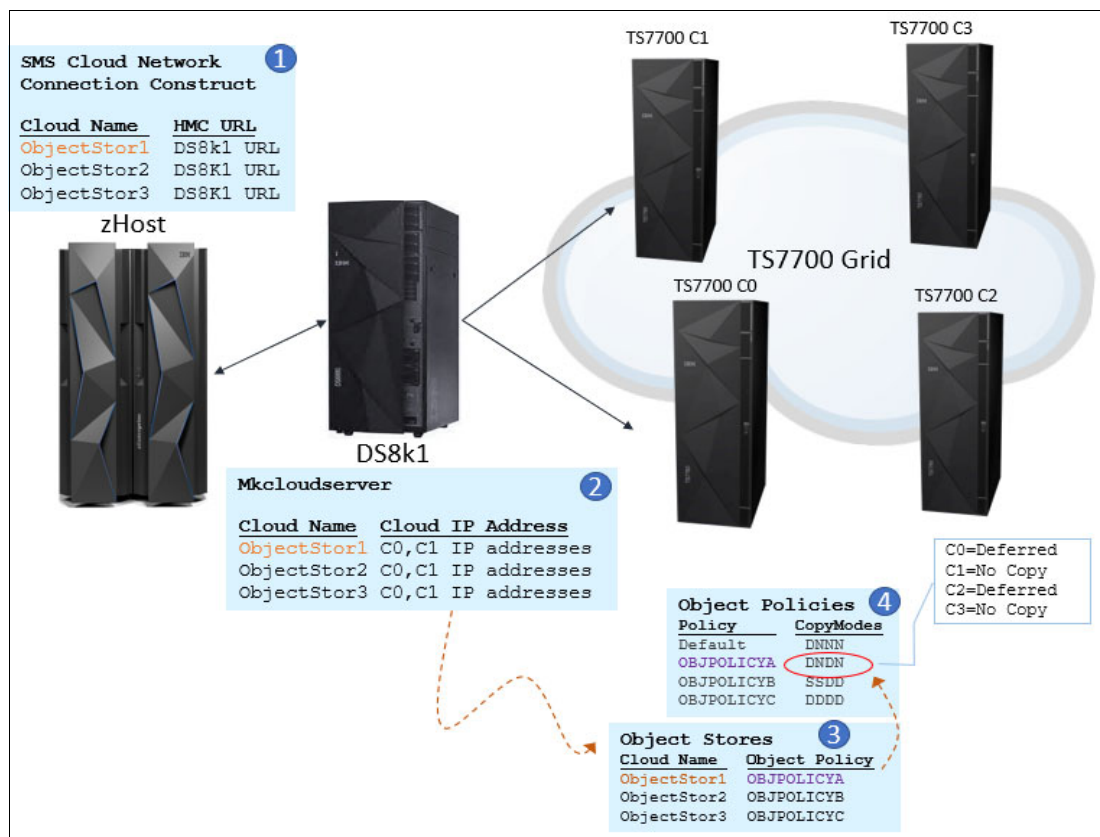


*Figure 1-5   TS7700 object policy management*

In Figure 1-5, a user defines the Cloud Network Connection Construct in the z/OS SMS page by providing the Cloud Name, the URL for the DS8000 HMC, and other required information. This construct is used for transparent cloud tiering.

After you enter the information, you configure the DS8000 connection to the object store by using the DSCLI `mkcloudserver` command. The `mkcloudserver` command requires the Cloud Name and the IP addresses of the object store that is being targeted. If a TS7700 is the object store, then the IP addresses are the primary and alternate grid link IP addresses. A DS8000 can target one or two TS7700s in the same grid for each Cloud Name.

Configure the TS7700 completely before you run `mkcloudserver.` If the Cloud Name does not exist on the TS7700 Object Store MI page when `mkcloudserver` is run, the command fails.

Figure 1-5 on page 8 shows the stages of an operation during a command such as **PUT** *object*:

1. z/OS host generates a **PUT** command for Cloud Name *ObjectStor1*. *ObjectStor1* targets the HMC for DS8k1.

2. DS8k1 receives the **PUT** command for Cloud Name *ObjectStor1* and matches it with a list of configured object stores. DS8k1 matches the Cloud Name and sends the command to one of the IP addresses provided during configuration. In this case, the IP addresses are for both TS7700 Cluster 0 (C0) and Cluster 1 (C1). DS8k1 balances the load between the two clusters, so the **PUT** goes to one of the two systems.

3. The **PUT** command arrives at one of the two TS7700 clusters. The TS7700 matches the Cloud Name with those configured by the user in the Object Store MI page. *ObjectStore1* is assigned object policy OBJPOLICYA.

4. The object policy dictates how the TS7700 handles the replication of the object throughout the grid. In this case, OBJPOLICYA has a copy mode setting of DNDN. This string value represents each of the four object-enabled clusters in the grid clusters 0–3. The first character from the left is the copy mode value for cluster 0. The second character is the copy mode value for cluster 1, and so on. 'D' means deferred copy, 'N' means No Copy, and 'S' means synchronous copy. This example of DNDN can be interpreted by using the following explanation:

   – The first 'D' is for cluster 0 (C0) and means that this cluster gets a deferred copy.
   – The second character, 'N', means cluster 1 (C1) gets no copy.
   – The third character, 'D', means cluster 2 (C2) gets a deferred copy.
   – The fourth character, 'N', means cluster 3 (C3) gets no copy.

If the **PUT** command arrives in C1, then C1 would do a remote write to either C0 or C2 because only these two clusters get a copy of this data as indicated by the 'D' in the C0 and C2 positions in the DNDN string. A deferred copy will be queued on the other cluster to later receive a copy.

If the **PUT** command came in through C0, then C0 writes the data locally and C2 queues a deferred copy.

If a **PUT** command is directed to a Cloud Name that is assigned an object policy with synchronous copies, such as OBJPOLICYB in Figure 1-5 on page 8, then the TS7700 fork-writes to both C0 and C1. Deferred copies are also performed on C2 and C3 as indicated by the 'SSDD'.

One side of a synchronous mode copy can continue even if one of the two clusters that are part of the forking is not available. In this case, the TS7700 enters a sync-deferred state. This means that the TS7700 returns a *success with warning* to the DS8000 to indicate that it can write to only one of the two clusters. The TS7700 is marked as degraded and sends a z/OS host notification that the TS7700 is in a sync-deferred mode.

After the TS7700 cluster becomes available again, it will prioritize sync-deferred copies over any other copy transaction until all sync-deferred copies are complete. Then, the cluster exits the sync-deferred state and notifies the z/OS host that it has returned to normal.

For more information about the Object Policy and Object Store Management Interface (MI) pages, see Chapter 7, "TS7700 management interface" on page 51.

### 1.3.3  Object, container, and token handling within the TS7700

This section discusses how z/OS host applications, DS8000 Transparent Cloud Tiering, and the TS7700 Advanced Object Store functions perceive and manage objects and containers.

#### Containers
Containers cannot be created or deleted by a TS7700 user directly. The z/OS host application can create and delete containers on any object store. Each application uses a unique naming convention for any containers that it creates. DFSMShsm automatically generates its own container name by using an internal format and will periodically create containers. An example of a DFSMShsm container is `Syszarc.arciml.mig.20210101`.

A DFSMSdss container name is defined by a user and dss adds a SYSZADR prefix to the name. Containers are not automatically generated for dss data but are created when a user runs the CLOUDUTILS CONTAINER command. An example of a DFSMSdss container is `Syszadr.Backup1.20210101`.

Additional information about CLOUDUTILS can be found in Example 3-3 on page 33.

Other host applications have different container naming and creation conventions. Follow their guidelines when setting up to use DS8000 Transparent Cloud Tiering. DS8000 Transparent Cloud Tiering supports the following characters in the container name (excluding commas):

`A to Z, a to z, 0 to 9,#,-,.,_`

When a z/OS host application needs to create a container, it sends a create container command to the DS8000 who's URL is associated with the DFSMS Cloud Network Connection Construct name (also referred to as Cloud Name). DS8000 calls a create container command for the TS7700 cloud that is associated with this Cloud Name. The TS7700 creates the container entry on all object store-enabled clusters in the grid for that Cloud Name.

A container in the TS7700 is a directory in the file system and is always associated with an existing Cloud Name. This container is equivalent to a bucket, vault, or other similar terms and is used to store object data.

The same container name can exist under different Cloud Names and would be considered a different container. A container name must be unique within a Cloud Name.

Containers cannot be modified and do not hold individual properties or policy values within the TS7700. Policies are set at the Cloud Name level.

Container creation follows a real-time consistency model. When a create container command comes down from a z/OS host, the TS7700 makes all object-store enabled clusters aware of the container in real-time.

Container deletion can follow either a real-time or an eventual consistency model. When the z/OS host sends a delete container command, the TS7700 marks the container as delete-pending and immediately update all object store-enabled clusters. This is handled with real-time consistency.

The actual deletion might be minutes, hours or more depending on object status and other conditions within the grid. However, a z/OS host will always be notified that the container no longer exists on any future query.

If the z/OS host re-creates the same container before a delete-pending container process has completed, the container will be reset to an active state but with a new container ID. The new container ID does not affect objects that are marked for deletion under the previous container ID. In other words, those objects are still subject to deletion under the old container ID even if the container is reactivated. The new container ID and the object deletions are internal to the TS7700s, so no notifications are sent to the zhost or DS8000.

Containers with objects cannot be deleted (or be marked for deletion) unless all objects associated with it are already marked for deletion. The exception to this case is if the z/OS host application supports a container force delete command. DFSMSdss CLOUDUTILS is one example of a host application that supports force deletions of containers. If force delete is used, all objects in the Cloud Name/container, are marked for deletion. For more information, see CLOUDTUILS command for DFSMSdss.

## Objects

An object is an individual piece of data. Many objects can be associated with each other to form a data set, a full volume of data, or a full dump of data depending on the host application. Objects are always associated with a Cloud Name and a container.

Objects cannot be created or deleted by a TS7700 user directly. The z/OS host application creates and deletes objects as part of normal operation. Objects are immutable. They are deleted and created anew if the data has changed but an existing object is never modified.

Objects might be replaced if the same Cloud Name+container+object exists during a `PUT` (write) operation. This is considered an intentional host overwrite. In this case, the old object is marked for deletion and a new object is written.

**Note:** At the time of writing, DFSMS applications do not support host overwrite operations through transparent cloud tiering. A verification check for the existence of the object is done as a prelude to an object `PUT`. If the object exists, it is deleted before a new object is written. Other z/OS host applications might support overwrites. Check with your z/OS host application vendors to understand how overwrites are handled.

Each object is associated with an object ID, so the new object has a different object ID compared to the old object even if they have the same object name. Therefore, the same object name might exist under the same Cloud Name/container but will always have a different object ID. Each of these objects is considered an entirely different object.

During an object `PUT`, the accessing cluster is the cluster that receives the `PUT` command. The cache-target cluster is the cluster that is targeted to receive a copy of the data if the cache-target cluster is different from the accessing cluster.

During an object `PUT`, the TS7700 records the object name, ID, and other information in a database table entry on the accessing cluster, the cache-target cluster, and the fork cluster (if synchronous copy is used).

All other FC 5283 enabled clusters are notified of the object through an object token process. These database table entries are updated asynchronously. All future operations for the object trigger a grid-wide audit and instant token reconciliation. This provides the TS7700 with a strong consistency model for object data handling. The token processing operation is discussed in the next section.

When a z/OS host application initiates a DELETE, the TS7700 audits the grid for the object and ensure any tokens involving the object are reconciled. It then marks the object delete-pending on the accessing cluster.

All other FC 5283 enabled clusters are notified of the new status asynchronously. Any future `GET`, `PUT`, `LIST`, `HEAD`, and `DELETE` commands will also return the object as "not found". Therefore, deletion of an object is also a strong consistency model. The actual physical deletion of the objects will happen over time by using a background process.

When a z/OS host application initiates a `GET` (read) of the object, TS7700 audits the grid and ensures that all tokens are reconciled for the object, and then picks the best cache-target cluster. If the accessing cluster that receives the `GET` has a copy of the object, it will always be the cache-target cluster.

Because objects can range in size from just a few bytes to terabytes, DS8000 divides large data objects into 2 GB slices to improve performance. For example, a 20 GB track is stored as ten 2 GB objects. The object name includes an index value for each chunk starting at 0000000 and increments for each chunk and is tracked in the TS7700 as individual objects.

Example 1-1 provides an example of a 20 GB track that is stored as objects in the TS7700:

*Example 1-1   How a 20 GB object is broken up into extents, or chunks*

```
PREFIX1:EXTENTS0000000
PREFIX1:EXTENTS0000001
PREFIX1:EXTENTS0000002
PREFIX1:EXTENTS0000003
PREFIX1:EXTENTS0000004
PREFIX1:EXTENTS0000005
PREFIX1:EXTENTS0000006
PREFIX1:EXTENTS0000007
PREFIX1:EXTENTS0000008
PREFIX1:EXTENTS0000009
```

### Token handling

The TS7700 uses tokens to update object-enabled clusters across the grid during PUT and DELETE operations. Object tokens are meant to update the metadata for the object, not the actual data. After the initial write of the object on the TS7700(s), all other clusters with the Advanced Object Store feature enabled will be informed of object operations using object tokens regardless of whether the cluster will get a copy of the object. This is an asynchronous operation.

These tokens process object metadata in bulk by using a consistency group concept. In other words, one object token can send metadata for hundreds or thousands of objects. This reduces the amount of traffic that is sent across the grid.

The accessing cluster is the initial driver and owner of that token. Object token ownership is not expected to change unless something happens to that cluster.

Unlike the tokens the TS7700 uses for logical volume data processing, object tokens are temporary records. After the token updates all object store clusters with its group of objects, it is closed.

The object token's primary purpose is to ensure that all object-enabled clusters in the grid are made aware of object creations and deletions. The object token also plays a role in driving grid copies. During token processing of new objects on remote clusters, the object replication policy is evaluated. If the object policy indicates the cluster requires a copy, it inserts an object record into its object copy queue before closing out the token process. Object replication is discussed further in 6.2, "Object replication" on page 42.

Object tokens also play a role in updating object-enabled clusters across the grid when objects need to be deleted. Object deletions are also handled in bulk and one token can update the deletion state of many objects. The process is the same as object creation except creation tokens are pushed to the other clusters in the grid when they close out. The TS7700 allows only a set number of objects in a creation token and there is a minimal amount of time the token can be left open. This allows for faster updates for new object processing across the grid.

For deletions, the token processing is less urgent, so consistency tokens that handle object deletion are generated by using a background process on the accessing cluster (the cluster that received the delete command) and are processed at a lower priority.

As previously mentioned in this document, token processing could be escalated to be processed immediately when a new command arrives for an object. The TS7700 reconciles all open tokens for the object before validating and proceeding with `PUT`, `GET`, `and DELETE` operations to ensure the most up-to-date information is being considered.

### 1.3.4  DS8000 to TS7700 communication

Since the original release of *DS8000 Object Store* (FC 5282) in microcode level 8.50.x, DS8000 can be configured to connect to one or two TS7700 clusters. In this earlier release, DS8000 performs forked (synchronous) write to both TS7700s if it was connected to two clusters.

Beginning in 8.52.200.x microcode level with the release of theTS7700 Advanced Object Store (FC 5283) function, DS8000 no longer fork-writes to the two TS7700s. The DS8000 now load balances between the two clusters, and the task of creating copies on additional clusters in the grid is handled by the TS7700 through object replication settings. Object replication is discussed further in section 6.2, "Data Redundancy" on page 33.

With FC 5283, there are three layers of load balancing in the DS8000:

1. Load balance between DS8000 CECs. The DS8000 will send data transactions for odd LSS's to CEC1 and even LSS's to CEC0. This means that it is possible that only one CEC is being utilized if all TCT operations occur on only even LSS's or odd LSS's.

2. Each DS8000 CEC will load balance between the two TS7700 clusters (-primaryTS7700IPs for the first cluster and -secondaryTS7700IPs for the second cluster).

3. DS8000 will round-robin through each cluster's IP address sets so that the same link is not always used.

Retries on other clusters or other links can occur during a failure depending on when the failure occurred and for which operation. For example, if an initial `PUT` command fails before any data is written, the command is routed to another link or cluster. If a `PUT` command fails after writes have started to the TS7700, the DS8000 will fail back to the host, so the host can control cleanup operations.

## 1.4  DS8000 multi-cloud

In DS8900 microcode level R9.2, DS8000 introduced multi-cloud functionality. This allows the DS8000 to be configured and connect with up to 8 object stores. This can be a combination of traditional object stores and TS7700 object stores, or all 8 connections can be configured to the same object store type. For instance, TCT can be configured to the same two clusters in the same TS7700 grid by using different Cloud Network Connection Constructs.

Figure 1-5 on page 8 shows an example of how using different Cloud Network Connection Constructs can drive different policy behavior in the same TS7700 grid. This figure shows three cloud construct names, ObjectStor1, ObjectStor2, and ObjectStor3. Step 3 in the figure shows how the three cloud names are associated with different object policies (OBJPOLICYA, OBJPOLICYB, OBJPOLICYC). Each object policy has a different replication policy behavior, which provides the ability to customize workloads by choosing a different Cloud Name for each different object policy and its associated copy modes.

Using different Cloud Names in the same grid also has the benefit of having separate statistical pools. Statistical data is collected at the Cloud Name level in a TS7700 Grid.

Figure 1-6 shows all the statistics available for DS8000 Transparent Cloud Tiering as of microcode level 8.52.200.x. Those listed in the light green boxes are statistics at the Cloud Name bucket level.
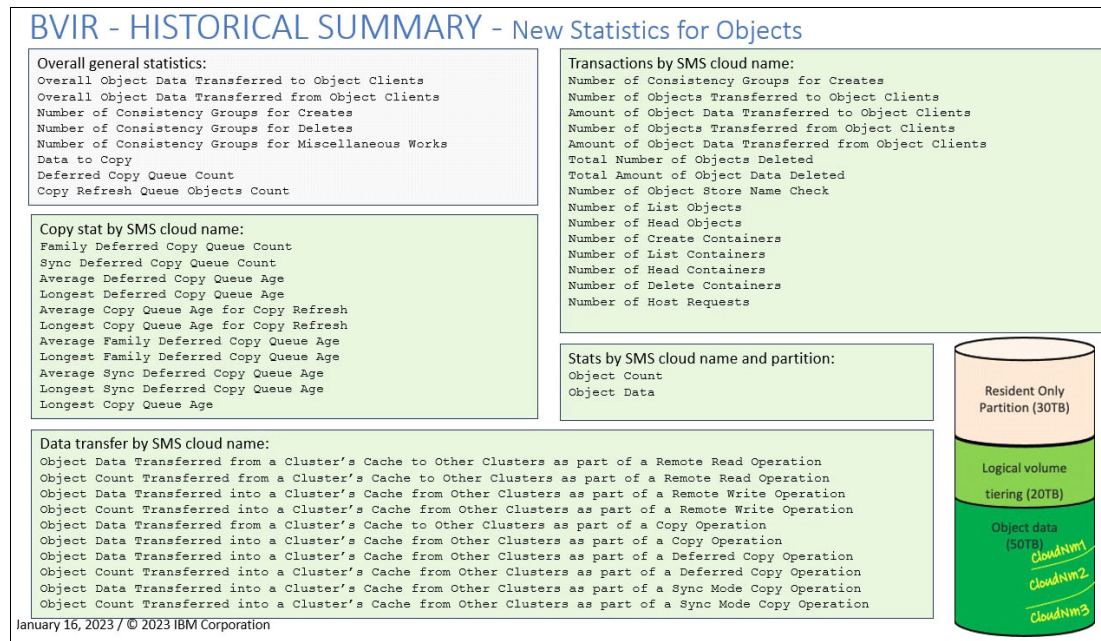


*Figure 1-6   Object Historical Statistics*

9.4, "VEHSTATS statistical reports" on page 73 provides more information about retrieving these statistics through BVIR and generating VEHSTATS reports. For more information about these statistics, see TS7700 Statistical Data Format and TS7700 Bulk Volume Information Retrieval Function User's Guide white papers.

The DS8000 can also be configured so that each Cloud Name points to a different TS7700 grid to further separate different workloads.

# Hardware and software requirements

This chapter provides information about hardware and software requirements.

This chapter includes the following sections:

## 2.1  TS7700 supported configurations

The DS8000 Object Store (FC 5282) and Advanced Object Store (FC 5283) functions are supported in a stand-alone configuration or in a grid configuration that can include up to 8 systems. DS8000 can target one or two TS7700s in the same grid per DS8000 system. For more information, see 6.2, "Object replication" on page 42.

The DS8000 Object Store and Advanced Object Store features are supported on TS7770 (VED) and future models. For the most optimal performance, use 10 Gb Ethernet links and 128 GB memory in a TS7700 that is a target for DS8000 Transparent Cloud Tiering. As of TS7700 microcode level 8.53.x, all new TS7770s come standard with 128 GB memory.

The TS7700 uses port 350 for transparent cloud tiering transactions from the DS8000. Port 350 is the same port that is used for grid-wide transactions, such as copy operations and remote read/write operations.

FC 8083 includes installation of two new SSD drives into the Power9® server of the TS7770 and is required when ordering FC 5283.

> **Note:** As of December 2021, FC 5282 is no longer available for purchase on the IBM ordering site. Only FC 5283 for DS8000 object store can be ordered.

## 2.2  TS7700 microcode requirements

The DS8000 Object Store (FC 5282) function was introduced with the TS7700 microcode level 8.50.x.x. Target clusters must be at this microcode level or higher to enable DS8000 Object Store. The TS7700 Advanced Object Store feature (FC 5283) was introduced with TS7700 microcode level 8.52.2x.x. Target clusters must be at this microcode level or higher to enable Advanced Object Store. Other clusters can be in the grid at an earlier microcode level but cannot be a DS8000 object target.

FC 5282 and FC 5283 cannot exist in the same grid unless you are migrating from FC 5282 to FC 5283. Support for FC 5282 to FC 5283 migration began in microcode level 8.53.x.x. FC 5283 must be installed on all clusters in the grid that have FC 5282 before this migration begins. FC 5283 must not be installed on any other cluster in the grid if FC 5282 exists.

If no objects exist on a TS7700 cluster for FC 5282, FC 5282 can be removed avoiding the need to go through the migration. For more information about migrating from FC 5282 to FC 5283, see 10.2, "FC 5282 to FC 5283 migration" on page 78. For more information about the removal of FC 5282, see 10.1, "Removing FC 5282" on page 78.

To do a copy refresh migration of objects, TS7700 microcode level 8.53.x is required. For more information about this migration, see Chapter 10, "Migration services" on page 77.

## 2.3  DS8000 supported configurations

DS8000 Transparent Cloud Tiering is supported on all current DS8000 configurations:

- ► Simplex
- ► Metro Mirror
- ► Global Mirror
- ► Metro Global Mirror
- ► HyperSwap®

DS8880 or DS8900 are supported for the TS7700 FC 5282 DS8000 Object Store feature. TS7700 FC 5283 Advanced Object Store is only supported from DS8900 models.

A standard 1 Gbps Ethernet interface is available on the DS8000 and can be upgraded to 10 Gbps Ethernet. A 10 Gbps Ethernet connection is recommended for this workload.

> **Note:** DS8900 10 Gbps Ethernet *cannot* direct connect to TS7700 GRID adapters because of a difference in fiber optic mode (SR versus LR connectivity). DS8880 10 Gbps adapters can direct connect.

Regardless of which feature being used, a network connection between the DS8000 and TS7700 is required within the user's environment.

DS8000 Transparent Cloud Tiering uses REST API for operational transactions between the z/OS host and the DS8000. The default port used for this transaction on the DS8000 is 8452 and is the same port that is used for the DS8000 GUI and DSCLI.

## 2.4  DS8000 microcode requirements

The DS8000 Transparent Cloud Tiering function is part of the zSynergy license bundle and requires the following prerequisites to support the DS8000 Object Store feature (FC 5282) in the TS7700:

- ► A DS8880 with R8.5SP4 or later
- ► DS8900 with R9.0 or later

TS7700 Advanced Object Store (FC 5283) is supported on a DS8900 R9.3 or later. FC 5283 is not supported on DS8880 systems.

A DS8900 multi-cloud environment also requires a DS8900 running R9.3 or later and is not supported on DS8880 systems. For more information about multi-cloud support, see 1.4, "DS8000 multi-cloud" on page 13.

For FC 5282, if DFSMSdss full volume dump is routed through the DS8000 by using transparent cloud tiering, then DS8880 microcode level R8.5 SP6 or later is required. When using a DS8900, microcode level R9.1 or later is required. If FC 5283 is used, then DFSMSdss full volume dump is already supported in later code levels.

If you require DS8000 object compression or encryption to a TS7700, the DS8000 must be a DS8900 with microcode level R9.2 or later. DS8000 object compression and encryption are not available for DS8880 systems when targeting a TS7700.

The DS8000 also includes configuration steps. This document provides high-level configuration steps in 3.2.3, "DS8000 configuration" on page 27. For more in-depth

information about DS8000 Transparent Cloud Tiering and configuration requirements, see the following resources:

► *IBM DS8000 Transparent Cloud Tiering: DS8000 Release 9.3,* SG24-8381

► Configuring transparent cloud tiering

## 2.5 IBM Z host considerations

In the z/OS transparent cloud tiering solution, z/OS applications that support transparent cloud tiering write object data to the cloud object store by way of an IBM System Storage DS8000. Each host application can have its own configuration requirements at the application level. This document provides only general information about object flow from z/OS to the TS7700 and how to configure DFSMS for transparent cloud tiering.

The following is a list of IBM Z applications that support transparent cloud tiering:

► DFSMShsm automatic data set migration using Management Class policies

► IBM Db2 image copies using IBM FlashCopy with DFSMSdss then MIGRATE STORAGEGROUP

► DFSMSdss data set disk backup

► DFSMSdss native full volume dump by using FlashCopy

► DFSMShsm full volume dump

► Db2® system level backup using DFSMShsm triggered FlashCopy®

For more information about z/OS transparent cloud tiering, see the following resources:

► DFSMS transparent cloud tiering
► IBM DS8000 Transparent Cloud Tiering: DS8000 Release 9.3, SG24-8381
► Using DFSMS transparent cloud tiering

The z/OS host must be at V2R4 base level or later. To use native DFSMSdss full volume dump, the PTF for APAR 0A57526 is required. PTFs are available from the *IBM.Function.DFSMSCloudStorage* fix category.

If DS8000 object compression will be used, additional APARs might be required to configure a new compression setting called `TCTCOMPRESSION` in *DEVSUPxx parmlib*.

**3**

# Configuration and setup

This chapter provides information about how to configure DS8000 Transparent Cloud Tiering to a TS7700.

This chapter includes the following sections:

To write objects from a DS8000 to a TS7700, you configure multiple products including z/OS, DS8000, and the TS7700. In Appendix A, "Transparent cloud tiering setup worksheet" on page 85, a worksheet is provided to help ensure all the steps are performed during preparation and configuration. These steps are discussed in more detail in this chapter.

# 3.1 Prerequisites

Complete the following prerequisites before continuing to the next section:

► Gather Information that is needed during setup:

– The DS8000 serial number that is connected to the TS7700.

– DS8000 Ethernet IP addresses for each central electronics complex (CEC) (primary and if applicable, secondary).

– The TS7700 Grid Network IP addresses that are targeted.

► The TS7700 must be cabled to the network that is shared with the DS8000 by using theTS7700 ports that correspond to grid or inter-cluster communication.

– The network configuration between DS8000 CECs and the TS7700 Grid Network is designed with the expectation that every DS8000 network interface has reliable connectivity to every TS7700 network path. If your data center network configuration does not include all link paths between the DS8000 and TS7700 and if only like-subnet IP addresses have available paths, then VTD_EXEC.369 is required. For instance, if the interfaces are connected to physically separated IP subnets that cannot be routed to another subnet, the VTD_EXEC is needed to prevent packet failure. This VTD_EXEC restricts those routing paths that do not lie within the same local subnet. Do not use this VTD_EXEC in any other situation. Otherwise, loss of communication can occur.

– Cisco ACI is not supported in this environment. ACI must be disabled. Otherwise, routing fails.

► Port 350 is used and must be available for DS8000 Transparent Cloud Tiering traffic.

► Ensure that the TS7700 has at least one available logical cache partition slot to be reserved for object storage. If the TS7700 system is tape or cloud-attached, it might already be using multiple cache partitions. At least one partition must be available. Otherwise, you cannot enable the feature code.

► Ensure that you have a FC 5282 or FC 5283 license key.

> **Note:** FC 5282 has been replaced with FC 5283. If FC 5282 has been installed on the system, it is recommended to move to FC 5283. No additional enhancements will be made to FC 5282. Information on how to migrate to FC 5283 can be found in 10.3, "Object migration services" on page 80.

► If you are using FC 5283 Advanced Object Store, then FC 8083 is required. FC 8083 adds two new SSD drives to the Power9 server for object metadata storage.

► If encryption of user data is required between the TS7700 and DS8000 for transparent cloud tiering, or between two TS7700 clusters in the grid for object replication data security, then install FC 5281 Secure Data Transfer on each cluster in the grid.

► 128 GB of memory is recommended for this feature. Beginning with microcode level R5.3, 128 GB memory is required for all TS7700 systems.

► Ensure that the TS7700, DS8000, and z/OS are at the appropriate code levels.

► If you are adding FC 5283, determine whether FC 5282 exists in the TS7700 Grid Network. If FC 5282 exists, then additional steps are needed as described in 10.1, "Removing FC 5282" on page 78 and 10.2, "FC 5282 to FC 5283 migration" on page 78.

**Important:** Do not activate FC 5283 on a TS7700 cluster if any other cluster in the grid contains FC 5282. FC 5283 can be activated on TS7700 clusters with FC 5282 if the intent is to migrate from FC 5282 to FC 5283. If no objects have been written to the TS7700 Grid by using FC 5282, a migration might not be needed.

The following list provides a high-level overview of the steps that are required for setup. See 3.2, "Detailed configuration steps" on page 21 for details about the steps for each IBM product:

1. Ensure that the TS7700 is at the proper microcode level.

2. If additional system memory is needed, then schedule an SSR to install it.

3. If using FC 5283, an IBM SSR must install FC 8083.

4. Install TS7700 Feature Code (FC) 5282 or 5283.

5. Run the DS8000 Object Store or the Advanced Object Store MES process on TS7700 SSR activity.

6. Enter DS8000 IP addresses on TS7700 (SSR activity).

7. If DS8000 object encryption will be enabled, Feature Code 5281 must be installed.

8. Modify the Cache Partition size if needed.

9. If using FC 5283, set up Object Policies.

10. If using FC 5283, set up Object Stores and assign policies.

11. Ensure that the DS8000 is at the proper microcode level.

12. Set up DS8000 TCT Ethernet ports using the `setnetworport` command.

13. Ensure that the network path is established between the DS8000 and TS7700 grid network.

14. Run `mkcloudserver` command to configure to the DS8000 to the TS7700.

15. Ensure that the z/OS is at the proper software level.

16. Configure the z/OS Cloud Network Connection Construct.

17. Configure the z/OS application as specified by the application requirements.

18. Verify and configure other DFSMS settings.

## 3.2 Detailed configuration steps

This section gives more detailed information about the configuration steps and provides an introduction to understanding the data flow for this feature. It is not meant to include all requirements, configuration steps, and limitations. For more information about configuration and other transparent cloud tiering concepts, see "Other publications" on page 95.

**Note:** For more information about this feature, you can join the DS8000 Transparent Cloud Tiering IBM TechXchange Community group where you can start a discussion, search the library, or view upcoming events on this subject. This is a public forum where IBM Storage users meet, share, discuss, and learn.

## 3.2.1  Network considerations

Network connections to use DS8000 TCT focuses on three areas:

1. The network between the z/OS host and DS8000 for FICON® CCW transactions

2. The network between the z/OS host and DS8000 via Ethernet as a REST API proxy

3. The network between the DS8000 and the TS7700 for all object transactions to the object store

### FICON CCW Path

The FICON CCW path is the standard path used to send and receive CCW messaging and read and write tracks between the mainframe and DS8K for normal operations. This same FICON communication path is used to receive commands from the zHost for TCT data movement of customer EXTENTS data. This paper assumes this FICON connection is already established. No network changes are needed to support TCT. Therefore, no additional details will be provided about this connection.

### REST API Proxy

DFSMS requires a REST API proxy to send control commands as well as a path to send meta data objects it creates to manage cloud objects. Control commands include things like doing `LIST`, `HEAD`, and `DELETEs`. The path supports `PUT` and `GET` operation for the meta data objects.

An Ethernet network path must be established between the z/OS mainframe and any DS8000 target, including any secondary DS8000s that may become a primary target for a zHyperlink switchover. In the case Global Mirror and/or Metro Global Mirror exists, network paths between any DR z/OS system and those GM/MGM systems must also be established.

On the DS8000, this is the HMC Ethernet ports and should already be established as part of DS8000 setup. On the z/OS, this is the OSA connection. Port 8452 is needed for the cloud proxy connection. It is assumed the z/OS OSA connection is already configured as well. If it is not, contact your z/OS Mainframe representative for configuration steps.

To configure z/OS to a DS8000, a DS8000 user must be created with monitor access on the DSGUI. The user ID and password will be used on the z/OS ISMF Cloud Network Connection Construct panel when creating the cloud connection. This panel can be found under the ISMF Cloud window (option "S").

For more detailed information about configuring z/OS DFSMS to the DS8000 HMC, including information about certificate management see *IBM DS8000 Transparent Cloud Tiering Guide*, SG24-8381.

### DS8000 and TS7700 Network

This section focuses on the connectivity between the DS8000 and the TS7700.

The following list describes some DS8000 to TS7700 network considerations:

► DS8000 Ethernet ports are connected to the customer network to create a path between DS8000 subnet and the TS7700 Grid network. From the DS8000, configure the Ethernet ports to the routing table by using the `setnetworkport` and `lsnetworkport` commands.

► If the TS7700 is a stand-alone system, it will also need to be cabled and configured to the network because the grid links are not being used.

► When the DS8000 is connected to two clusters in a grid with DS8000 Object Store FC 5282, data is offloaded synchronously to both clusters by using forking. See Figure 3-1 on page 23.

► When the DS8000 is connected to two clusters in a grid with Advanced Object Store FC 5283, traffic is load balanced between the two clusters for better efficiency. See Figure 3-2.

► The DS8000 supports up to eight separate object stores when you use microcode level R9.2x or later. This can be a mixture of private or public cloud. If the TS7700 is the object store target, the DS8000 can configure different object store cloud names to the same TS7700 Grid or to different TS7700 Grids as shown in Figure 3-3 on page 24.



*Figure 3-1   DS8000 object store configured*



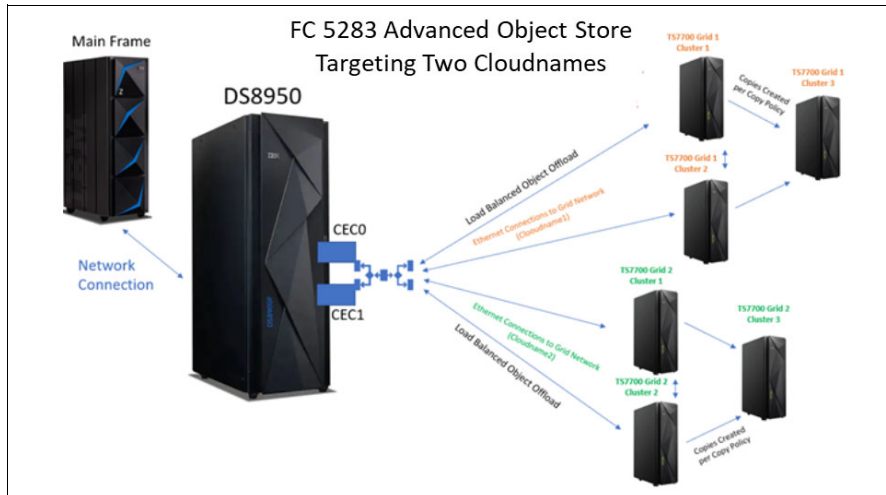*Figure 3-2   DS8000 with advanced object store configured*

*Figure 3-3   DS8000 with multi-cloud advanced object store configured*

### Considerations and requirements

*Table 3-1   Additional details and information related to solution components*

| Component | Requirements |
|---|---|
| Cabling | If the target TS7700 is already in a running grid, then it is already cabled to a network. Ensure that the DS8000 has a network path to the TS7700 Grid network. If the TS7700 is a stand-alone system, it must be cabled to a network the DS8000 can access. A short procedure is run by an IBM SSR to establish the network ports on the TS7700. |
| Networking considerations | Ensure that the DS8000 has a network path to the TS7700 Grid network. The network configuration between DS8000 CECs and the TS7700 Grid Network is designed with the expectation that every DS8000 network interface has reliable connectivity to every TS7700 network path. Each DS8000 port must be able to receive responses back from any TS7700 port configured, even those on different subnets. If this is not possible, VTD_EXEC.369 may be required. This VTD_EXEC restricts those routing paths that do not lie within the same local subnet. Do not use this VTD_EXEC in any other situation otherwise loss of communication can occur.<br><br>CISCO ACI is not supported in this environment and must be disabled. |
| Port information | Port 350 is required for data transfer between a DS8000 and TS7700.The TS7700 uses port 350 for grid communication, therefore the DS8000 must ensure this port is open. |

| Component | Requirements |
|---|---|
| Ethernet adapters | TS7700 comes with 1Gb Ethernet adapters by default. Optional 10 Gb LR adapters can be added to the TS7700 cluster and are recommended for this workload.<br><br>DS8880 and DS8900F by default, come with 1 Gb Ethernet adapters with RJ45 connectors. Optional 10 Gb Ethernet adapters can be added to any DS8880/DS8900F<br><br>DS8880: The 10 Gb Ethernet adapter uses SFP+ LR (long reach) connectors and can be directly connected to a grid network on the TS7700.<br><br>DS8900F: The 10 Gb Ethernet uses SFP+ SR (short reach) connectors. An Ethernet switch is required between DS8900F and TS7700.<br><br>There are two ports available from each Ethernet card per CEC (4 ports total) whether 1 Gb or 10 Gb. These ports are used for redundancy only (not bonded). Additional feature codes may be required to enable the secondary ports on these adapters.<br><br>It is recommended that if redundancy is required, each port of the Ethernet card routes through a different switch such that network hardware boundaries can be enforced if there is switch failure, port failure, SFP failure, and so on. |

## 3.2.2  TS7700 configuration

This section contains information for the following topics:

▶ Enabling the object store feature codes:
  – If installing FC 5282, go to "Installing FC 5282"
  – If installing FC 5283, go to "Installing FC 5283" on page 26
▶ Configuring DS8000 IP addresses in the TS7700
▶ Resizing Logical Cache Partitions
▶ Data In-flight Encryption
▶ Object Policy Management

### Installing FC 5282

Feature code 5282 is activated on the TS7700 Management Interface Feature License page after it is at microcode level of 8.50.x.x or later. The DS8000 offload feature codes are only required on clusters that are a DS8000 Transparent Cloud Tiering target. No mixture of FC 5282 and FC 5283 is allowed in a grid.

> **Important:** If a TS7700 grid contains clusters with FC 5282 and the feature will be upgraded to FC 5283, an FC 5282 to FC 5283 migration must be performed. This procedure is discussed in section Chapter 10, "Migration services" on page 77. The steps in this section do not cover the FC 5282 to FC 5283 migration scenario.

After the FC is added, the IBM SSR must run an enablement procedure from the TS7700 service window. The enablement process does the following steps:

a. Take the TS7700 into service-prep and offline.
b. Enable *DS8000 Object Store* on the TS7700 service window.
c. Bring the TS7700 back online.

### Installing FC 5283

If FC 5283 Advanced Object Store is being used, an SSR must first install FC 8083, which adds two 3.84 GB SSDs to the TS7700 Power9 server. The two new SSD drives provide higher capacity and greater IOPS to store and handle object metadata transactions.

After FC 8083 is added, the IBM SSR must activate FC 5283 and run an enablement procedure from the TS7700 service window. Only an IBM SSR can activate FC 5283 by using the TS7700 service page at microcode level 8.52.200.x or later. The enablement process includes the following actions:

a. Place the TS7700 into service-prep and take it offline.
b. Enable the TS7700 Advanced Object Store on the TS7700 service window.
c. Bring the TS7700 back online.

### Configuring DS8000 IP addresses in TS7700

After FC 5282 or FC 5283 is installed, an SSR must enter the DS8000 primary and secondary (if one exists) IP addresses for each CEC into another TS7700 service window. This process can be done while the TS7700 is offline for the feature enablement process or after the TS7700 is fully online. For each DS8000, enter the following information:

► DS8000 Machine Serial Number
► DS8000 Primary IP address for CEC0
► DS8000 Primary IP address for CEC1
► DS8000 Alternate IP address for CEC0 (optional)
► DS8000 Alternate IP address for CEC1 (optional)

DS8000 IP addresses can be added, removed, or deleted by using the TS7700 service windows at any time.

### Resizing logical cache partitions

During either the FC 5282 or FC 5283 enablement process, a single logical cache partition is created for objects. This object cache partition is created using a default size of 3 TB. The cache partition can be resized to your requirements by using the Cache Partition page on the TS7700 MI. For more information about cache partitions, see Chapter 5, "TS7700 cache partitioning" on page 37.

### Data in-flight encryption

If a secure data transfer of objects between the DS8000 and TS7700 is required, FC 5281 Secure Data Transfer must be activated on any TS7700 that is a DS8000 Transparent Cloud Tiering target. More information about how FC 5281 Secure Data Transfer works between a DS8000 and TS7700 can be found under 6.4, "Encryption" on page 46.

### Object Policy management

After FC 5283 Advanced Object Store is installed, a new MI navigational icon that is labeled **Object Store** is listed. Select **Object Store** → **Object Policy** from the list. See Figure 3-4 on page 27.

*Figure 3-4   Object store icon options*

Create a new object policy by clicking **Create Object Policy** on the page and follow the instructions to create a policy. One default object policy also exists and can be modified for usage.

Next, create an object store by clicking **Object Store** → **Object Store**. See Figure 3-4. Click **Create Object Store** on the page. The *object store name* must match the name the user defined in the z/OS SMS Cloud Network Connection Construct page. For more information about the ISMF page, see 3.2.4, "z/OS Configuration" on page 30.

> **Note:** The names are case-sensitive. z/OS always lists the names in uppercase but the names are converted to lowercase in the DS8000. The names are always stored as lowercase in the TS7700 when this feature is used

An object policy can be chosen from a drop-down list. The object policy should match the policy behavior that is expected for this workload.

Examples of these MI pages can be found in Chapter 7, "TS7700 management interface" on page 51.

> **Note:** The TS7700 must go through the entire enablement process *before* the DS8000 is configured because during the DS8000 configuration the DS8000 attempts to communicate with the TS7700.

After the configuration settings are established, the TS7700 is ready to accept object data from the DS8000. Next, go to the DS8000 section and complete any needed configuration steps. For more information, see 3.2.3, "DS8000 configuration" on page 27.

### 3.2.3  DS8000 configuration

The following list describes the high-level steps for setting up DS8000 for transparent cloud tiering to a TS7700:

- ▶ Configure the Ethernet Ports.

- ▶ Configure to the cloud using `mkcloudserver`.

- ▶ If a second or more object store will be configured as part of DS8000 multi-cloud, the `managecloudserver -action applypndgconfig` command will need to be run after each `mkcloudserver` command.

- ▶ Network Considerations.

- ▶ Configuring DS8000 local user account and HMC for z/OS RESTapi control commands and metadata.

DS8000 verifies the connection path and tests that objects and containers can be successfully created and stored on to the TS7700 when the `mkcloudserver` command is used. If DS8000 encryption is used, the DS8000 verifies that the Secure Data Transfer feature is activated in the TS7700 and starts to communicate with the TS7700 by using secure transmissions.

When you configure the DS8000 transparent cloud tiering, you will use the DS8000 Command-Line Interface (DS CLI) or the DS8000 Storage Management GUI to create a local user account to support the DS8000 as a proxy for metadata and administrative commands from the z/OS.

## Configuring Ethernet ports

The DS8000 comes standard with 1Gb Ethernet ports that are reserved for transparent cloud-tiering traffic. Optionally, 10Gb Ethernet adapters can be installed for improved throughput capabilities. It is recommended to use 10Gb Ethernet for DS8000 Transparent Cloud tiering. Regardless of which, the ports must be configured before they can be used.

Only two ports per DS8000 CEC (total of 4 ports) are supported and should be either the 1Gb ports or the 10Gb ports. A mixture of 1Gb and 10Gb is not supported.

Use the `lsnetworkport` command from the DSCLI to display the current Ethernet configuration. Identify the Ethernet ports that will be used by the transparent cloud tiering feature and enable them by using the `setnetworkport` command.

Detailed instructions on identifying and enabling these ports are available in the *IBM DS8000 Transparent Cloud Tiering Guide*, SG24-8381 under the section labeled **Ethernet Configuration.**

### *Configuring to the cloud by using mkcloudserver*

To configure and verify connection to a TS7700, use the `mkcloudserver` command at the DSCLI. For more information about the various options for `mkcloudserver`, see mkcloudserver.

> **Note:** The following examples show the most commonly used options to configure to a TS7700. Each DS8000 Transparent Cloud Tiering user may require different options. When entering the `mkcloudserver` command, set the `-type` parameter **TS7700**.

Enter the TS7700 Grid link connection information by using the `mkcloudserver` command with the `-primary7700IPs` parameter. This parameter takes a list of up to four comma-separated Grid IP addresses from one TS7700. If two TS7700s are targeted for redundancy and load balancing purposes, use the optional `-secondary7700IPs` parameter and list the IP addresses for the second TS7700 cluster.

Example 3-1 on page 29 shows two examples of configuring to a TS7700.

*Example 3-1   Configure TS7700 as a cloud target with mkcloudserver command*

```
(single)
mkcloudserver -type TS7700 -primary7700IPs 1.1.1.1,2.2.2.2 -nossl
cloud_network_connection_name

(forking)
mkcloudserver -type TS7700 -primary7700IPs 1.1.1.1,2.2.2.2 -secondary7700IPs
3.3.3.3,4.4.4.4 cloud_network_connection_name
```

The first mkcloudserver example shows a configuration to a single TS7700 with two grid IP addresses, **1.1.1.1** and **2.2.2.2**. The **-nossl** option is specified, so secure data transfer is not used during transactions between the DS8000 and the TS7700. Last, the SMS Cloud Network Connection name is specified as ***cloud_network_connection_name*** in the example.

The second mkcloudserver example shows a configuration to two TS7700s. IP addresses **1.1.1.1** and **2.2.2.2** are defined for the first cluster and **3.3.3.3** and **4.4.4.4** for the second cluster. In this example, secure data transfer is used because **-nossl** is *not* specified. Again, the SMS Cloud Network Connection name is specified as ***cloud_network_connection_name***.

The DS8000 assigns the set of DS8000 IP addresses differently depending on whether one, two, three or four IP addresses are specified in the **-primary7700IPs** and **-secondary7700IPs** fields. The following table provides an understanding of how the IP addresses are assigned in each CEC (LPAR) in the DS8000 when using DS8000 R9.33.x and lower microcode levels. This can be seen in Table 3-2.

*Table 3-2   Port Assignments to CEC R9.33.x and Lower*

| Configuration | CEC0 | CEC1 |
|---|---|---|
| 1 port (1.1.1.1) | 1.1.1.1 | 1.1.1.1 |
| 2 ports (1.1.1.1, 2.2.2.2) | 1.1.1.1 | 2.2.2.2 |
| 3 ports (1.1.1.1,2.2.2.2,3.3.3.3) | 1.1.1.1,3.3.3.3 | 2.2.2.2 |
| 4 ports (1.1.1.1,2.2.2.2,3.3.3.3,4.4.4.4) | 1.1.1.1,3.3.3.3 | 2.2.2.2,3.3.3.3 |

In Table 3-2, if a user configures one port in the -**primary7700IPs** field, DS8k will assign it to both CECs. If a user configures two ports in -**primary7700IPs**, DS8k will split the IPs so that CEC0 uses the first port in the list and CEC1 uses the second port in the list. For three ports, CEC0 will use the first and third ports in the list and CEC1 will use the second. Last, if four ports are specified, DS8K will assign the first and third IP Addresses to CEC0 and the second and fourth IP Address to CEC1. This was done for better load balancing.

In DS8000 microcode level 9.4, changes were made to the two and three port configurations for greater redundancy. This can be seen in Table 3-3.

*Table 3-3   Port Assignments to CEC R9.4 and Higher*

| Configuration | CEC0 | CEC1 |
|---|---|---|
| 1 port (1.1.1.1) | 1.1.1.1 | 1.1.1.1 |
| 2 ports (1.1.1.1, 2.2.2.2) | 1.1.1.1, 2.2.2.2 | 1.1.1.1, 2.2.2.2 |
| 3 ports (1.1.1.1,2.2.2.2,3.3.3.3) | 1.1.1.1,2.2.2.2,3.3.3.3 | 1.1.1.1,2.2.2.2,3.3.3.3 |
| 4 ports (1.1.1.1,2.2.2.2,3.3.3.3,4.4.4.4) | 1.1.1.1,3.3.3.3 | 2.2.2.2,3.3.3.3 |

In Table 3-3, the assignment of ports on CEC0 and CEC1 is different for two and three port configurations beginning in DS8000 microcode level R9.4. Both ports in the two-port configuration and all three ports on the three-port configuration will be used by both CECs. The one and four port configurations remain the same.

> **Note:** When using -**secondaryTS7700IPs**, the assignment pattern is the same for the second set in all code levels.

## 3.2.4  z/OS Configuration

Different steps are required for setup in z/OS depending on which host application is used for transparent cloud tiering. This section provides a few high-level, common configuration steps for transparent cloud tiering and common steps to set up DFSMShsm and DFSMSdss. Other configuration steps might be required depending on other DFSMS applications. Review the *IBM DS8000 Transparent Cloud Tiering Guide*, SG24-8381 and the z/OS DFSMS Transparent Cloud Tiering Publication Updates (includes OA51622).

### Cloud Network Connection Construct

As part of the z/OS transparent cloud tiering offering, a new Cloud Network Connection Construct was introduced. A Cloud Network Connection Construct will need to be defined in the ISMF Cloud window (option "S") in the SMS source configuration data set

To configure the cloud network connection, consider the following points:

► Set up any digital certificate requirements for transactions between the z/OS and DS8000 for transparent cloud tiering.

► Configure the Cloud Network Connection Construct in z/OS by using the ISMF Cloud window (option S) in SMS. Various fields are required including:

  – Create a Cloud Network Connection Construct Name. The construct name is used when you run the DS8000 `mkcloudserver` command and on the TS7700 Object Store MI page.

  – For objects to be moved to a TS7700 Object Store, TAPE-OBJECT must be specified in the Provider field.

  – The HMC endpoint will have to be entered as well. This was discussed in 3.2.1, "Network considerations" on page 22.

► See 2.3, "DS8000 supported configurations" on page 17 for a list of z/OS transparent cloud tiering documents that contain instructions on setting the other fields.

Figure 3-5 is an example of the Cloud Network Connection Construct panel that is found under the ISMF Cloud window.
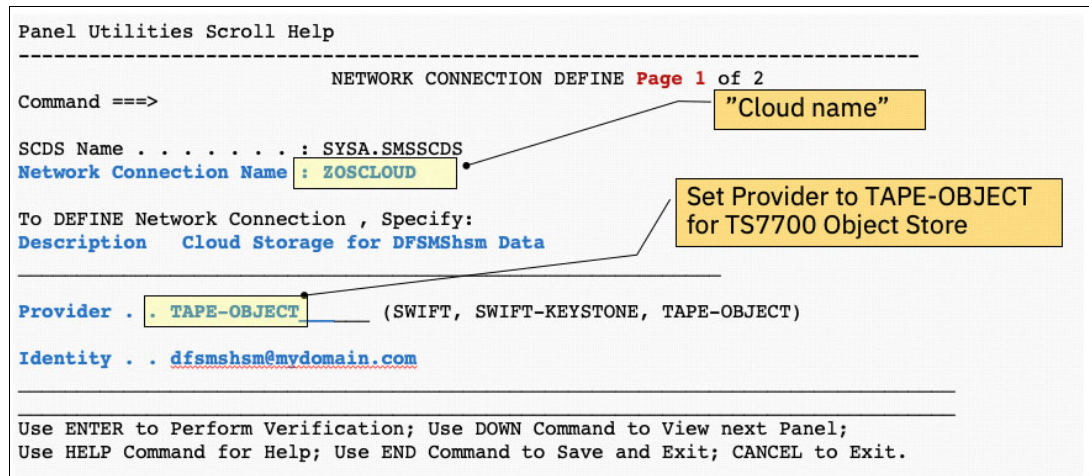
```
Panel Utilities Scroll Help
-----------------------------------------------------------------------------
                          NETWORK CONNECTION DEFINE Page 1 of 2
Command ===>

SCDS Name . . . . . . . : SYSA.SMSSCDS
Network Connection Name : ZOSCLOUD

To DEFINE Network Connection , Specify:
Description    Cloud Storage for DFSMShsm Data

_____

Provider . . TAPE-OBJECT_____  (SWIFT, SWIFT-KEYSTONE, TAPE-OBJECT)

Identity . . dfsmshsm@mydomain.com
_____
_____
Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

*Figure 3-5   Network Connection Name and Provider fields in the ISMF cloud window*

> **Note:** DFSMS supports a network connection name with a combination of alphanumeric and numeric characters and three special characters: $,@, and #. However, if the target object store is a TS7700, only the @ character is supported. Using a cloud name that includes the characters $ and # results in a failure during DS8000 configuration of a TS7700.

## DFSMShsm data set level migration

If you use DS8000 Transparent Cloud Tiering for the `DFSMShsm MIGRATE` command, new Management Class options are available for transparent cloud tiering, and a new migration level, Migration Level Cloud (MLC), is introduced. This section provides information about these new settings and level.

When using DS8000 Transparent Cloud Tiering for `DFSMShsm MIGRATE` command, a Management Class policy must be configured. In SMS Management Class, a new `Level 2 Days Non-usage` attribute is used for direct migration to the cloud storage for a *cloud_network_connection_name*. It supports two values: `0` and `NOLIMIT`.

► When `0` is specified, the data set migrates to cloud storage (MLC) when the `Primary Days Non-usage` value is met if the data still is on Level 0.

► When `NOLIMIT` is specified, the data set does not migrate to cloud storage and migrates to Level 2 (ML2) based on the value of `Level 1 Days Non-usage`.

`Level 2 Days Non-usage` takes priority over `Level 1 Days Non-usage`. DFSMShsm does not support the movement of migration copies from ML1 or ML2 to MLC.

New migration actions can also be made based on the size of the data set (in tracks) if the `Primary Days Non-Usage` criterion is met. `Size Less Than or Equal To` and `Size Greater Than` are optional fields that specify a size threshold.

Studies show that the larger the data track size, the more MIPS savings is realized. The new migration options are provided as another way to tailor migrations to fit the needs of each user. Very small tracks have little MIPS savings and are better left on the primary disk storage system.

Figure 3-6 shows the number of MIPS that is required for data movement by size during the z/OS DFSMS backup and archival process.
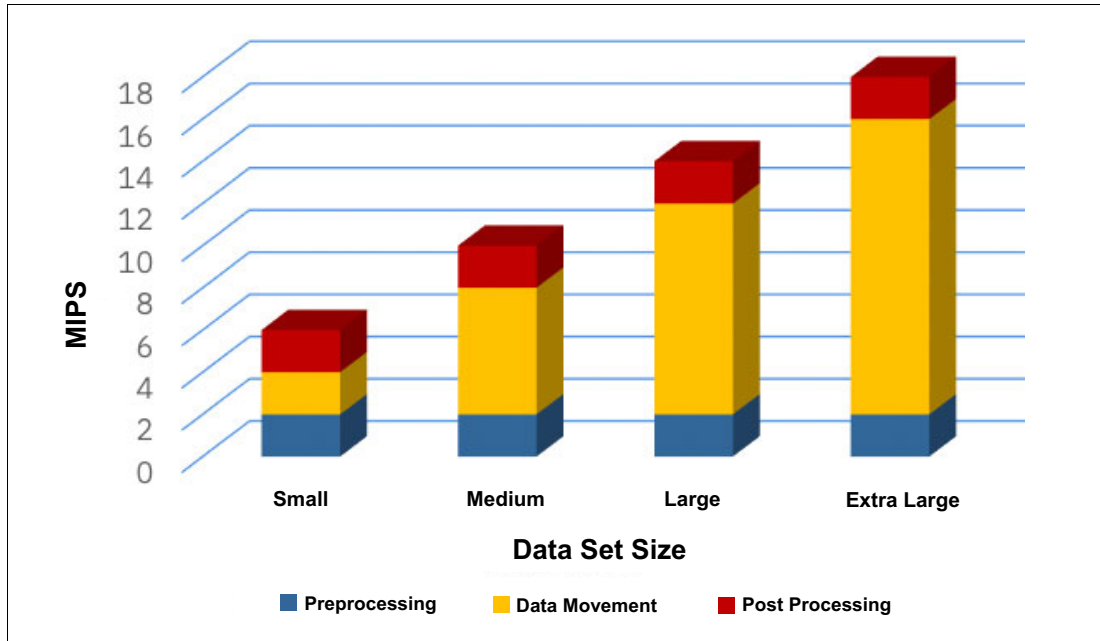


*Figure 3-6   Number of MIPS by data set size using ML2*

When the z/OS with DS8000 transparent cloud tiering solution is used, the number of MIPS in the z/OS remains low and constant, as shown in Figure 3-7.
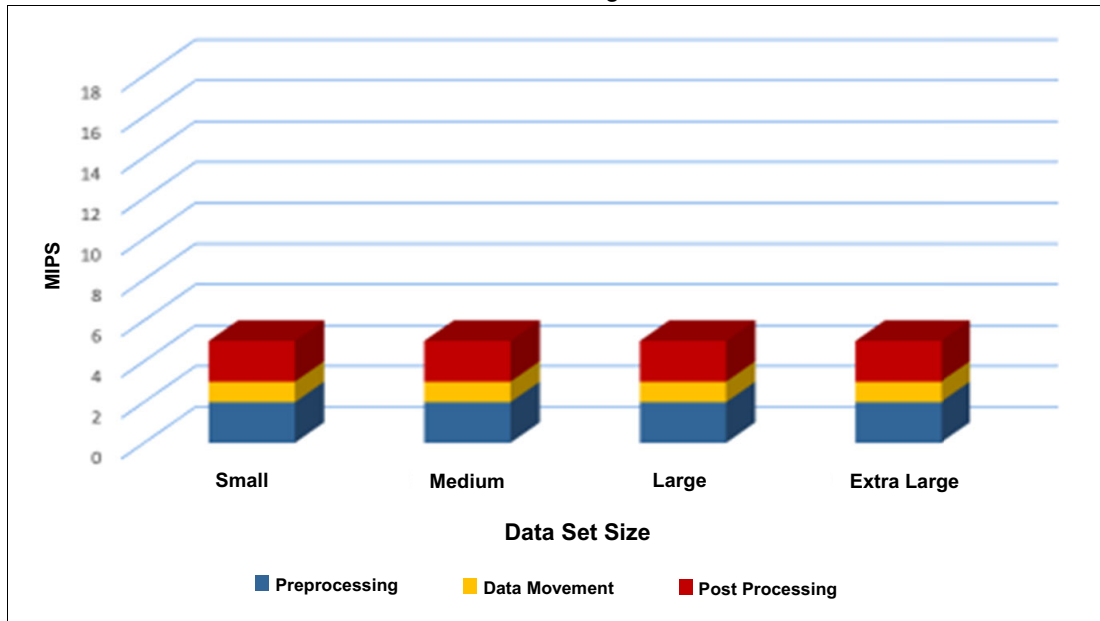


*Figure 3-7   Number of MIPS by data set size by using MLC*

You can choose to move all data sets larger than a certain size through the `CLOUD` option to gain the most MIPS benefit and leave smaller data sets to follow another path such as `ML2`, or remain on primary storage.

After the Management Class migration criteria are met by either days of non-usage or size requirement and if the **CLOUD** action is specified, then DFSMShsm requests the data movement through the DS8000.

The SMS Cloud network connection construct contains a list of cloud attributes and values that are used by DFSMShsm and DFSMSdss commands to manage data set migration. **DFSMShsm MIGRATE** and **DFSMShsm HMIGRATE** commands now include the **CLOUD** keyword. When this keyword is specified, the data sets are migrated to the requested cloud. The *cloud_network_connection_name* must match an SMS Cloud definition that is shown in Figure 3-5 on page 31.

## DFSMSdss Full Volume Dump

z/OS also supports the ability to issue native DFSMSdss Full volume dump to a TS7700 using dss **DUMP** and **RESTORE** commands. Native dss users specify the cloud network connection construct name (cloudname) container and object prefix information. An example **DUMP** command is shown in Example 3-2.

*Example 3-2   DFSMSdss Full Volume Dump command*

```
DUMP INDDNAME(DISK1) CLOUD(cloudname)
   CCREDS(password) CONTAINER(DSSDUMPS)
   OBJECTPREFIX(objectprefix1) WAIT(0,0)
   ALLDATA(*) ALLEXCP
```

For more information about the **DFSMSdss DUMP** command, see DUMP FULL command syntax.

DFSMSdss also provides additional function to list and delete dump information by using **CLOUDUTILS LIST** and **CLOUDUTILS DELETE** commands. These commands are controlled by a new facility class profile.

**CLOUDUTILS LIST** provides the ability to list all containers within a cloudname, list all dumps within a container or list one or more dumps in a container by the object prefix name. The following is an example of the **CLOUDUTILS LIST** command format (Example 3-3):

*Example 3-3   DFSMSdss CLOUDUTILS LIST*

```
CLOUDUTILS LIST CLOUD(cloudname) CONTAINER(DSSDUMPS)
```

**CLOUDUTILS DELETE** provides the ability to delete one or more dumps within a specified container or delete an empty container. The following is an example of the **CLOUDUTILS DELETE** command format (Example 3-4):

*Example 3-4   DFSMSdss CLOUDUTILS DELTE*

```
CLOUDUTILS DELETE CLOUD(cloudname) CONTAINER(DSSDUMPS)
```

For more information, see about the **CLOUDUTILS** command, see CLOUDUTILS command for DFSMSdss.

For more information about these and other z/OS transparent cloud tiering settings, see "Online resources" on page 95. For additional information about DSS Full Volume Dump, see DFSMSdss Full Volume Support for Transparent Cloud Tiering.

### Transparent cloud tiering compression

If DS8000 compression is used, then enable `TCTCOMPRESSION` in DEVSUPxx parmlib to compress the object going to a TS7700. Data that is already compressed by using zEDC and stored in the DS8000 does not need to be compressed. With this setting enabled, DFSMS will inform DS8000 when it needs to compress data. For more information about compressed objects going to the TS7700, see 6.3, "Compression" on page 45.

For a list of DFSMS applications that currently support DS8000 Transparent Cloud Tiering see 2.5, "IBM Z host considerations" on page 18.

**4**

# Transparent cloud tiering object data flow

This chapter provides an overview of the transparent cloud tiering object data flow.

After z/OS, the DS8000, and the TS7700 are configured, DFSMShsm or DFSMSdss data is ready to be exported to the TS7700 in the form of objects.

For DFSMShsm automatic data set migration, data begins migrating if the following conditions are met in the Management Class specified:

▶ `Level 2 Days Non-usage` is set to `0`.

▶ `Primary Days Non-usage` value is met.

▶ Data sets in level 0 assigned to a Management Class that targets cloud as a migration tier.

▶ `Size Less Than` or `Equal To and Size Greater Than` criteria must be met if the optional fields are used.

DFSMShsm starts the migration of the data to be processed by DS8000. DS8000 sends the data (in the form of objects) to the TS7700 indicated by the *cloud_network_connection_name* by using a `PUT` command.

When a recall is needed, DFSMShsm starts a restore of the data set by way of the DS8000 by using a `GET` command to the TS7700.

**35**

If the Management Class `Level 2 Days Non-usage` is set to `NOLIMIT`, the data set is migrated based on the values of `Level 1 Days Non-usage`, `Size Less Than or Equal To`, and `Size Greater Than`. Data sets that are migrated to ML2 storage are otherwise handled the same as they are now (as logical volumes). Figure 4-1 shows the migration paths.
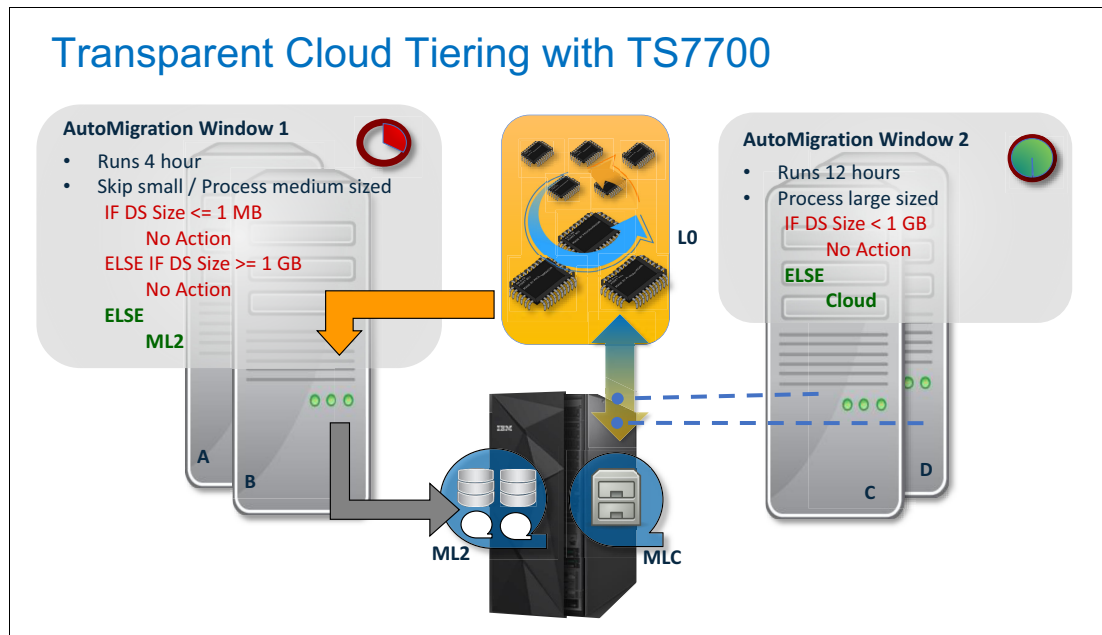


*Figure 4-1   ML2 and MLC Auto Migration paths*

Since TCT object data movement no longer occurs through the mainframe, automatic migration windows can be extended since this is no longer a concern of high MIPS usage. This can be seen in Figure 4-1. It is recommended to keep very small tracks on the DS8000, small-medium tracks that have little impact to host MIPS continue to target tape (ML2), and use TCT for medium to large tracks. This provides the most efficient and impactful configuration.

DFSMSdss full volume dumps can be started by using a batch JCL routine natively. A native DSS user creates a container name and object prefix, indicates the volume to dump, and specifies the target Cloud and other parameters. When the `DUMP` command is run, it starts the dump process with the DS8000 using `PUT` commands. A DSS user can also run commands to `LIST` or `DELETE` dumps and containers.

The DS8000 reads and writes data to and from the TS7700 by using existing Ethernet interfaces on both systems. On the TS7700, this is the grid links and can be 1 Gbps or 10 Gbps Ethernet adapters. The DS8880 supports 1 Gbps Ethernet with an optional upgrade to 10 Gbps adapters. The DS8900 supports 10 Gbps adapters.

# TS7700 cache partitioning

This chapter includes a discussion of aspects of cache partitioning as it pertains to the TS7700.

This chapter includes the following sections:

- ▶ 5.1, "Origin of logical cache partitions" on page 38
- ▶ 5.2, "Logical cache partitions for cloud migration" on page 38
- ▶ 5.3, "Logical cache partitions for objects" on page 38

## 5.1 Origin of logical cache partitions

TS7700 Logical Cache Partitioning is supported on microcode release level 8.32.x.x and later. Cache partitioning has two main purposes.

The first purpose of cache partitioning is to separate data into logical *buckets*. What those buckets symbolize depends on business models and needs. For example, the buckets can be used to separate data for a sales division versus a research division of a company, or if working in a multi-tenancy structure, the data can be separated by clients.

The second purpose of cache partitioning is to migrate data to a lower tier of storage for long-term archiving or backup purposes. At the time of writing, the lower tier of storage can be a physical tape library, or cloud storage, for example IBM Cloud® or Amazon AWS.

In the TS7700, Storage Class constructs are created to set Tape Volume Cache preferences for a specific cache partition. These preference levels determine how soon volumes are removed from cache after they are copied to tape.

The Storage Class then targets a cache partition. Setting a Tape Volume Cache preference that targets a cache partition, along with other construct policies, allows for customization of data handling depending on the need. For instance, the Sales division of a company might want their data to remain in cache as long as possible for faster access, and the Research division can store most of their data into a more cost-efficient storage medium for archival purposes.

Eight Logical Cache Partitions are available: cache partition 0 (CP0) through cache partition 7 (CP7). CP0 is reserved for resident-only logical volume data, which is data that comes over the IBM FICON channel as logical volumes; is expected to remain in disk cache indefinitely; and is not copied to physical tape.

Before microcode level 8.42.x.x, CP1–CP7 were activated with FC 5274 (Tape attachment) only. These cache partitions are managed in the MI Cache Partition window and assigned to logical volumes through the STORCLASS construct. The window was viewable only on a tape-attached TS7700.

## 5.2 Logical cache partitions for cloud migration

When using TS7700 microcode level 8.42.x.x or later, partitions CP1–CP7 can also be used to export logical volume data to a cloud system through the TS7700 Cloud Storage Tier offering. Therefore, the Cache Partition window is also visible when Cloud Storage Tier FC 5278 is activated to manage tiering data to the cloud behind the TS7700.

## 5.3 Logical cache partitions for objects

With the introduction of the DS8000 Object Store feature in the TS7700 8.50.x.x microcode release, cache partitions again are used to separate data, although with some minor differences. In this initial release, only one cache partition is available for object data usage. This cache partition is created during the DS8000 Object Store or TS7700 Advanced Object Store enablement process, as described in 3.2.2, "TS7700 configuration" on page 25, with a default size of 3 TB.

The object partition includes the following rules and restrictions:

- An object partition can be set as *primary object partition,* and one primary object partition must exist if DS8000 Object Store or Advanced Object Store is enabled. The primary object partition is denoted by a green star to the left of the partition number on the Cache Partition window, as shown in Figure 7-6 on page 54. The concept of having a primary object partition is similar to the primary tape or cloud partition that is used today. However, in microcode level 8.50.x.x, primary object partitions are symbolic because only one object partition is used.

- One inactive cache partition must be available for the DS8000 Object Store or Advanced Object Store feature to be enabled. If all cache partitions are used for logical volume or Cloud Storage Tier workload, the enablement fails.

- This object partition cannot be deleted as one partition must always be available for object data processing.

- A second object partition cannot be created. Selecting **Create Partition** on a tape or cloud attached TS7700 creates a virtual volume partition but not an object partition. If the TS7700 model is not connected to tape or the cloud, the Create Partition option is disabled.

- An object partition can be resized. Increasing the object partition size of partitions CP1–CP7 decreases the CP0's available space. Decreasing the object partition size returns the space back to CP0. Enough space must be available in CP0 (minus any space that is taken because of IBM FlashCopy operations) to meet the new allocation size of a tiered partition.

   **Note:** CP0 available space cannot be less than 2 TB.

- An object partition can be renamed.

For more information about the MI Cache Partition window, see 7.4, "Cache partition" on page 54.

One new concept was introduced with object partitions. Object partitions are divided into even smaller buckets inside the TS7700. These buckets are separated by the z/OS `cloud_network_connection_name`. This happens when an object store is created in the TS7700 Object Store MI page for FC 5283 Advanced Object store. Each Object Store created on the MI creates a separate file set in the TS7700 cache. The z/OS host application will send the `cloud_network_connection_name` on each s3 command and this name is matched up with the Object Store name already created. This places the object into the appointed object partition.

Beginning with DS8000 R9.2.x, one DS8000 can be configured to more than one object stores (`cloud_network_connection_name`) and each of these object stores can target one or more TS7700 clusters, including the same TS7700 cluster. Also, more than one DS8000 system can write data to a TS7700. If more than one `cloud_network_connection_name` is configured to a TS7700 cluster, the TS7700 creates separate object store buckets inside the object cache partition. Figure 5-1 shows CP1 (an object partition) has two values for `cloud_network_connection_name`: Cloudname1 and Cloudname2.
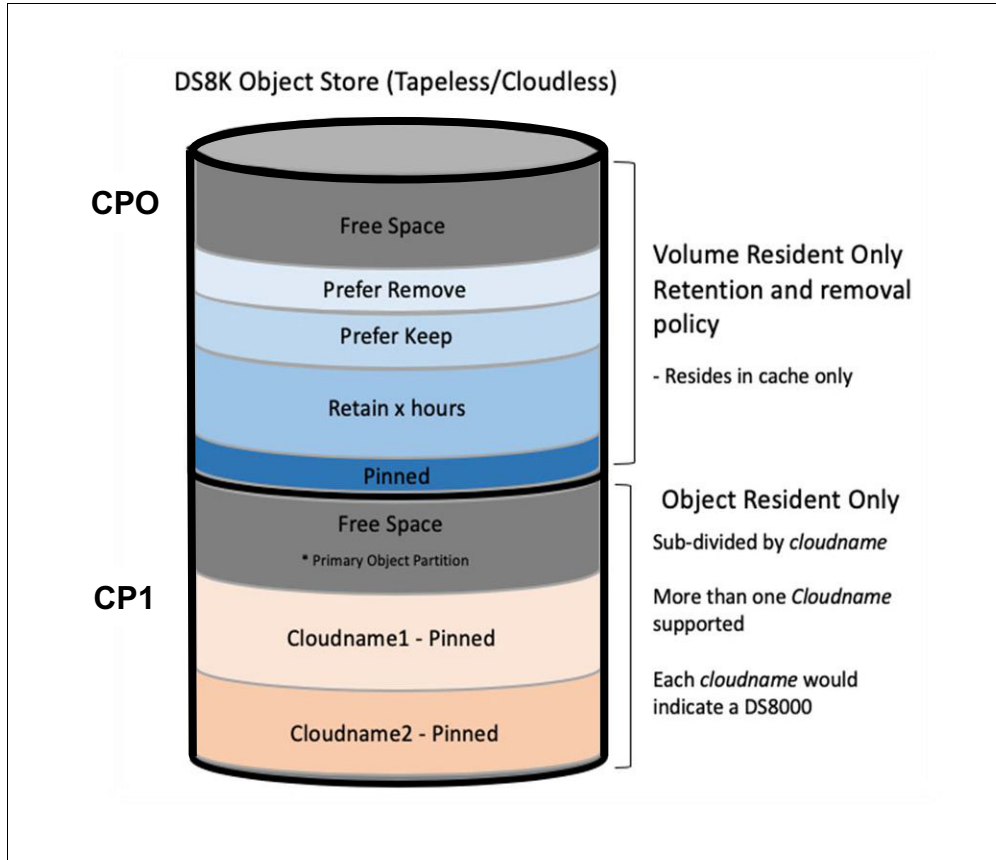


*Figure 5-1   TS7700 object store partition*

Dividing a cache partition by assigning a different value to `cloud_network_connection_name` allows more flexibility for object data management, such as setting different replication policies and providing separate statistical data for different workloads.

Figure 1-6 on page 14 and section 9.4, "VEHSTATS statistical reports" on page 73 describes the additional statistical data available for DS8000 Transparent Cloud Tiering. This statistical data is available beginning 8.52.200.x code and higher.

If workloads are not related between DS8000s, it is recommended to use a different `cloud_network_connection_name` for each DS8000. This avoids data collisions if the two systems happen to use the same container and object name for two different workloads.

**6**

# TS7700 object management

The TS7700 is known for its advanced policy management capabilities. For Tape logical volume data, Data Class, Management Class, Storage Class, and Storage Group policies have been available since the origination of the TS7700. These policies cannot be assigned to objects. Beginning microcode level 8.52.200.x and later, the TS7700 includes replication policy with the Advanced Object Store feature (FC 5283).

This chapter describes some of the current data management policies and how they affect object data management.

This chapter includes the following sections:

# 6.1 Tiering objects

The term *tiering* is used when the TS7700 migrates data to a lower tier of storage such as a physical tape library or an object store. The term *resident-only* is used when data is not migrated to a lower tier of storage but, instead, remains resident in the TS7700 disk cache until it is scratched or deleted.

DS8000 Transparent Cloud Tiering objects are resident-only. That is, objects remain in the cache and cannot be copied or moved to physical tape or to a cloud device that is attached to the TS7700 as part of the Cloud Storage Tier offering. Only tape logical volumes can be migrated to a lower tier of storage. Therefore, Storage Group and Storage Class options for data migration are not available for object data.

# 6.2 Object replication

Depending on which feature code is enabled, additional copies of object data can be placed on a TS7700 cluster by the DS8000 through a forking mechanism or by the TS7700 through object replication. The following sections describe how each feature code handles replicating objects within a TS7700 grid.

## 6.2.1 FC 5282 enabled

The DS8000 can connect one or two TS7700s to migrate object data. If two TS7700s are targeted, a synchronous copy is performed to both clusters by using a forking mechanism. Both clusters must be in the same TS7700 grid.

The forking process for synchronous copy is performed at the DS8000 level and does not use any TS7700 grid replication process (Management Class or Object Policy). TS7700 managed replication is not supported by the FC 5282 feature.

A maximum of two TS7700s can be targeted by a single DS8000. Multiple DS8000s can target the same cluster or different clusters in a grid as shown in Figure 6-1 on page 43. Those TS7700s must be at microcode level 8.50.x.x or later with FC 5282 enabled and can be a part of a larger grid.
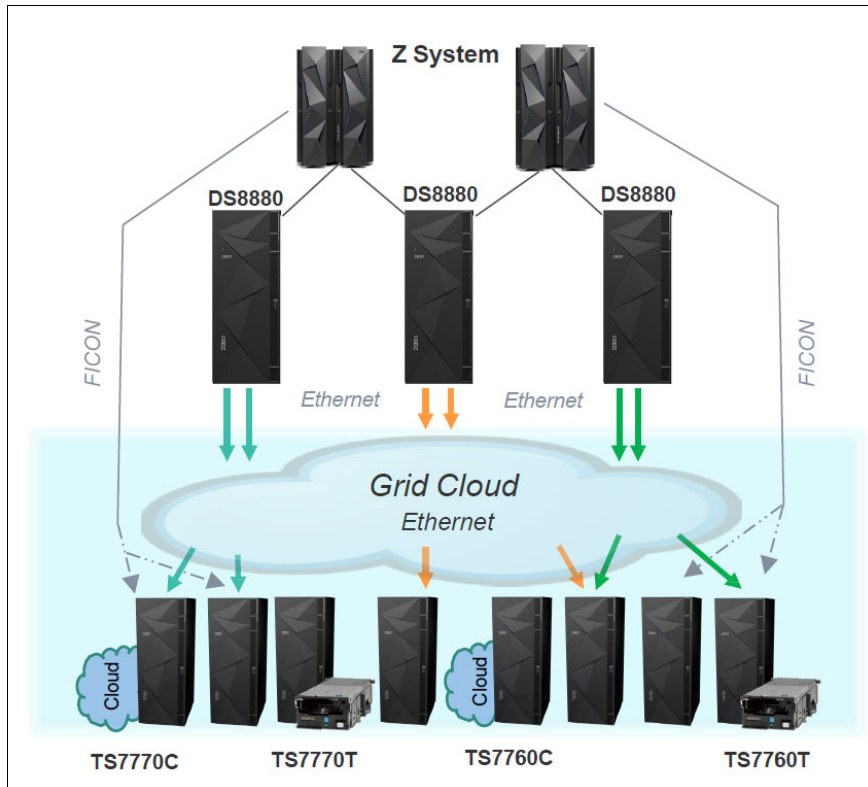
*Figure 6-1   IBM Cloud Synergy DS8000 object store*

If a DS8000 is configured to migrate to one TS7700 and object data is written, it cannot be changed to forked copies later. All data must be removed and the DS8000 reconfigured to set up forking.

## 6.2.2 FC 5283 enabled

The DS8000 can migrate object data to one or two TS7700s. If two TS7700s with FC 5283 Advanced Object Store are targeted, the DS8000 balances the load between the two target clusters. Both clusters must be in the same TS7700 grid. DS8000 no longer fork writes to two clusters like it does for FC 5282.

Additionally, copies of those objects can be distributed among any combination of grid members by using Object Policy settings in the TS7700. Any TS7700 that requires a copy of the objects must have FC 5283 installed.

Object replication policies are created at the Management Interface (MI) by using the Object Policy page. There are three options for the copy handling for each FC 5283 enabled cluster in the grid. These options are selected when creating or editing a specific policy and include synchronous (must select at least 2 clusters), deferred and no copy.

Figure 6-2 is an example of the Create Object Policy wizard from the Object Policy page.
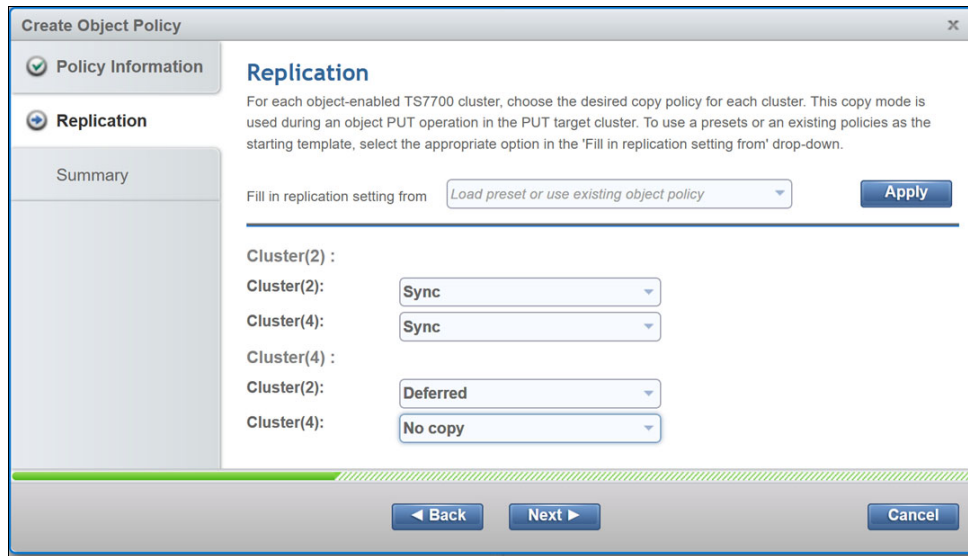


*Figure 6-2   Create object policy Replication GUI page*

The Policy can be set to run different copy behaviors based on which cluster the data was sent to by the DS8000. For example, in Figure 6-2, if the object is sent to Cluster(2), the copy action would be to create a synchronous copy on both cluster(2) and cluster(4). However, if the object is sent to cluster(4) the copy action would be to create a copy on cluster(2) and no copy on cluster(4).

Policies are assigned to each defined *cloud_network_connection_name* in the TS7700 Object Store page as shown in Figure 6-3.
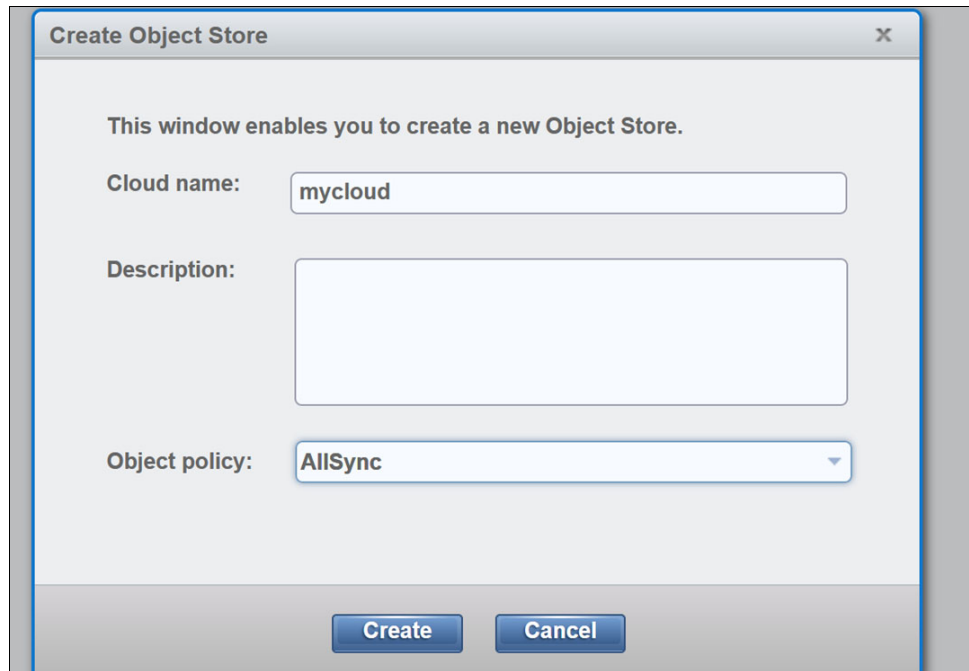


*Figure 6-3   Create Object Store GUI*

In this example, policy **AllSync** is assigned to cloud name **mycloud**. The TS7700 can be configured with multiple values for the *cloud_network_connection_name* from the same DS8000, each with a different object policy to trigger different replication behaviors. This is useful when one workload requires synchronous copies but another needs only asynchronous copies.

## 6.3  Compression

DS8000 uses zlibgNX hardware compression for object data going to a TS7700. This is only supported on DS8900 models and uses IBM POWER9 Hardware Accelerated Compression for low resource usage.

`If compression will be used for TCT,` `TCTCOMPRESSION` must be enabled in z/OS. DFSMS controls whether DS8900 compresses the object data or not. The following is information about object data compression.

► If the data is already compressed and stored in the DS8000 by using zEDC, DFSMS will inform DS8000 not to compress.

► If the data is already compressed and stored in the DS8000 using IBM z15® compression, DFSMS will inform DS8000 not to compress.

► If the data is encrypted by the host by using data set level encryption, DFSMS does not ask the DS8900 to compress the data as encrypted data does not compress well and can potentially inflate the data.

► Data that is compressed or encrypted by the z/OS host or by DS8000 remains compressed on the TS7700. The TS7700 does not attempt to compress the data further.

- ► Object data from a DS8880 model is not compressed.
- ► TS7700 software compression through Data Class is not available for DS8000 object data.

# 6.4 Encryption

The TS7700 supports encryption for data at rest by using FC 5272 (Disk Encryption with Local Key Management) or FC 5276 (Disk Encryption with External Key Management). DS8000 object data that is written to these systems benefits from these disk encryption features if they are enabled.

Also, object data in flight between a DS8900 and a TS7700 can be encrypted by using Secure Data Transfer functions. To encrypt object data, all target TS7700s require FC 5281 Secure Data Transfer be activated. The activation of this feature is a concurrent activity and is done using the TS7700 Management Interface Feature License window.

After this feature is activated, the DS8000 must be configured again. This means that `mkcloudserver` must be run without the `-nossl` flag. For more information about configuration, see 3.2.3, "DS8000 configuration" on page 27.

When `mkcloudserver` is run without the `-nossl` flag, the DS8000 validates that FC 5281 is activated on the TS7700 specified. If it is not, transparent cloud tiering configuration fails.

Only DS8900 models support Secure Data Transfer to a TS7700. The DS8900 uses the Power9 Crypto Accelerator for maximum performance. After the DS8900 is configured for encryption, it requests a secure data connection with the TS7700 by using TLS1.2 with an AES 256-bit key. Data arriving at the TS7700 is decrypted using TLS. To ensure that data at rest is secure, use disk encryption on the TS7700.

When data is recalled by the z/OS host, the DS8000 again requests a secure data connection with the TS7700 using TLS1.2 with an AES 256-bit key.

The TS7700 will also use the Power9 Crypto Accelerator to maximize performance.

The TS7700 also uses FC 5281 Secure Data Transfer for data transfers between clusters in a grid. A TS7700 acting as a client allows a user to enable and disable Secure Data Transfer or change between AES 128-bit and AES 256-bit key sizes. A TS7700 acting as a server supports Secure Data Transfer for both key sizes when FC 5281 is activated on the system. For DS8000 to TS7700 communication, the DS8000 is always the client and the TS7700 is always the server. Therefore, A TS7700 user that disables Secure Data Transfer or changes the encryption key size has no effect on the DS8000 to TS7700 transfer because the TS7700 server does not initiate the connection and will always support encryption. The only requirement is that FC 5281 be installed on the TS7700.

Secure Data Transfer uses an internal HTTPS SSL certificate on both DS8000 and TS7700. A user can replace the SSL certificates with their own self-signed certificate. On the TS7700, this is done on the SSL Certificates MI page under the **Access navigation** icon, as shown in Figure 6-4.
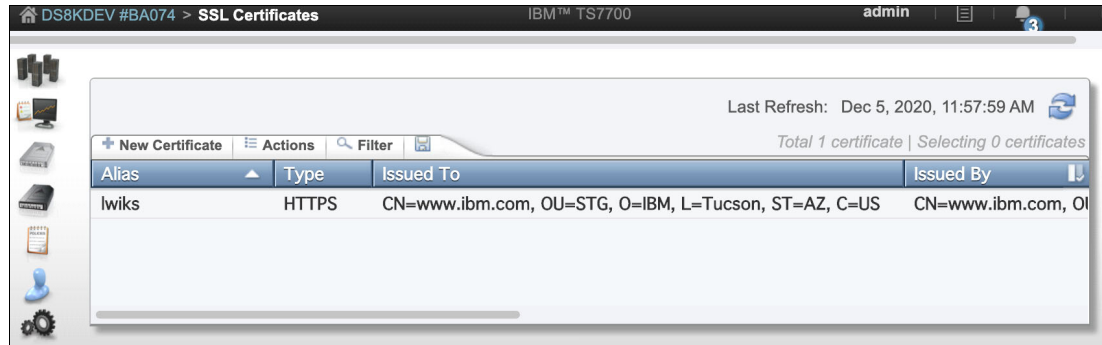


*Figure 6-4   MI SSL Certificates window*

A default certificate with the alias name of *Iwiks* is supplied with every TS7700. Select this certificate. Open the **Actions** dropdown menu and click **Replace**. A new user certificate can be uploaded in the window provided.

On the TS7700, the *Iwiks* certificate is used for several other operations such as MI login, Secure Data Transfer of data between TS7700 clusters in the Grid, and for LDAP. Because the TS7700 Grid is constantly synchronizing metadata between clusters, it also updates the certificate to all TS7700 clusters when it is replaced. This replacement cannot happen between a TS7700 and DS8000.

When replacing the *Iwiks* certificate on any TS7700 where DS8000 is targeting one of the clusters in that Grid, and encryption is enabled between the DS8000 and the TS7700, the following tasks must happen:

1.  All DS8000s targeted clusters in that grid are required to have their transparent cloud tiering services stopped.

2.  A new public key that matches the new certificate must be installed.

3.  The DS8000 can be reconfigured using the `mkcloudserver` command `-rootcaloc` parameter. The `-rootcaloc` is followed by the path and file name of the public certificate.

> **Important:** Install the DS8000 public certificate before uploading the new SSL Certificate to any TS7700 in the grid. Omitting this step results in DS8000 Transparent Cloud Tiering job failures.

For more information about configuring the DS8000 to use Secure Data Transfer, see *IBM DS8000 Transparent Cloud Tiering: DS8000 Release 9.3*, SG24-8381.

## 6.5  Removal policies

The TS7700 removal policy, which is known as Volume Copy Retention Group, provides a way to remove data from a disk-only TS7700 as active data reaches full capacity. Volumes become candidates for removal if enough copies exist on peer clusters within a Grid and a specified retention period elapses.

This removal policy is *not* supported for DS8000 Transparent Cloud Tiering object data. Object data can be removed from TS7700 cache by only the original z/OS host application that wrote it.

> **Important:** Ensure that the object cache partition does not run out of allocated space. Running out of object partition space causes job abends. Use the `Library Request` `RSDOLOW/RSDOHIGH` to be notified when an object partition reaches certain threshold levels.

## 6.6  Disaster recovery

### With FC 5282 enabled

If a DS8000 is targeting two TS7700s by using forking, and one of the TS7700s becomes unavailable, the DS8000 continues to write to the available TS7700. DS8000 sends a system health message to the z/OS SYSLOGs about the out-of-sync condition and starts cataloging any transactions that were not completed.

After the TS7700 becomes available, DS8000 synchronizes the object data. When both TS7700s are back in sync, DS8000 sends another system health message to the z/OS SYSLOGs to indicate that the out-of-sync condition is cleared. If the DS8000 cannot communicate with either TS7700, the job fails.

The TS7700 features procedures to add new clusters to a grid, merge clusters that contain data into a grid, or remove (unjoin) a cluster from a grid. A TS7700 can also go offline for a code upgrade or other scheduled outages. It is recommended to wait until the out-of-sync condition is cleared before any of these activities are performed. Activities, such as unjoining a cluster from a grid that contains object data, can result in unexpected data loss if data is out of sync.

> **Note:** The BVIR copy audit function does not detect data inconsistencies for DS8000 objects.

### With FC 5283 enabled

If a DS8000 is targeting two TS7700s for load balancing, and one of the TS7700s becomes unavailable, the DS8000 continues to write to the available TS7700. If the failure occurs while establishing connection to a TS7700 cluster, DS8000 attempts reasonable connection retries by using other potential communication paths. If all communication paths are down, the command can be sent to the second cluster. If the failure occurs after a connection was established and write operations have already begun, the operation will fail back to the host, so that proper cleanup can occur.

For a cluster that is a target of a DS8000, copy policy settings determine where object copies reside. If a copy target cluster is unavailable, it will reconcile all consistency tokens after it returns to an online state. During this process, the TS7700 cluster detects whether a copy is required and adds it to its own copy queue. For more information about consistency tokens, see 1.3.3, "Object, container, and token handling within the TS7700" on page 10.

The TS7700 features procedures to join new clusters to a grid, merge clusters that contain data into a grid, or remove (unjoin) a cluster from a grid. A TS7700 can also go offline for a code upgrade or other scheduled outages. It is recommended that all copies are reconciled before any of these activities are performed. Activities, such as unjoining a cluster from a grid that contains object data, can result in unexpected data loss if data is out of sync.

Beginning TS7700 microcode level 8.52.200.x, BVIR OBJECT COPY AUDIT is available to verify copy consistency in the grid. This BVIR command should be run before a TS7700 cluster with DS8000 Transparent Cloud Tiering objects is unjoined from the grid.

## 6.7  Throttling

You can modify settings on the T7700 by using the TS7700 host command line requests to slow down, or *throttle*, host IO, offload to tape, or various copy activities based on different triggers. The user can modify settings for when to trigger throttling and how much throttling to trigger (in milliseconds) through Library Request (LI REQ) commands.

At microcode level 8.52.200.x and later, the TS7700 supports deferred copy throttling (DCT) when the object copy queue begins to backup. The TS7700 uses the same DCT settings as those used for the tape logical volume copy queue and combines the overall statistics for inbound copies. The following are the LI REQ keywords support:

| Keyword 1 | Keyword 2 | Keyword 3 | Keyword 4 | Comp | Dist | 7740 | 7700D | 7700T | 7700C |
|-----------|-----------|-----------|-----------|------|------|------|-------|-------|-------|
| SETTING | THROTTLE | COPYFT | ENABLE/DISABLE | N/A | Y | Y | Y | Y | Y |
| SETTING | THROTTLE | DCOPYT | Value | N/A | Y | Y | Y | Y | Y |
| SETTING | THROTTLE | DCTAVGTD | Value | N/A | Y | Y | Y | Y | Y |
| SETTING | THROTTLE | DCTCURTD | Value | N/A | Y | Y | Y | Y | Y |

*Figure 6-5   shows the supported LI REQ Throttle keywords*

To better understand these settings and their usage, see TS7700 Library Request Command.

**7**

# TS7700 management interface

This chapter provides an overview of the changes to the management interface (MI) for the DS8000 Object Store or Advanced Object Store feature.

This chapter includes the following sections:

- ► 7.1, "Cluster summary" on page 52
- ► 7.2, "Historical summary" on page 52
- ► 7.3, "Cache utilization" on page 53
- ► 7.4, "Cache partition" on page 54
- ► 7.5, "Creating a storage class" on page 55
- ► 7.6, "Backup and restore settings" on page 55
- ► 7.7, "Object Policy" on page 56
- ► 7.8, "Object Store" on page 58
- ► 7.8, "Object Store" on page 58
- ► 7.9, "Dual Control" on page 60

# 7.1 Cluster summary

The cache tube widget displays a summary of how cache is used within the TS7700. In clusters with no tape or cloud attachment, the cache tube provides a breakdown of cache by preference groups (prefer-keep, prefer-removal, and pinned). In clusters with tape or cloud-attached, the cache tube provides a breakdown of cache, by cache partition.

When you use microcode level 8.50.x.x or later, if DS8000 Object Store is enabled, the cache capacity tube always shows capacity usage by cache partitions, even on a tapeless or cloudless TS7700, as shown in Figure 7-1.



*Figure 7-1   MI cluster summary cache capacity*

# 7.2 Historical summary

Historical Summary is available under Monitor → Performance on the TS7700 MI, as shown in Figure 7-2. When you click the **Select Metrics** icon in the Historical Summary page, you can see the Select Metrics window. The *Primary Used* and *Data in Cache* view under the Storage section of Select Metrics include object data cache usage if DS8000 Object Store or Advanced Object Store is enabled.
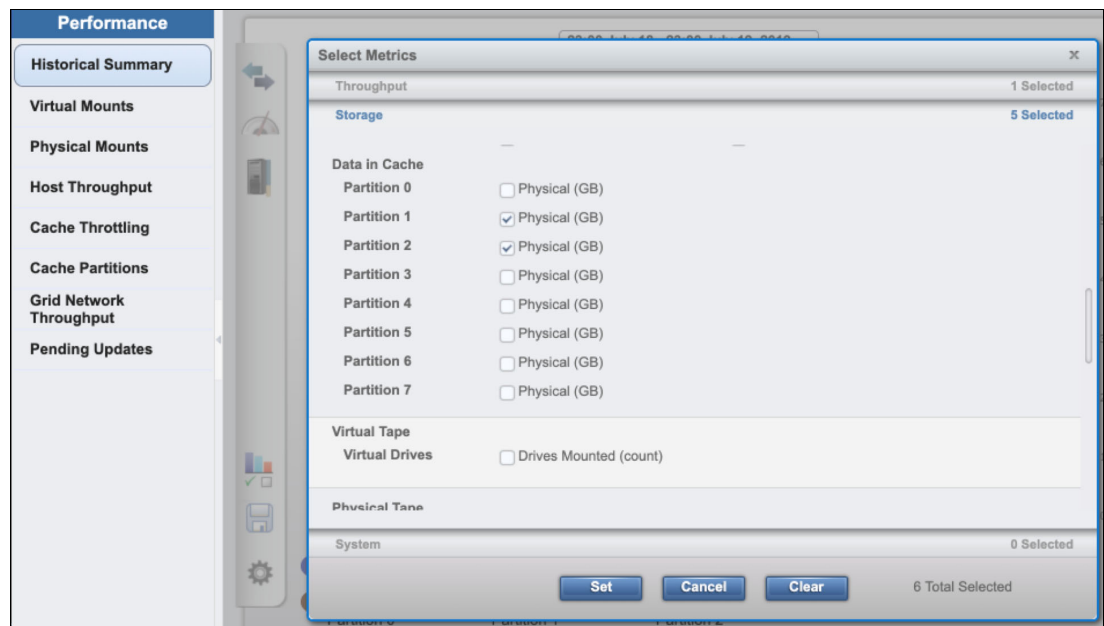


*Figure 7-2   MI Historical Summary - Storage window*

## 7.3  Cache utilization

This Cache Utilization window, as shown in Figure 7-3, is displayed on a tapeless and cloudless TS7700 only. It includes an option to show the total number of objects in cache.
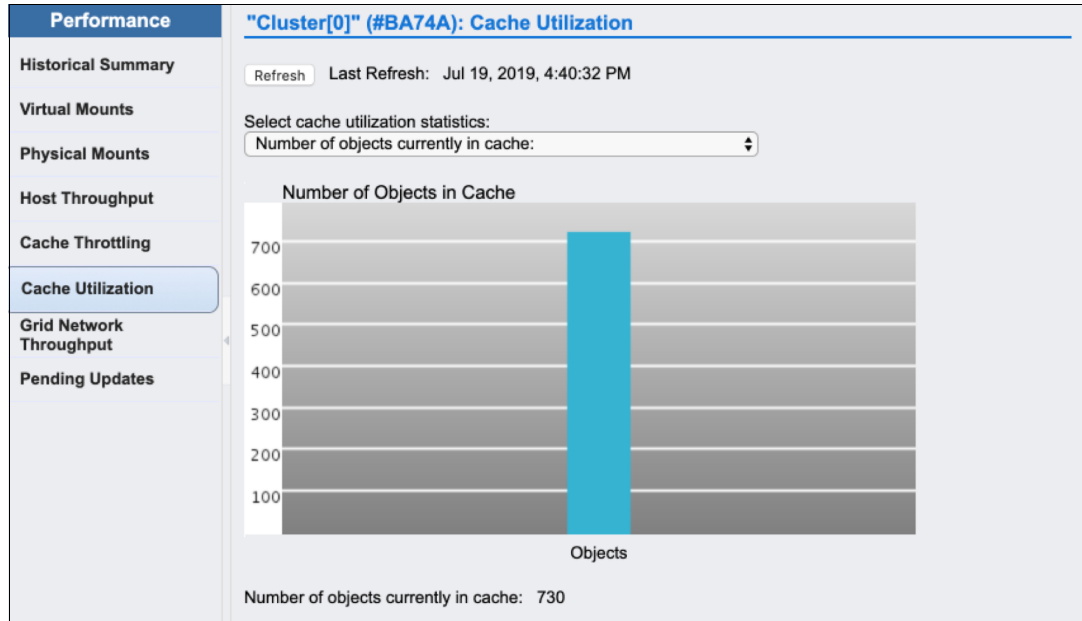


*Figure 7-3   MI cache utilization with object counts*

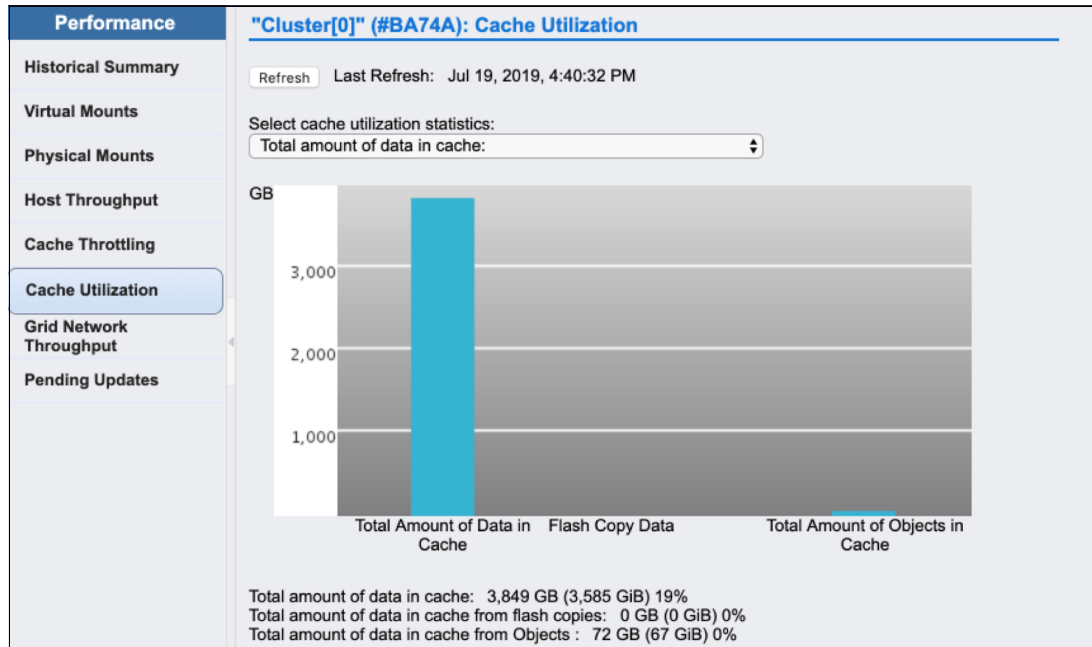The Cache Utilization window also includes the total amount of data in cache, as shown in Figure 7-4.



*Figure 7-4   MI cache utilization cache capacity*

## 7.4  Cache partition

The Cache Partition window, as shown in Figure 7-5, is under **Virtual** in the MI navigation menu. When DS8000 Object Store is enabled, one object partition is created. This object partition can be resized by using the **Resize** action. Because objects cannot be premigrated to tape or cloud, no option is available to choose a delayed premigration limit as there is for a logical volume partition.
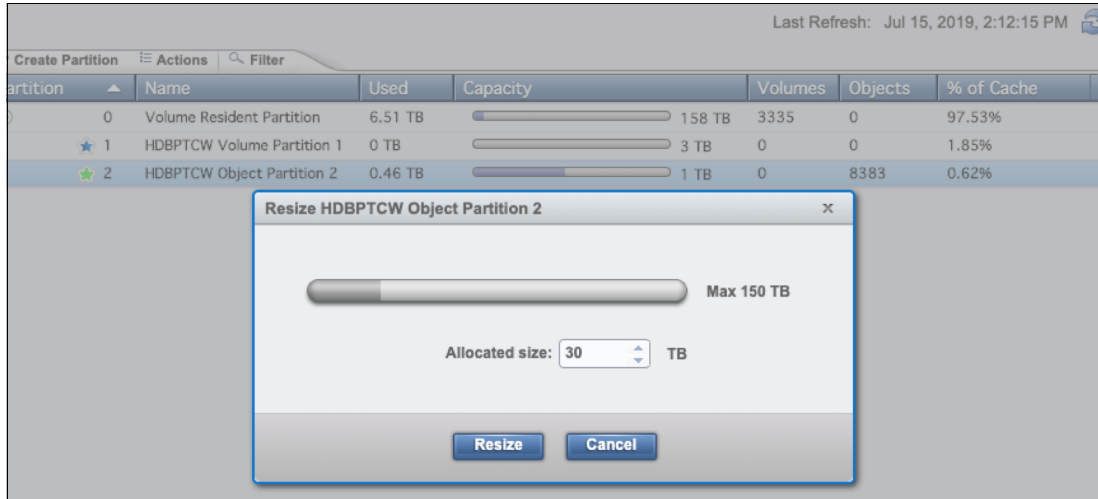


*Figure 7-5   MI object cache partition resize*

A column also displays the object count. This count is the total number of objects that are in an object cache partition. Figure 7-6 shows the Cache Partition window on a cloud attached TS7700 with DS8000 Object Store or Advanced Object Store enabled. Partition 2 is the object partition and contains 3351 objects. Partition 1 is a logical volume partition and contains no volume count.
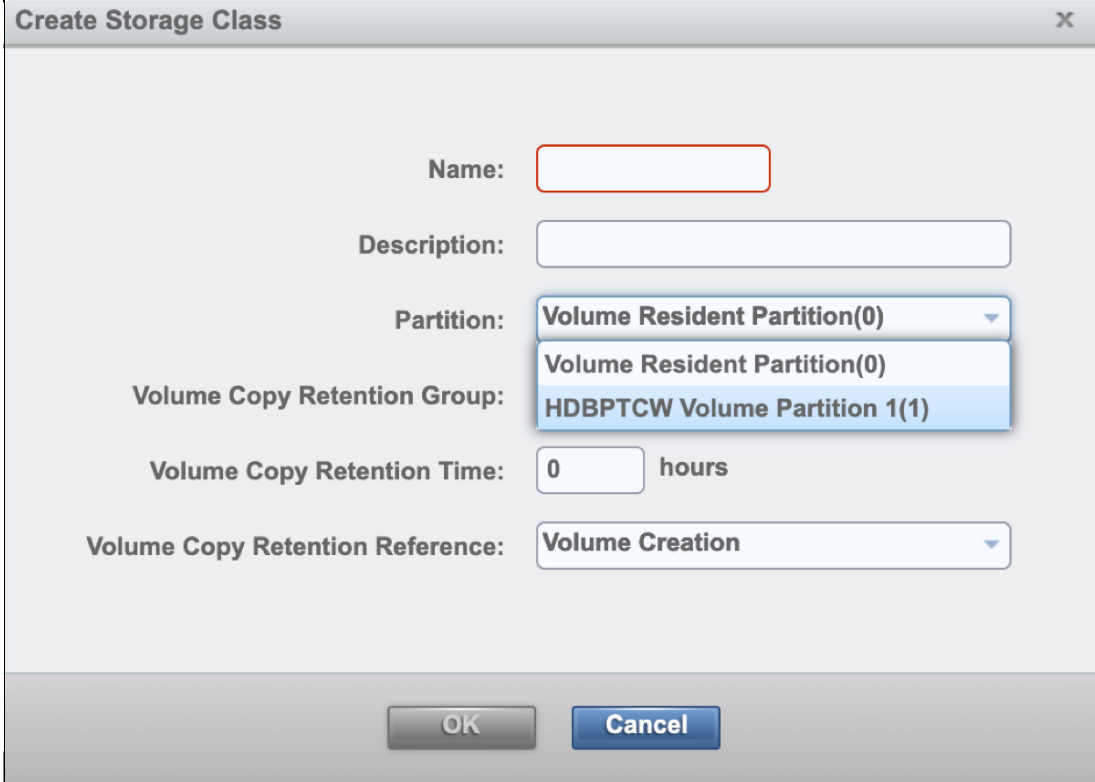


*Figure 7-6   MI cache partition with objects*

The panel also shows used capacity versus allocated capacity and is an aid to determining what percentage of available cache partition space is still available.

> **Note:** Resizing an object partition takes longer than resizing a logical volume partition. This difference is the result of internal setup requirements and is expected. Progress can be monitored in the Task window that is available under **Monitor → Task** on the MI.

## 7.5  Creating a storage class

Because object data cannot be migrated or premigrated to tape or cloud, Storage Class options are not available. When a Storage Class is created or modified from the Create Storage Class window, as shown in Figure 7-7, the object partition cannot be selected from the Partition drop-down menu. Only logical volume partitions are displayed.



*Figure 7-7   MI Create Storage Class window*

## 7.6  Backup and restore settings

Backup settings and restore settings can be found under **Settings** → **Cluster Settings**. These windows provide a way to back up settings that can be restored on another TS7700.

On a TS7700 that is tape-attached or cloud-attached, these settings include Cache Partitions. Object partitions feature special configuration properties that are cluster-specific and cannot be used on other TS7700s. For this reason, object partitions are not included in Backup Settings and are not part of a restore process.

Object Policy and Object Store definitions are also not included in backup and restore settings.

# 7.7  Object Policy

Beginning with microcode level 8.52.200.x, if Advanced **Object Store** is enabled, a navigation menu icon for Object Store is available as shown in Figure 7-8. MI pages that are listed under the **Object Store** menu icon help you to create, modify, and delete object policy and also help you to set up an object store on TS7700. Object policies are assigned to object stores.



*Figure 7-8   Object Store navigation icon*

The MI page Object Policy is available under the **Object Store** icon as shown in Figure 7-8. After you click **Object Store**, use the page to add, modify, and delete object policies. Object policies are used to set up copy policies for object redundancy across a TS7700 grid. Object policies are grid scope and pertain to how an object is managed for each cluster in the grid.

You can define object policies for every object-enabled cluster in the grid by accessing this page from one of those clusters. The maximum number of object policies that can exist on a TS7700 grid is 256. If a policy does not exist on TS7700 that is object-enabled, the default object policy is used. The default object policy can be modified but cannot be deleted. Figure 7-9 shows an example of the Object Policy MI page.
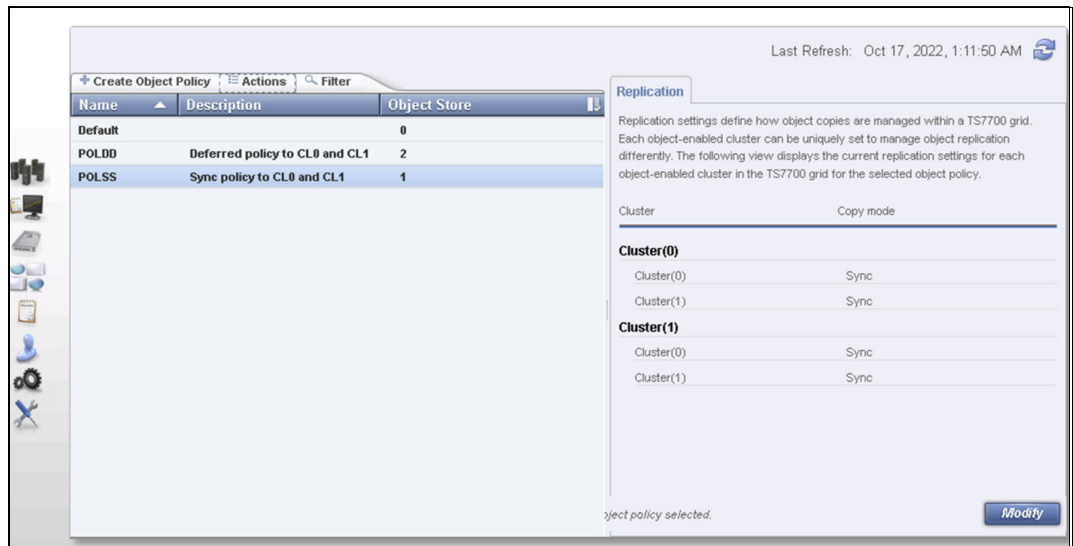


*Figure 7-9   MI Object Policy Window*

By clicking **Create Object Policy** in the Object Policy page, the Create Object Policy window opens as shown Figure 7-10. You can provide a name and a description of the policy in the Policy Information tab.
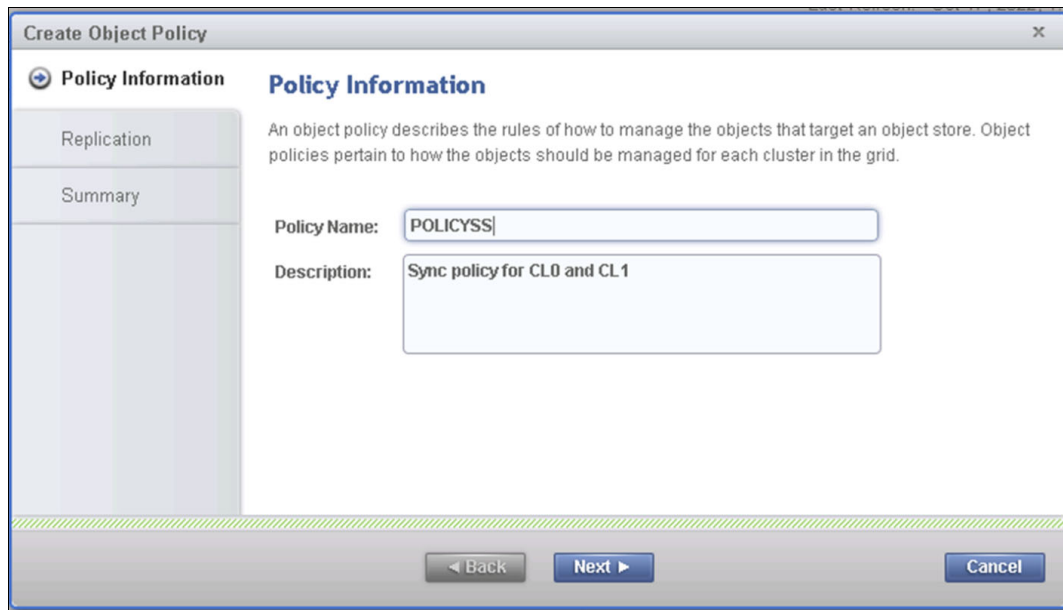


*Figure 7-10   MI Create Object Policy window - Policy Information tab*

Click **Next**, and you are redirected to the **Replication** tab as shown in Figure 7-11. You can choose the copy mode for each object-enabled cluster. You have the option to select between synchronous copy, deferred copy, or no copy.
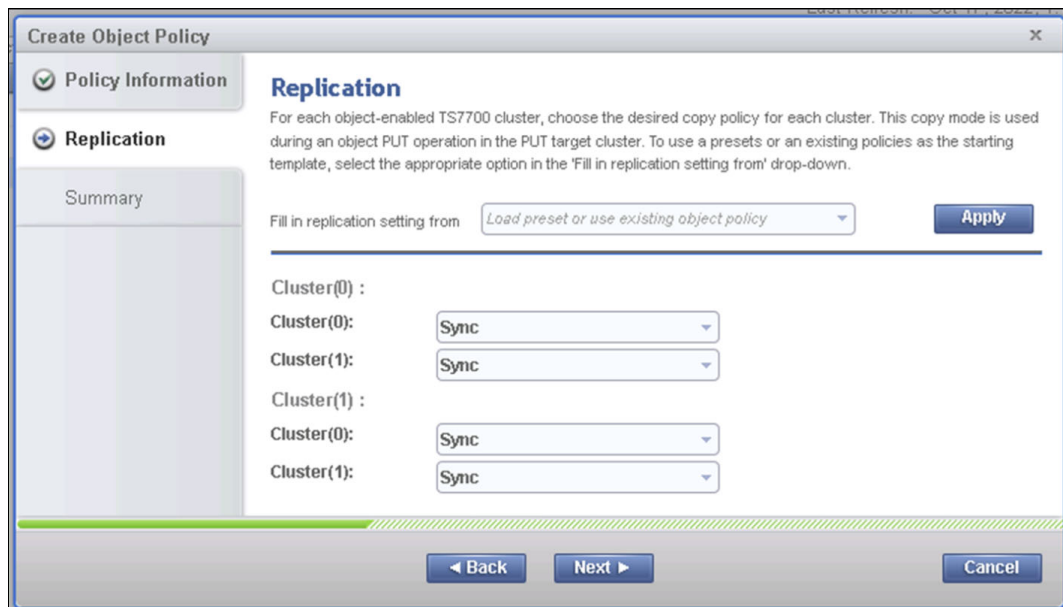


*Figure 7-11   MI Create Object Policy windows - Replication tab*

By clicking **Next**, the replication policy settings that are selected are displayed in the **Summary** tab as shown in Figure 7-12 on page 58.
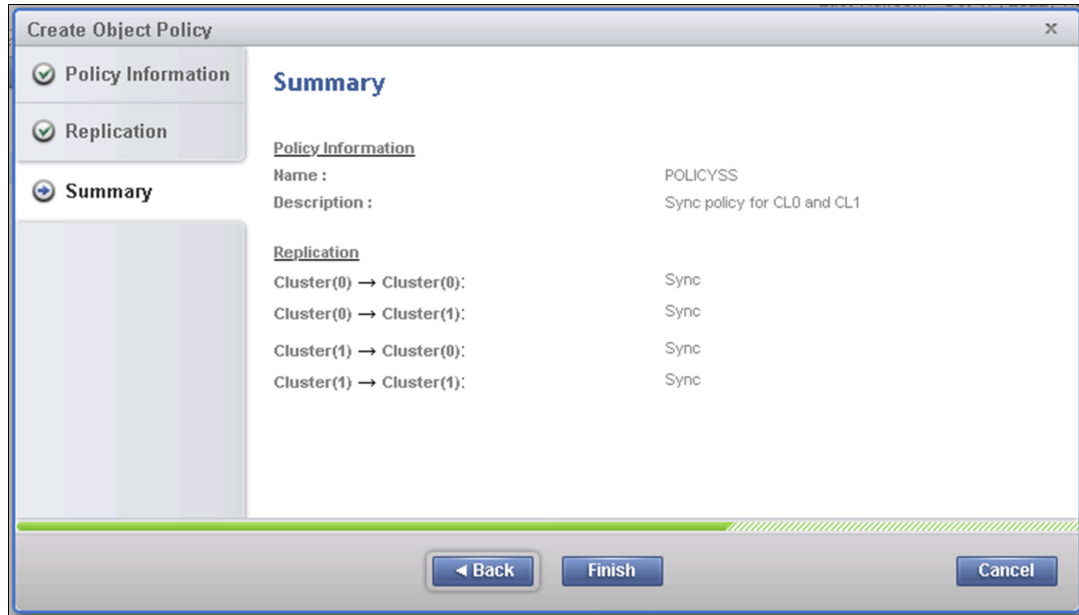
*Figure 7-12   MI Create Object Policy windows - Summary Tab*

## 7.8  Object Store

The MI page Object Store is also available under the navigational icon **Object Store** as shown in Figure 7-13 on page 59. Use the Object Store page to create, modify, and delete object store definitions.

To use DS8000 Transparent Cloud Tiering, you must create a z/OS DFSMS cloud name in the ISMF Cloud Network Connect Construct page. A Cloud Network Connection Construct name (cloud name) must be created on TS7700 before DS8000 configuration and zHost transactions are sent to the TS7700. The name that is specified on TS7700 must match the ISMF cloud name. The cloud name is case-sensitive. It is always displayed on the ISMF panel as uppercase but is sent to the TS7700 as lowercase. The TS7700 always stores and displays the name in lowercase. If the cloud name does not exist on TS7700, the DS8000 configuration process fails. If the cloud name is removed after configuration, host jobs that are using transparent cloud tiering fail.

The Object Store page is used to add cloud names to a TS7700 grid. These cloud names are used as object store targets for transparent cloud tiering. Each cloud name must be assigned an object policy. The Object Store page also lists the containers that are created by the z/OS host application. When you view the Object Store page, you can see any object stores that are created on any object-enabled cluster in the grid. Object stores are replicated on the other object-enabled clusters. A maximum of 256 object stores can be created for a TS7700 grid.

*Figure 7-13   MI Object Store window*

By clicking **Create Object Store** in the Object Store page, the Create Object Store window appears as shown Figure 7-14.



*Figure 7-14   MI Create Object Store window*

## 7.9 Dual Control

Dual Control provides management interface security by requiring two-person authorization when certain sensitive settings are changed in the TS7700 Management interface. For microcode level 8.52.200.x and later, the new Object Policy and Object Store pages are covered under Dual Control.

The Dual Control window, as shown in Figure 7-15, is under the **Access** icon in the MI navigation menu. If Dual Control is enabled, the following new settings require checker approval:

▶ Modify Object Policy
▶ Delete Object Policy
▶ Modify Object Store
▶ Delete Object Store



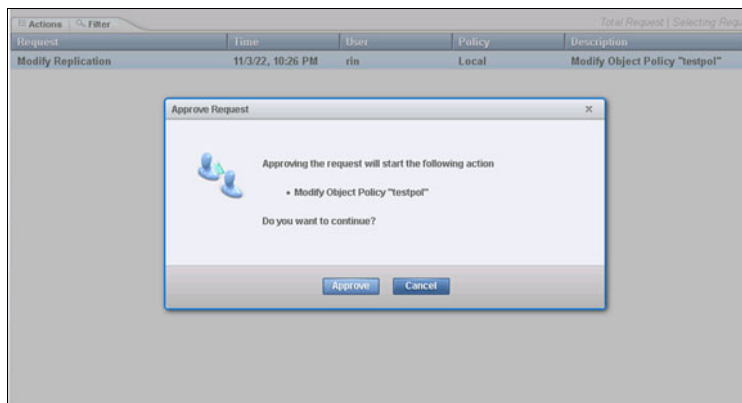*Figure 7-15   MI Dual Control window*

**8**

# Host commands

This chapter describes host command modifications and additions for the DS8000 Object Store feature or Advanced Object Storage.

This chapter includes the following sections:

# 8.1 RSDTLOW/RSDTHIGH

The **SETTING** command uses the following format:

```
LIBRARY REQUEST,<distributed-library>,SETTING,ALERT,[RSDTLOW/RSDTHIGH]
```

**RSDTLOW** and **RSDTHIGH** are thresholds in GBs at which the TS7700 generates a message that indicates that a logical volume cache partition exceeded a low or high residential capacity warning limit. These settings work as they do today, but are not used for object partitions when DS8000 Object Store is enabled.

# 8.2 RSDOLOW/RSDOHIGH

**RSDOLOW** and **RSDOHIGH** were introduced in R5.0 under keyword1 **SETTING2**. Beginning with R5.22 they were moved to the new keyword1 **OBJSET1.**

The **SETTING2** command uses the following format:

```
LIBRARY REQUEST,<distributed-library>,SETTING2,ALERT,[RSDOLOW/RSDOHIGH]
```

The **OBJSET1** command uses the following format:
```
LIBRARY REQUEST,<distributed-library>,OBJSET1,ALERT,[RSDOLOW/RSDOHIGH]
```

**RSDOLOW** and **RSDOHIGH** work similarly to **RSDTLOW** and **RSDTHIGH**, but are used for object partitions. If DS8000 Object Store or Advanced Object Store is enabled, these commands can be used to generate a warning message.

Table 8-1 lists the keywords that are used for this command and whether the command can be run for the composite library, distributed library, or both.

*Table 8-1   RSDOLOW/RSDOHIGH keywords*

| Keyword1 | Keyword2 | Keyword3 | Keyword 4 | Description | Comp | Dist |
|---|---|---|---|---|---|---|
| *Setting2 or Objset1* | **ALERT** | **RSDOHIGH** | **<value>** | The threshold in GBs of resident data for all object cache partitions that, when reached, triggers a message from the TS7700 indicating that the amount of resident data has exceeded a high warning limit. | N | Y |
| *Setting2 or Objset1* | **ALERT** | **RSDOLOW** | **<value>** | The threshold, in GBs of resident data for all object cache partitions that, when reached, triggers a message from the TS7700 indicating that the amount of resident data has exceeded a low warning limit. | N | Y |

Example 8-1 shows the `ALERTS` section of the **SETTING2** command after **RSDOLOW** and **RSDOHIGH** is set.

*Example 8-1   SETTING2 RSDLOW/RSDOHIGH values*

```
ALERTS
PDRVSLOW =            0 PDRVSCRT =           0
LMTDTHR  =            0
CAGALOW  =            0 CAGAHIGH =           0
CAGLOW   =            0 CAGHIGH  =           0
RSDOLOW  =          500 RSDOHIGH =         700
```

Example 8-2 shows the **LI REQ,<distributed-library>,OBJSET1** response after **RSDOLOW** and **RSDOHIGH** are set. OCPYHIGH and OCPYLOW shown in the example are future alerts and will display with the **OBJSET1** output but cannot be set now.

*Example 8-2   OBJSET1 RSDLOW/RSDOHIGH values*

```
OBJSET1 V1 .0
 ALERTS
 OCPYHIGH =           0  OCPYLOW =           0
 RSDOHIGH =        2000  RSDOLOW =        1500
```

# 8.3  CACHE2

The **CACHE2** command uses the following format:

```
LIBRARY REQUEST,<distributed-library>,CACHE2
```

The **LI REQ CACHE2** command replaces the **LI REQ CACHE** command. Table 8-2 lists the features of the **LI REQ CACHE2** command.

*Table 8-2   LI REQ CACHE2 command description*

| Keyword1 | Keyword2 | Keyword3 | Keyword 4 | Description | Comp | Dist |
|----------|----------|----------|-----------|-------------|------|------|
| **CACHE2** | | | | Requests information about the current state of the cache and the data that is managed within it associated with the specified distributed library. | N | Y |

The most significant change to the command is a new TCO column, which maps to the following components:

► T for Tape migration
► C for Cloud migration
► O for partition contains Objects

A **Y** under the T, C, or O indicates that the partition supports this option. An **N** indicates that the partition does not support this item. If the O column contains an **N**, the partition is a logical volume partition.

Example 8-3 on page 64 shows the output of the **CACHE2** command. In this output, Partition 1 is a logical volume partition that supports migration of data to tape and Partition 2 is an object partition.

*Example 8-3   LI REQ,CACHE2 command*

```
SHOWING RESULTS FOR COMMANDS: CACHE2
TAPE VOLUME CACHE STATE V5 .0
TOTAL SPACE INSTALLED/ENABLED:  162TB/ 162TB
TOTAL ADJUSTED CACHE SPACE USED: 6509GB
CACHE ENCRYPTION STATUS: NOT CAPABLE
OVERCOMMITTED CACHE PARTITIONS: NONE
CACHE RESIDENT ONLY PARTITION
 PRIVATE CACHE SPACE USED :  243GB
 SCRATCH CACHE SPACE USED: 6265GB
CP   ALLOC    USED      PIN     PKP    PRM    COPY    CPYT
 0   158TB  6508GB  2088GB  4419GB  0.0GB  0.0GB       0
 FLASH COPY INFORMATION
 INDEX ENABLED    SIZE
     1     NO   0.0GB
     2     NO   0.0GB
     3     NO   0.0GB
     4     NO   0.0GB
     5     NO   0.0GB
     6     NO   0.0GB
     7     NO   0.0GB
     8     NO   0.0GB
TIERED CACHE PARTITIONS
 CP TCO ALLOC    USED    PG0     PG1  PMIGR D_PMIGR   COPY  PMT CPYT
  1 YNN 3000GB  0.0GB  0.0GB   0.0GB  0.0GB   0.0GB  0.0GB    0    0
  2 NNY 1000GB  462GB  0.0GB   462GB  0.0GB   0.0GB  0.0GB    0    0
  3 NNN  0.0GB  0.0GB  0.0GB   0.0GB  0.0GB   0.0GB  0.0GB    0    0
  4 NNN  0.0GB  0.0GB  0.0GB   0.0GB  0.0GB   0.0GB  0.0GB    0    0
  5 NNN  0.0GB  0.0GB  0.0GB   0.0GB  0.0GB   0.0GB  0.0GB    0    0
  6 NNN  0.0GB  0.0GB  0.0GB   0.0GB  0.0GB   0.0GB  0.0GB    0    0
  7 NNN  0.0GB  0.0GB  0.0GB   0.0GB  0.0GB   0.0GB  0.0GB    0    0
EXECUTING COMMANDS: CACHE2
```

# 8.4  GRLNKACT

The **GRLNKACT** command uses the following format:

```
LIBRARY REQUEST,<distributed-library>,STATUS,GRLNKACT
```

The **GRLNKACT** command was first introduced in microcode release 8.42.x.x to display point-in-time Grid link activity for all clusters in the Grid but at the perspective of the cluster that is specified in the distributed library ID field, as shown in Table 8-3.

*Table 8-3   GRLNKACT command description*

| Keyword1 | Keyword2 | Keyword3 | Keyword 4 | Description | Comp | Dist |
|----------|----------|----------|-----------|-------------|------|------|
| **STATUS** | **GRLNKACT** | | | Requests information about Grid link point-in-time activity. | N | Y |

In TS7700 microcode level 8.50.x.x, fields were added to support DS8000 Object Store and Grid to Grid Migration (GGM) activity.

Example 8-4 on page 65 is a sample output of the **LI REQ,STATUS,GRLNKACT** command.

*Example 8-4   LI REQ,STATUS,GRLNKACT command output*

```
GRLNKACT STATUS V1 .0
CLUSTER  INDEX: 2 LINK COUNT:  4 Time: Thu Apr 11 00:05:13 CUT 2019
GRID LINK ESTABLISHED SOCKET CONNECTIONS-----------------------------
LN INTF IP             C0  C1 C2 C3 C4 C5 C6 C7 MQ  RFA  CLD
L0 en10 9.11.219.178   106 82 0  0  0  0  0  87 268 7    1
L1 en4  9.11.219.190   5   2  0  0  0  0  0  2  3   6    2
L2 en11 9.11.219.191   3   4  0  0  0  0  0  0  0   7    1
L3 en5  9.11.219.192   3   4  0  0  0  0  0  0  0   7    4
LN INTF IP             GGM  OBJ
L0 en10 9.11.219.178   8    6
L1 en4  9.11.219.190   8    6
L2 en11 9.11.219.191   0    0
L3 en5 9.11.219.192    0    0
NET ACTIVITY -----TCP RECV/SEND ADAPTER BUFFER ACTIVITY BYTES---------
LN   TxMBs RxMBs MQ_REC  MQ_SND   GFA_REC  GFA_SND  CLD_REC  CLD_SND
L0   81    225   0       0        0        124792   0        0
L1   58    156   0       0        0        0        0        2089817
L2   2     0     0       0        0        0        0        0
L3   42    0     0       0        0        0        0        1046017
TOT  183   381   -       -        -        -        -        -
LN   GGM_REC  GGM_SND  OBJ_REC   OBJ_SND
L0   0        0        2099835   4179634
L1   1460     0        1050791   1039795
L2   0        0        0         0
L3   0        0        0         0
GRID LINK THROUGHPUT ESTIMATES-MB/s------------------------------------
DIR C0  C1  C3  C4  C5  C6  C7  GRD_TOT  CLD  GGM  OBJ  TOT
Tx  2   3   1   0   0   0   1   7        78   0    98   183
Rx  0   0   0   0   0   0   0   0        247  1    133  381
GRID CLOUD TIER EXPORT AND IMPORT ACTIVITY---------------------------
ACTIVE EXPORT VOLUME COUNT: 5
ZCL060 ZCL061 ZCL062 ZCL063 ZCL065
ACTIVE IMPORT VOLUME COUNT: 0
```

The fields that were added in microcode release 8.50.x.x in support of the DS8000 Object Store feature are listed in Table 8-4.

*Table 8-4   DS8000 Object Store feature descriptions*

| Grid link established socket connections | |
|---|---|
| OBJ | The number of TCP Sockets that is established between the TS7700 and all DS8000s for each link (L0-L3). |
| **Net activity** | |
| OBJ_REC | Amount of data, in bytes, that was sent for write activity by all DS8000s to this cluster as part of a transparent cloud tiering **PUT** command in the last 15 seconds. This value is per link. |
| OBJ_SND | Amount of data, in bytes, that was read by all DS8000s from this cluster as part of a transparent cloud tiering **GET** command in the last 15 seconds. This value is per link. |
| TxMBs TOT | The total transmitted activity of all links in the last 15 seconds. This total now includes OBJ_SND. |

| | |
|---|---|
| RxMBs TOT | The total received activity of all links in the last 15 seconds. This total now includes OBJ_REC. |
| **Grid link throughput estimates - MBps** | |
| OBJ Tx | Estimated Grid link throughput for object data that is sent from this cluster to all DS8000s in the last 15 seconds. |
| OBJ Rx | Estimated Grid link throughput for object data that is received by this cluster to and from all DS8000s in the last 15 seconds. |
| TOT Tx/Rx | The total estimated transmit and receive grid link throughput activity. These totals now include OBJ throughput. |

**Note:** The cluster number for which this data was read is no longer listed in the Grid Link Throughput Estimates-MBps section of this output to make room for the new OBJ field. The local cluster number is always zero and has no value. For example, in Example 8-4 on page 65, column C2 is not included because this data was generated for the distributed library ID that belongs to cluster 2.

## 8.5  OCOPY, SUMMARY

The **OCOPY** command uses the following format:

LIBRARY REQUEST,<d*istributed-library>*,OCOPY,SUMMARY

The **OCOPY** keyword is available in microcode release 8.52.200.x (R5.22) and later to support Advanced Object Store. The **OCOPY** keyword is only supported on clusters that enable the Advanced Object Store. The **OCOPY** command requests a cluster to collect information about copies for objects. Copies for tape logical volumes are not included in this response. For information about tape logical volumes copies, see the **LI REQ, COPY** command. Currently, only the **SUMMARY** second keyword is supported and returns a summary of all object copies associated with the distributed library, as shown in Table 8-5.

*Table 8-5   OCOPY supported keywords*

| Keyword1 | Keyword2 | Keyword3 | Keyword4 | Description | Comp | Dist |
|---|---|---|---|---|---|---|
| OCOPY | SUMMARY | | | Requests summary information about all object copy jobs. | N | Y |

Example 8-5 is a sample output of the **LI REQ,OCOPY,SUMMARY** command:

*Example 8-5*  **LI REQ,OCOPY,SUMMARY** command output

```
OCOPY SUMMARY V1 .0
 DISTRIBUTED LIBRARY cluster1 CLUSTER 1
----------------------------------------------------------------------
COPY COUNT BY ORIGINATING CLUSTER BY OBJECT COPY QUEUE
CLUSTER      0      1      2      3      4      5      6      7      *
----------------------------------------------------------------------
SyncDef      0     NA     NC     NC     NC     NC     NC     NC      0
    Def      2     NA     NC     NC     NC     NC     NC     NC      0
CpyRfsh      0     NA     NC     NC     NC     NC     NC     NC      0
----------------------------------------------------------------------
```

```
COPY SIZE GB BY ORIGINATING CLUSTER BY COPY QUEUE
CLUSTER       0       1       2       3       4       5       6       7       *
-----------------------------------------------------------------------
SyncDef       0      NA      NC      NC      NC      NC      NC      NC       0
    Def   4295M      NA      NC      NC      NC      NC      NC      NC       0
CpyRfsh       0      NA      NC      NC      NC      NC      NC      NC       0
-----------------------------------------------------------------------
LONGEST OBJECT COPY QUEUE AGE :          33
```

The first and second matrices show the total copy count and size grouped by the object copy queue (per each line) and originating cluster (per each column) on the distributed library in the request.

Each line shows the object copy queue type:

**SyncDef**        A copy job with Synchronous copy mode. A copy job is posted when any event occurs which prevents the synchronous updates to one of the two clusters assigned to synchronous copy mode.

**Def**            A copy job with Deferred copy mode

**CpyRfsh**        A copy job posted by "COPYRFSH" from a migration event

For more information about these Library Request commands, see: TS7700 Library Request Command.

**9**

# Statistical data

Beginning with microcode release 8.52.200.x (R5.22), Bulk Volume Information Retrieval (BVIR) introduced three new request commands that provide information for the Advanced Object Store feature. The commands are `OBJECT STATUS`, `COPY AUDIT` and `OBJECT COPY QUEUE`.

The existing `HISTORICAL STATISTICS` command was also extended to include statistical information about Advance Object Store transactions. `VEHSTATS` statistical reports, which are generated from response files of the command, added new reports for Advanced object store. This chapter provides general descriptions of these BVIR commands and VEHSTATS reports.

This chapter includes the following sections:

- ► 9.1, "BVIR OBJECT STATUS" on page 70
- ► 9.2, "BVIR COPY AUDIT" on page 71
- ► 9.3, "BVIR OBJECT COPY QUEUE" on page 72
- ► 9.4, "VEHSTATS statistical reports" on page 73

For more information about BVIR, see TS7700 Series Bulk Volume Information Retrieval Function User's Guide Version 5.4.

For more information about VEHSTATS statistical reports, see: VEHSTATS Decoder Version 2.5 and TS7700 Series Statistical Data Format Version 5.3a.

## 9.1  BVIR OBJECT STATUS

The OBJECT STATUS request provides status information about all objects under a cloud name, a cloudname and container combination, or a set of objects by cloudname, container, and object prefix.

The request format of **OBJECT STATUS** is nontraditional and it can be more than one line, as shown in Example 9-1.

*Example 9-1   OBJECT STATUS request*

```
VTS BULK VOLUME DATA REQUEST
OBJECT STATUS MyCloud#@$00000000000000000001 87600
OBJECT ID 5E4145C500BC614EVVSSSSSSSSSSSLLLLCCCC
CONTAINER ContainerName_256-BYTES#...00000000000000000000000000000000000000
CONTAINER 0000000000000000000000000000000000000000000000000000000000000000000
CONTAINER 0000000000000000000000000000000000000000000000000000000000000000000
CONTAINER 00000000000000000000000000000000000000000001
OBJPREFIX Object_PreFiX_#128-BYTES/00000000000000000000000000000000000000000
OBJPREFIX 00000000000000000000...00000000000000000000000000000000
```

Each line includes request commands:

► Line 1: **VTS BULK VOLUME DATA REQUEST**

► Line 2: **OBJECT STATUS** *<cloudname> <hours>*

 – *<cloudname>* - Needs to match the DFSMS Cloud Network Connection name

 – *<hours>* - value from 0-87600. 0 means all records. Any other value returns the objects that were created within this number of hours based on the object creation time.

► Line 3: **OBJECT ID** *<object id>*

 – If the value is 0, the response contains up to the first 4 million records. Any subsequent request requires the OBJECT ID of the last record from the previous response and will send the next 4 million records starting with that OBJECT ID.

► Line 4-N:

 – Up to 4 lines: **CONTAINER** *<container name>*

 • Optional input that can narrow the response by the container by sending the container name.

 • Container names can be up to 256 characters so must be split into multiple lines due to an 80-character limit per line.

 – Up to 2 lines: **OBJPREFIX** *<object prefix>*

 • Optional input that can narrow the response by object prefix. This can be used to return all objects for a data set or dump. A container name is required to use this input.

The output of the **OBJECT STATUS** request is csv format as shown in Example 9-2. The record 6 of the output serves as a column header for the proceeding rows of the response record, so that it is displayed to the user.

*Example 9-2   OBJECT STATUS output (partial)*

```
VTS BULK VOLUME DATA REQUEST
OBJECT STATUS
11/10/2022 08:11:48 VERSION 01
S/N: HDCB0  LIB ID: D0030

CLOUD_NAME,CONTAINER_NAME,OBJECT_ID,OBJECT_NAME,DS8K_SERIAL_CREATE,DS8K_SERIAL_DE
vtscloud0,SYSZADR.DSSDUMP1,6348E53C225240000000000HDCB000000008,RINTEST/D221014A/
vtscloud0,SYSZADR.DSSDUMP1,6348E576551C80000000000HDCB000000009,RINTEST/D221014A/
vtscloud0,SYSZADR.DSSDUMP1,6348E5766A2C00000000000HDCB00000000A,RINTEST/D221014A/
vtscloud0,SYSZADR.DSSDUMP1,6348E57679B600000000000HDCB00000000B,RINTEST/D221014A/O
BJPREF
```

## 9.2  BVIR COPY AUDIT

The **OBJECT COPY AUDIT** request performs an audit of the databases on a set of specified TS7700 distributed libraries to determine whether there are any objects that do not have a valid copy on at least one of them the clusters in the grid.

The request format of **OBJECT STATUS** can also be more than one line, as shown in Example 9-3.

*Example 9-3   OBJECT AUDIT request*

```
VTS BULK VOLUME DATA REQUEST
OBJECT COPY AUDIT INCLUDE BA38E
DATASET 0
COPYMODE NO
```

Each line includes request commands:

▶ Line 1: **VTS BULK VOLUME DATA REQUES**T

▶ Line 2: **OBJECT COPY AUDIT INCLUDE|EXCLUDE** *<lib-id0>,…,<lib-id7>*

   – List of clusters, by distributed library ID, to either be included or excluded as part of the audit.

   – At least one libid is required.

   – Each *lib-id* is comma separated if more than one is specified.

- Line 3: `DATASET 0 | ` *`NEXT_INDEX_KEY`*
  - If 0 is passed in, TS7700 provides the first set of audit output up to 4 million.
  - To get the next 4 million in the series, pass in the *`NEXT_INDEX_KEY`* from the previous response.
  - When *`NEXT_INDEX_KEY`* contains "NO MORE RECORDS", the current response contains the last of the available audit data.
- Line 4: `COPYMODE YES|NO`
  - Determines whether the audit takes into consideration the copy modes.
  - If `YES`, TS7700 checks whether the libid requires a copy and considers it only if it does require a copy.
  - If `NO`, TS7700 performs the audit without considering copy modes.

The output of the `OBJECT COPY AUDIT` is shown in Example 9-4. In this example, the audit does not find any objects that are not valid on the specified TS7700 cluster.

*Example 9-4   OBJECT COPY AUDIT output*

```
BVIR OBJECT COPY AUDIT RESPONSE
VTS BULK VOLUME DATA REQUEST
OBJECT COPY AUDIT
07/16/2021 19:38:36 VERSION 01
S/N: 35C10 LIB ID: BA27A
NEXT_DATASET: TS7700BJECTID00112233445566778899001+2021-07-16-06.46.27.984190
DB2BACKUPS,SysZAdm2021020,OBJ:EXTENTS,OBJID00004, DDDDNNNN, NNNNNNNN, N
DB2BACKUPS,SysZAdm2021020,OBJ:EXTENTS,OBJID00022, DDNNNNNN, NNNNNNNN, N
```

# 9.3  BVIR OBJECT COPY QUEUE

The `OBJECT COPY QUEUE` request produces the list of objects within the cluster that need to have a copy.

The request format of `OBJECT COPY QUUE` is the traditional format as shown in Example 9-5.

*Example 9-5   OBJECT COPY QUEUE request*

```
VTS BULK VOLUME DATA REQUEST
OBJECT COPY QUEUE
```

The output of the **OBJECT COPY QUEUE** request is csv format as shown in Example 9-6. The record 6 of the output serves as a column header for the proceeding rows of the response record, so that it is displayed to the user.

*Example 9-6   OBJECT COPY QUEUE output (partial)*

```
VTS BULK VOLUME DATA REQUEST
OBJECT COPY QUEUE
11/10/2022 09:03:04 VERSION 01
S/N: HDCB0  LIB ID: D0030

CLOUD_NAME,CONTAINER_NAME,OBJECT_ID,OBJECT_NAME,CONSISTENCY_ID,QUEUE,TASK,
vtscloud0,SYSZADR.DSSDUMP2,636477F2644010000000000HDCB000000037,RINTEST/D2
vtscloud0,SYSZADR.DSSDUMP2,636477F2312DF0000000000HDCB000000035,RINTEST/D2
vtscloud0,SYSZADR.DSSDUMP2,636477F1B98110000000000HDCB000000034,RINTEST/D2
vtscloud0,SYSZADR.DSSDUMP2,636477F1735900000000000HDCB000000032,RINTEST/D2
```

# 9.4  VEHSTATS statistical reports

The output of **VEHSTATS** provides four types of statistical reports for Advanced Object Store:

► **H360BJSG** - hnode Object Store General Historical Record
► **H37CLOSN** - hnode Object Store Activity by Clusters
► **H37OSNCL** - hnode Object Store Activity by Store Names
► **H380SNPT** - hnode Object Store by Name and Partition Historical Record

The **H360BJSG** report provides overall information concerning the configuration of the Object Store as shown in Figure 9-1.



*Figure 9-1   VEHSTATS H36OBJSG report*

H37CLOSN and H37OSNCL provide overall information including transactions between object clients, data transfers among clusters, and object copy queues. H37CLOSN is sorted by clusters as shown in Figure 9-2.



*Figure 9-2   VEHSTATS H37CLOSN report*

H37OSNCL presents the same information as H37CLOSN but is sorted by Store Names as shown in Figure 9-3.



*Figure 9-3   VEHSTATS H37OSNCL report*

**H380SNPT** provides statistics that are grouped by Object Store names and Partition as shown in Figure 9-4.



*Figure 9-4   VEHSTATS H38OSNPT report*

# Migration services

This chapter provides information about how to migrate DS8000 Transparent Cloud Tiering object data within the TS7700 or how to move from FC 5282 to the new FC 5283 on an existing system.

This chapter includes the following sections:

## 10.1  Removing FC 5282

In the instance where FC 5282 is installed on one or more clusters in the grid or on a stand-alone system but where no object data or containers exist on any cluster, `VTD_EXEC.385` can be run to remove FC 5282 before installing FC 5283. This process is available with approval and is run by the IBM Remote Code Load (RCL) team. This is an offline procedure:

1. The RCL team brings the TS7700 cluster offline.

2. The RCL team uses the Service Menus to run `VTD_EXEC.385`.

3. The RCL team brings the TS7700 cluster online.

4. After `VTD_EXEC.385` completes successfully, FC 5282 should no longer exist and the new FC 5283 can be installed.

This procedure provides an easy way to remove FC 5282 if the feature was not yet used or if a small amount of object data was written to the TS7700 and can be easily recalled. All objects and containers must be removed from the TS7700 before running the `VTD_EXEC,` otherwise, the `VTD_EXEC` fails.

Object containers are created by the host and require host commands to delete them. Each host application might have a different process to delete objects and containers. For instance, the DFSMShsm data set migrations process deletes containers after all objects have been deleted by using `HDELETE`. This is an asynchronous process, and users should contact application experts to determine how often the process runs. DFSMSdss provides the `CLOUDUTILS` utility to delete objects and containers.

If you plan to remove FC 5282 to replace with FC 5283, all FC 5282 must be removed from the entire TS7700 grid before installing FC 5283 anywhere into the grid. The two features are different, and using both methods in the same grid is not supported.

`VTD_EXEC.385` to remove FC 5282 (option 'r') is supported on TS7700 microcode levels R5.0 and later.

## 10.2  FC 5282 to FC 5283 migration

In the instance where FC 5282 is enabled on one or more clusters and object data exists object metadata must be collected and stored in the TS7700 to allow for FC 5283 compatibility. The TS7700 must be on microcode level 8.53.x or later to use this process.

This process is available as a concurrent activity on the TS7700. Meaning, the TS7700 cluster can remain online and accept DS8000 operations during the migration. However, during one step of the process, all DS8000 systems that are connected to the TS7700 grid that is going through this migration process must be unconfigured from the TS7700 for a short period. This will allow the transition from the older feature to the new feature.

After the process is started on the TS7700, the DS8000 can be reconfigured, and transparent cloud tiering object operations can be resumed. This process is run by IBM Expert Lab (formerly known as IBM Lab Services) who start and monitor the migration process.

This migration process does not move any data. The migration process populates new TS7700 object database tables, which were established during the FC 5283 installation process. The TS7700 file system is scanned, and the data is inserted into the object tables on every FC 5283 enabled cluster in the grid.

The following list provides an overview to perform this migration service:

1. IBM Expert Lab runs `VTD_EXEC.387` to determine the state of the object stores within the TS7700. If there are cloud names or containers with uppercase letters, convert all to lowercase. FC 5283 stores this metadata and directories as lowercase.

2. IBM Expert Lab installs FC 5283 on each cluster that has FC 5282 enabled. The following rules apply:

   – Only clusters with FC 5282 can have FC 5283 installed. A cluster that contains both feature codes indicates it is ready for the migration process.

   – FC 5283 cannot be installed on clusters that do not container FC 5282 if at least one other cluster in the grid has FC 5282. FC 5282 must either be removed, or the cluster must fully complete this migration process before introducing FC 5283 to other clusters in the grid.

   – Do not start the migration process until all FC 5282 clusters also have FC 5283 installed. This is true even if no objects were ever written to that cluster. Optionally, those clusters that have no objects or containers can go through the FC 5282 removal. See 10.1, "Removing FC 5282" on page 78.

   > **Note:** The DS8000 and TS7700 continue to operate by using the FC 5282 method until the actual migration process starts in step 6.

3. All clusters must be online when the migration process is started.

4. IBM Expert Lab completes any preparation steps for the migration process.

5. When the migration process is ready to begin, IBM Expert Lab has the customer unconfigure all DS8000 Transparent Cloud Tiering processes going to the TS7700 grid by using `rmcloudserver` on the DSCLI.

6. IBM Expert Lab runs `VTD_EXEC.385` with the migration option (option 'm'). `VTD_EXEC.385` does some validation steps, then updates the new object database tables with information about the cloud name file sets and container directories that exist on all FC 5282 enabled clusters in the grid. This step should take only a few minutes to complete.

7. After the previous step completes, the customer can reconfigure all DS8000 Transparent Cloud Tiering systems by using the `mkcloudserver` command. From this point forward, DS8000 Transparent Cloud Tiering uses the new FC 5283 functions when host transactions go to the TS7700.

8. `VTD_EXEC.385` will immediately move on to the next step with populating the object database tables. When the entire metadata migration process is complete, IBM Expert Lab sees a *VTD_EXEC.385 successful* message. The amount of time to complete the migration process varies based on the number of objects that are migrating. A reasonable estimate is 2 hours for every 100,000 objects.

9. When `VTD_EXEC.385` completes, it automatically removes FC 5282 from all clusters in the grid, and only FC 5283 is visible.

## 10.3  Object migration services

Object migration services provide the ability to migrate object data to other clusters in the grid. This process is similar to the existing tape logical volume migration known as Copy Refresh.

Copy Refresh for objects allows existing data to be copied to other clusters in the grid without requiring the host to trigger `GET` or `PUT` operations. This object migration service is run by the IBM Expert Lab team and is supported only when FC 5283 is installed.

Migration services are beneficial when there is a need to change the copy settings of existing objects within a grid for the following scenarios:

► Hardware refresh
► Data copy requirements changing, such as more copies required
► Data center moves

The service supports copying objects to other locations on the grid and assigning a new Object Policy to objects.

When using this service, objects can be targeted by using different options:

► All objects.

► All objects within a specific cloud name.

► All objects within a container or set of containers by prefix, for example, *Container\** for Container1–99.

► Objects by name or a set of objects by prefix.

► Assign any combination of the methods in this list to different policies (copy settings).

► Contact the IBM Expert Lab team to assist with your Object Migration needs.

# 11

# Events and notifications

This chapter provides general information about TS7700 events, host attention messages, and call home events that are related to Advanced Object Store. It also introduces the DS8000 event that notifies the link failure between TS7700 and DS8000.

This chapter includes the following sections:

# 11.1 TS7700 notifications

There are some expected TS7700 events that are generated related to Advanced Object Store:

- ► When the TS7700 cannot reconcile an object token and when this condition clears.
- ► When a sync-deferred event (one sync copy failed) occurs and when leaving a sync-deferred state.
- ► When TS700 cannot delete object containers and when this condition clears.
- ► When the TS7700 cannot delete objects and when this condition clears.

These events show up under the *Events notification* page in the TS7700 MI. z/OS is notified through CBR3750I messages. For more information, see IBM TS7700 Series Notification Messages White paper Version 5.3.

If sync mode copy is used for objects and sync-deferred occurs, TS7700 sends attention messages to z/OS under the following conditions:

- ► When a cluster enters the first sync-deferred copy queue entry (CBR3730E)
- ► When the last sync-deferred copy is processed, another attention message is sent indicating the condition is cleared (CBR3731)

The TS7700 sends Call Home events when it cannot delete objects after it exceeds the retry time, value.

Other host notifications can be sent if the RSDOLOW or RSDOHIGH cache partition full alerts were configured. For more information about these settings, see 8.2, "RSDOLOW/RSDOHIGH" on page 62.

## 11.2  DS8000 notifications

The DS8000 monitors the connections between itself and the TS7700. When the DS8000 detects a connection failure, events are created on the DS8000 Storage Management GUI as shown in Figure 11-1.
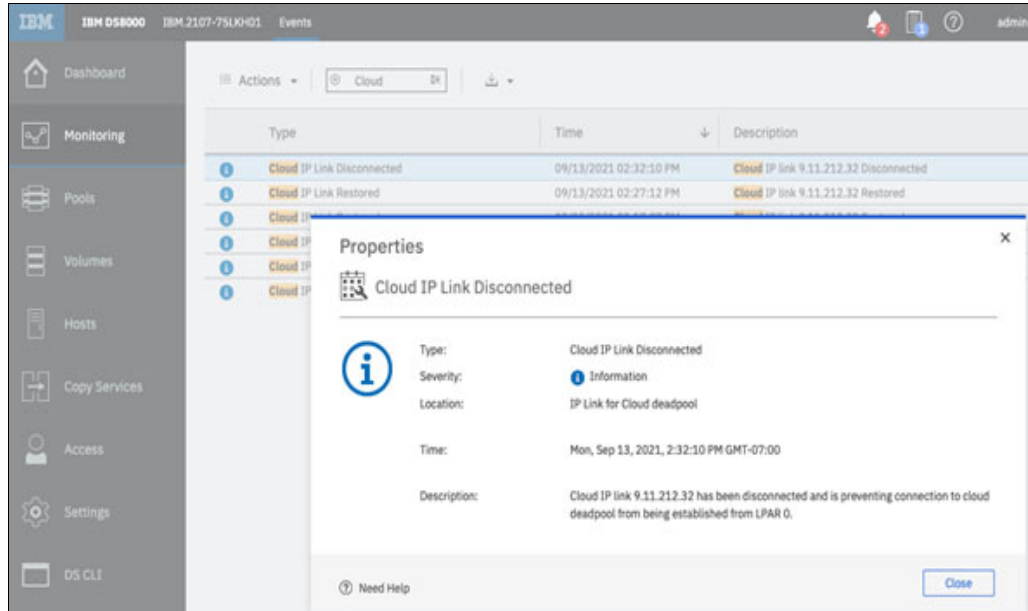


*Figure 11-1   DS8000 Event*

DS8000 discontinues to use any links that have failed but continues to monitor them. After the outage has cleared, DS8000 will resume using those links.

# A
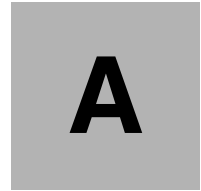
# Transparent cloud tiering setup worksheet

This appendix provides a summary worksheet of the steps needed to setup transparent cloud tiering across z/OS, DS8000, and TS7700.

Table A-1 describes each step needed for the setup process. This worksheet is only for FC 5283 Advanced Object Store configuration. Read through all the steps before beginning.

*Table A-1   z/OS, DS8000, TS7700 Setup Worksheet*

| Checklist Items | Value or Action | Notes |
|---|---|---|
| Gather TS7700 Grid Network IP address and cluster information. | First CLUSTER<br>Cluster#_____<br>IP Address1: ___.___.___.___<br>IP Address2: ___.___.___.___<br>IP Address3: ___.___.___.___<br>IP Address4: ___.___.___.___<br><br>Second CLUSTER<br>Cluster#_____<br>IP Address1: ___.___.___.___<br>IP Address2: ___.___.___.___<br>IP Address3: ___.___.___.___<br>IP Address4: ___.___.___.___ | Gather the TS7700 Grid network IP addresses. These are the primary and alternate links for each TS7700 cluster that will be an object store target. Each cluster can have two or four grid links.<br><br>Note: If you are targeting different TS7700 clusters from other DS8000 systems, use a separate worksheet. |

| Checklist Items | Value or Action | Notes |
|---|---|---|
| Gather DS8000 serial number and CEC IP Addresses. | DS8K Serial #: _____<br><br>DS8K CEC0:<br>IP Address1: ___.___.___.___<br>IP Address2: ___.___.___.___<br><br>DS8K CEC1:<br>IP Address1: ___.___.___.___<br>IP Address2: ___.___.___.___ | Gather the unique 7-digit serial number of the DS8000 being used for transparent cloud tiering.<br><br>Gather the Ethernet adapter IP Addresses that will be used for transparent cloud tiering. DS8000 could have one or two IP ports available per CEC.<br><br>Note: if you are setting up a metro mirror or global mirror system to target TS7700 clusters, use a separate worksheet for each DS8000. |
| Gather DS8000 HMC IP Addresses. | DS8K1:<br>IP Address1: ___.___.___.___<br>IP Address2: ___.___.___.___<br><br>DS8K2:<br>IP Address1: ___.___.___.___<br>IP Address2: ___.___.___.___<br><br>DS8K3:<br>IP Address1: ___.___.___.___<br>IP Address2: ___.___.___.___<br><br>DS8K4:<br>IP Address1: ___.___.___.___<br>IP Address2: ___.___.___.___ | DFSMS uses the DS8000 REST API Ethernet path through the HMC to send object metadata. Port 8452 is required to be open.<br><br>Gather the primary and alternate HMC IP Addresses for any existing primary DS8k, metro mirror, global mirror, or global metro mirror systems. |
| Verify microcode levels in the TS7700, DS8000, and zDFSMS. | Speak to your IBM Representative to determine the most current recommended microcode level to install on your systems.<br><br>Install PTFs from IBM.Function.DFSMSCloudStorage fix category | Ensure z/OS, DS8000, and TS7700 are at the appropriate code levels.<br><br>DS8000 Transparent Cloud Tiering is supported with the zSynergy Bundle. |
| Verify if TS7700 grid link Ethernet ports are enabled and cabled to the customer network. | If this is a stand-alone TS7700, grid ports are not enabled, and no Ethernet cables are connected.<br><br>Contact your IBM SSR to enable the Ethernet ports. | IBM SSR and customer network teams are required to do this work.<br><br>If this is an existing Grid, this step is not necessary as these steps have already been completed. |

| Checklist Items | Value or Action | Notes |
|---|---|---|
| Verify memory requirements. | Ensure DS8000 and TS7700 have the appropriate memory requirements for the microcode levels and for this feature. | Speak to your IBM Representative to determine whether your IBM systems contain the required memory to use this feature.<br><br>For R5.3 and later, all TS7770s must have 128 GB memory. |
| Determine whether you need 10 Gbps Ethernet on the DS8000 and TS7700. | DS8000 and TS7700 come standard with 1 Gbps Ethernet.<br><br>Determine whether 10 Gbps Ethernet exists. | 1 Gbps Ethernet is supported. However, it is recommended to upgrade to 10 Gbps Ethernet for this workload. Performance for both DS8000 Transparent Cloud Tiering data transfer and TS7700 copy transactions may be affected without 10 Gbps Ethernet. |
| Verify whether FC 5282 exists on any TS7700 in the grid. | If FC 5282 is installed on any TS7700 in the grid and you are upgrading to FC 5283, contact your IBM Representative to discuss how to remove or migrate to the new feature.<br><br>If you are unsure whether FC 5282 is installed on your TS7700 clusters, contact your IBM Representative to verify this before attempting to active FC 5283 in the grid. | FC 5282 is the original DS8000 Object Store feature. This feature is no longer available for order.<br><br>When activating FC 5283 Advanced Object Store, special procedures are run if any cluster in the grid contains FC 5282 regardless of whether it is an FC 5283 DS8000 target. |
| Verify that at least one Logical Cache Partition is available on the TS7700. | If the TS7700 is cloud attached or tape attached, the Cache Partition Management Interface (MI) page can be found under the **Logical** navigational icon. Verify that at least one cache partition is available. If cache partitions 1 – 7 are all configured, one must be removed.<br><br>If this TS7700 is a disk only system (no cloud or tape attached), then cache partitions are not used yet, and no checks are required. | FC 5283 requires at least one available logical cache partition before the installation procedure begins otherwise the procedure fails.<br><br>Each TS7700 cluster should be checked. |
| IBM SSR Installs FC 8083 on TS7700. | FC 8083 is a prerequisite for the TS7700 Advanced Object Store feature (FC 5283). This requires an IBM SSR to install two additional Power9 server SSD drives to handle object metadata management. | IBM SSR runs the FC 8083 MES process to install two additional SSD drives in each TS7700 that will be a transparent cloud tiering target.<br><br>This is a concurrent activity. |

| Checklist Items | Value or Action | Notes |
|---|---|---|
| IBM SSR installs FC 5283 on TS7700. | FC 5283 is the main TS7700 Advanced Object Store feature. This requires an IBM SSR to take the TS7700 cluster offline, run the FC 5283 process, and bring the TS7700 online. | IBM SSR runs FC 5283 process to install the Advanced Object Store feature on each TS7700 cluster that will be a transparent cloud tiering target.<br><br>This is not a concurrent activity. |
| IBM SSR enters DS8000 CEC IP Addresses on TS7700. | Provide an IBM SSR with the information documented under "Gather DS8000 serial number and IP Addresses" above.<br><br>An IBM SSR enters this information into the TS7700 using the SMIT service panel. | This allows the DS8000 into the TS7700 fire wall. If the IP addresses are not entered, the DS8000 configuration step fails.<br><br>This is a concurrent activity. |
| Resize Cache Partition on the TS7700. | After the FC 5283 MES process has completed and the TS7700 is online, resize the object cache partition.<br><br>Access the Cache Partition MI page under the **Logical** navigational icon. An object cache partition will now exist and is set to 3 TB space. Resize the cache partition as needed. | If this is a disk-only TS7700, the Cache Partition page will now be visible.<br><br>Increasing the size of this partition decreases the size of cache partition 0 (CP0), the tape logical volume resident only partition. Ensure you have enough space in CP0 before proceeding.<br><br>This must be done on each TS7700 that will be a DS8000 Transparent Cloud Tiering target.<br><br>This is a concurrent active. |
| Create or modify Object Policies on the TS7700. | After the FC 5283 process has completed and the TS7700 is online, create object policies by accessing the Object Policy MI page that is located under the *Object Store* navigational icon.<br><br>Ensure that replication copy modes are setup for each DS8000 Transparent Cloud Tiering target cluster to the proper user requirements. | After the FC 5283 process has completed, a new MI navigational icon will be available called **Object Store**. Two new MI pages are now available including the Object Policy page.<br><br>A default object policy exists. Users can modify this policy or create new ones with different replication policies.<br><br>This is a concurrent activity. |

| Checklist Items | Value or Action | Notes |
|---|---|---|
| Create Object Stores on the TS7700. | After the FC 5283 process has completed and the TS7700 is online, create object store entries by accessing the Object Store MI page that is located under the *Object Store* navigational icon.<br><br>The object store name must match the "name" set up in the z/OS ISMF panel for the *Cloud Network Connection Construct*.<br><br>Choose an object policy to use for this cloud name.<br><br>Multiple Cloud Network Connection Constructs can be created to target different object policy behavior. | After the FC 5283 process has completed, a new MI navigational icon will be available called Object Store. Two new MI pages are now available include the Object Store page.<br><br>This is a concurrent activity.<br><br>Note: The cloud name is entered into the TS7700 in lowercase. A z/OS system may display the name in uppercase. However, they are the same name. For example, $CLOUDNAMEA$ is the same as $cloudnamea$. |
| Active FC 5281 on the TS7700 if Secure Data Transfer will be used between the TS7700 and DS8000. | TS7700 customers can activate FC 5281 using the Feature License panel in the MI found under the **Settings → Cluster Settings** navigational icon.<br><br>After the feature license is activated, TS7700 customers can enable Secure Data Transfer from the Secure Data Transfer MI page found under the **Settings → Cluster Settings** navigational icon. | FC 5281 license must be activated on each TS7700 cluster that will be a DS8000 Transparent Cloud Tiering target.<br><br>Additionally, the FC 5281 license must be activated and configured (enabled) on each TS7700 cluster that will receive a copy of object data (through object replication) if secure data transfer is wanted between TS7700 clusters.<br><br>This is a concurrent activity. |
| Set up TS7700 cache partition usage alerts. | Set LI REQ OBJSET1, ALERT, RSDOLOW, and RSDOHIGH to appropriate values for the object cache partition size. | RSDOLOW and RSDOHIGH are alerts that are sent to the z/OS host through the FICON path to notify when the object cache partition reaches certain capacity limits. This is the *only* warning that is issued when object cache partitions are reaching critical capacity levels. |
| On the DS8000, set up the Ethernet ports. | Use the `setnetworkport` command on the DSCLI to configure the DS8000 IP Addresses in the routing table. | DS8000 comes standard with available Ethernet ports for transparent cloud tiering usage. However, these ports are not configured in the routing table until this step is performed. |

| Checklist Items | Value or Action | Notes |
| --- | --- | --- |
| Connect DS8000 to the TS7700 grid network. | DS8900 10 Gbps Ethernet option uses SR (short reach) and TS77700 10 Gbps Ethernet option uses LR (long reach) technology. An Ethernet switch is required in this configuration.<br><br>Ensure port 350 is open.<br><br>Cisco ACI is not supported in this environment. | This is a customer activity and requires careful planning. There are many different network configurations, and some might require conversations with network vendors and customer network teams.<br><br>This item may take the longest to plan and to set up, so planning should start early. |
| Determine whether TS7700 VTD_EXEC.369 is needed. | This VTD_EXEC is only required if the network configuration between DS8000 and TS7700 is not 100% pathing across all the links. If you already know that your configuration is not set up this way, contact your IBM Representative to request a review of VTD_EXEC.369 for your environment.<br><br>This VTD_EXEC is installed on each TS7700. | The network configuration between DS8000 CECs and the TS7700 Grid network is designed with the expectation that every DS8000 network interface will have reliable connectivity to every TS7700 network path. Each DS8000 port must be able to receive responses back from any TS7700 port configured, even those on different subnets.If this is not possible, then VTD_EXEC.369 may be required.<br><br>This is a concurrent activity. |
| Connect IBM Z to DS8000. | IBM Support required to connect the IBM Z to the DS8000 HMC. | An OSA connection between z/OS network and DS7000 HMC network is required.<br><br>This network will use RESTapi to communicate control commands and send z/OS meta data to the cloud. |
| Securing access between IBM Z and DS8000 | To ensure proper user and access security, configure RACF or a crypto-card to store CSFKEY, userids, and to set permissions.<br>Users will need to configure RACF, ICSF entry points, and CDA.<br>Additional steps are available to prohibit direct use of DFSMSdss and DFSMShsm by adding definitions to System Authorization Facility (SAF) and z/OS UNIX System Services. | To configure RACF protection for access controls to the DS8000, review the DFSMS TCT documents for configuration steps.<br><br>Use either chapter 6 and 7 of *IBM DS8000 Transparent Cloud Tiering: DS8000 Release 9.3*, SG24-8381 pages 11-13 of DFSMS transparent cloud tiering. |

| Checklist Items | Value or Action | Notes |
|---|---|---|
| Run **mkcloudserver** on the DSCLI on DS8000. | Run the **mkcloudserver** with the appropriate options.<br><br>The TS7700 Grid Network IP addresses entered previously under *Gather TS7700 Grid Network IP address and cluster information*, are used in this command.<br><br>When targeting one TS7700 cluster, all IP addresses for the First CLUSTER" should go under the *-primary7700IPs* field. If a second TS7700 will be targeted all the IP Addresses for the Second CLUSTER" should go under the *-secondary7700IPs* field.<br><br>Review this Redpaper and other publications to determine the correct options for your configuration.<br><br>Note: The *cloud_name* should be entered in the **mkcloudserver** command as all lowercase. | The **mkcloudserver** command begins communication with the TS7700 by creating and deleting a temporary container and object. If there is an issue, with the connection or permissions on the cloud, the command fails. The TS7700 should be fully setup before running this command.<br><br>If the problem persists after verifying that all the steps above have been performed and ensuring that the network paths are appropriate, open an IBM CSP ticket to have Support review DS8000 logs.<br><br>The two main reasons for failure on first-time setup are network pathing issues or missed configuration steps.<br><br>VTD_EXEC.369 might also be needed if not all paths between DS8000 and TS7700 are set up.<br><br>CISCO ACI is not supported in the environment and must be disabled.<br><br>Note: All IP addresses should target clusters within the same TS7700 grid. Targeting two clusters in different grids results in loss of access to data. |
| Configure additional object store targets by using **mkcloudserver.** | Additional object store targets can be configured on a DS8000 (up to 8). Run **mkcloudserver** for each object store name as specified in the previous step. If a second or more object store needs to be setup, run **mkcloudserver** again and then run **managecloudserver -action applypndgconfig** after each mkcloudserver for DS8000 to include the additional object stores to its configuration. | Enter the IP Addresses for the same or different TS7700s in the same grid.<br><br>Using different cloud names for the same TS7700 grid allows for different object policy behaviors. |

| Checklist Items | Value or Action | Notes |
|---|---|---|
| Configure one or more *Cloud Network Connection Construct* on the z/OS ISMF Cloud window (option S). | Create the cloud network connection construct on any production or DR z/OS system that will be running DS8000 Transparent Cloud Tiering. Configuration may be different depending on the DS8000 used.<br><br>The "*Network Connection Name*" must match the name set up in the TS7700 Object Store MI page and the name that is used in the previous `mkcloudserver` command.<br><br>Select **TAPE-OBJECT** under the Provider field when targeting a TS7700 object store. | As part of the z/OS transparent cloud tiering offering, a new *Cloud Network Connection Construct* was introduced. A *Cloud Network Connection Construct* can be defined in the ISMF Cloud window (option S).<br><br>For more information about how to configure other fields in this panel, see *IBM DS8000 Transparent Cloud Tiering: DS8000 Release 9.3*, SG24-8381. |
| Enable DS8000 Compression. | If DS8000 object compression will be used, enable TCTCOMPRESSION in DEVSUPxx parmlib. | With this setting enabled, DFSMS informs the DS8000 whether to compress object data. Any data that is already compressed on DS8000 using zEDC remains compressed going to the TS7700. Any data not already compressed is compressed by the DS8000 before sending the object to TS7700. |
| Configure z/OS host application settings. | Each host application has different setup requirements. Follow guides for those applications.<br><br>For example, when setting up for DFSMShsm data set level migrations, z/OS Management Class settings are required. If setting up for native DFSMSdss full volume dumps, JCL and potentially other setup commands such as CLOUDUTILS will be needed. | Use the appropriate sections in *DFSMS transparent cloud tiering* paper to review how to configure to the various DFSMS applications. |

| Checklist Items | Value or Action | Notes |
|---|---|---|
| z/OS DFSMSdss best practices. | ► Use an object prefix that allows for multiple volumes and multiple backups of the same volume to exist in the same container.<br>► Use CDA and the new CDACREDS keyword so credentials are not exposed in JCL.<br>► Define facility class profiles to control access to DFSMSdss dump and restore cloud functions. | |
| z/OS DFSMShsm best practices. | ► Change the default container creation cycle to 92 days, the default is 7 PATCH .ARCCVT.+50F X'5A'.<br>► Set a unique plex name per HSM plex SETSYS PLEXNAME(plexname).<br>► Define facility class profiles to control access to the DFSMShsm migrate command for users. | |

# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this paper.

## IBM Redbooks

The following IBM Redbooks publications provide additional information about the topic in this document. Note that some publications referenced in this list might be available in softcopy only.

► *IBM DS8000 Transparent Cloud Tiering: DS8000 Release 9.3*, SG24-8381

You can search for, view, download or order these documents and other Redbooks, Redpapers, Web Docs, draft and additional materials, at the following website:

**ibm.com**/redbooks

## Other publications

These publications are also relevant as further information sources:

► TS7700 Statistical Data Format:

https://www.ibm.com/support/pages/node/6354995

## Online resources

These websites are also relevant as further information sources:

► IBM DS8000 reference

https://www.ibm.com/docs/en/ds8900/9.4.0?topic=commands-mkcloudserver

► IBM Storage Community: DS8000 Transparent Cloud Tiering

https://community.ibm.com/community/user/storage/communities/community-home?CommunityKey=389634d3-147e-4db6-99d2-01852bb5a77f

► IBM z/OS CLOUDTUTILS command for DFSMSdss

https://www.ibm.com/docs/en/zos/3.1.0?topic=commands-cloudutils-command-dfsmsdss

► TS7700 Bulk Volume Information Retrieval Function User's Guide

https://www.ibm.com/support/pages/ts7700-bulk-volume-information-retrieval-function-users-guide

► TS7700 Statistical Data Format white-paper

https://www.ibm.com/support/pages/node/6354995

- ► z/OS DFSMS Transparent Cloud Tiering Support Document

  https://community.ibm.com/community/user/storage/viewdocument/zos-dfsms-transpa
  rent-cloud-tierin?CommunityKey=389634d3-147e-4db6-99d2-01852bb5a77f&tab=library
  documents

# Help from IBM

IBM Support and downloads

**ibm.com**/support

IBM Global Services

**ibm.com**/services

**ibm.com**/redbooks