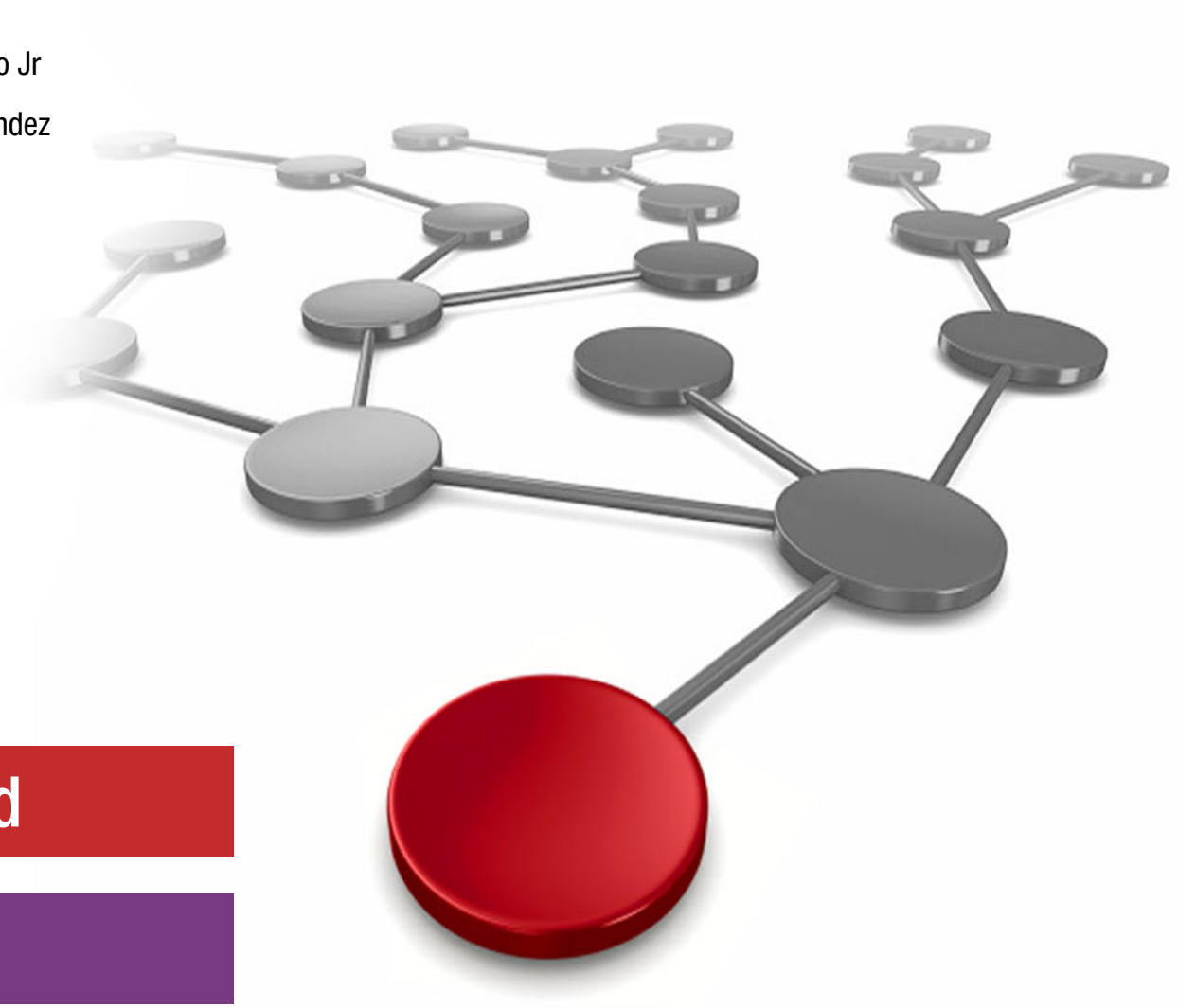
Redbooks
ibm.com/redbooks

IBM DS8880 and z/OS DFSMS: Transparent Cloud Tiering

Jose Gilberto Biondo Jr
Orlando Ariel Fernandez
Robert Gensler
Eddie Lin



Cloud

Storage

IBM

Redbooks

International Technical Support Organization

IBM DS8880 and z/OS DFSMS: Transparent Cloud Tiering

April 2018

SG24-8381-01

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

Second Edition (April 2018)

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.

AIX®
DS8000®
Easy Tier®
FICON®
FlashCopy®
HyperSwap®
IBM®
IBM Z®
RACF®
Redbooks®
Redbooks (logo)  ®
WebSphere®
z/OS®

The following terms are trademarks of other companies:

Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Preface

- This IBM® Redbooks® publication gives a broad understanding of storage clouds and the initial functionality that was introduced for mainframes to have Transparent Cloud Tiering.

IBM DFSMS and the IBM DS8880 added functionality to provide elements of serverless data movement, and for IBM z/OS® to communicate with a storage cloud. They introduced the following key areas:

- A gateway in the DS8880, which allows the movement of data to and from Object Storage be using a network connection.
- DFSMShsm enhancements to support Migrate and Recall functions to and from the Object Storage. Other commands were enhanced to monitor and report on the new functionality.
- DFSMShsm uses the Web Enablement toolkit for z/OS to create and access the metadata for specific clouds, containers, and objects.
- DFSMSdss enhancements to provide some basic backup and restore functions to and from the cloud.

This IBM Redbooks publication is divided into the following parts:

- Part 1 provides you with an introduction to clouds. You might be new to clouds or have a confused view of cloud terminology. If so, Part 1 is helpful in providing you with the basic knowledge you need.
- Part 2 shows you how we set up the Transparent Cloud Tiering in a controlled laboratory and how the new functions work. We provide points to consider to help you set up your storage cloud and integrate it into your operational environment.
- Part 3 shows you how we used the new functionality to communicate with the cloud and to send data and retrieve data from it.

Authors

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, Poughkeepsie Center.

Jose Gilberto Biondo Jr is an IT Specialist in Integrated Technology Delivery, ServerSystems Operations/Storage Management in IBM Brazil. He has nine years of experience in z/OS, working with storage management since 2007. Jose has written several IBM Redbooks publications. Although he works mainly with IBM storage products (DFSMSdfp, DFSMSdss, DFSMShsm, and DFSMSrmm), he also works with OEM software products. Jose's areas of expertise include installing and maintaining storage products, and process automation.

Orlando Ariel Fernandez is a Senior Technical Staff Member with the IBM Systems Unit, in Tucson, Arizona. He is the lead architect for z/OS DFSMShsm, which provides automated, policy-driven space management and backup/recovery for enterprise systems data. Glenn has over 25 years of experience developing storage software solutions.

Robert Gensler is a Senior Software Engineer with the IBM Systems Unit in Tucson. He has 17 years of experience in DFSMSdss. He holds a Master's degree in Computer Science from The University of Arizona.

Eddie Lin is a Senior Technical Staff Member with IBM Systems Storage located in Tucson, AZ. He is an architect for the IBM DS8000® Enterprise Storage product specializing in developing enablement and solutions in the realm of Cloud Storage and Cloud Computing. Currently Eddie is the lead architect for the DS8880 Transparent Cloud Tiering solution. Eddie has over 15 years of experience developing from the original DS8000 to the latest DS8880 line of enterprise storage.

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

- Lydia Parziale (IBM Redbooks Project Manager, IBM)
- Larry Coyne (IBM Redbooks Storage Project Leader, IBM) for advice and guidance on various aspects of storage techniques and devices.
- Bertrand Dufrasne IBM, IBM Redbooks Storage Project Leader
- Barbara McDonald, IBM DFSMS Chief Product Owner

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 ITSO 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 books to be as helpful as possible. Send us your comments about this book 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, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks
- Follow us on Twitter:

  http://twitter.com/ibmredbooks
- Look for 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

Part 1

Cloud Preparation

In this part, we introduce topics that are related to a storage cloud. The aim is to provide a basic understanding of storage clouds. If you are new to clouds, it is important you read this section.

This part includes the following chapters:

1

Storage Tiering History and the Value of TCT

In this chapter, we introduce you to storage tiering to help increase your understanding of how storage can be tiered to accommodate an application's data needs. We will also introduce you to Transparent Cloud Tiering (TCT) and the value it brings to clients.

The management of data across tiers is available for you to use across several devices on your site. We included this topic to show you what can be in place and why you might want to use cloud as a tiering option based on TCT.

This chapter includes the following topics:

1.1 Storage tiers

Systems have a finite amount of resources that can be used to store data, whether on online or with auxiliary storage. The use of different storage media and categorizing the data within each layer can be an efficient way to manage storage resources. This management allows critical data to be available in high-performance devices, and other data on lower-cost devices.

There are two solutions available for storage tiering: Hardware and software implementation. Each technique can be used alone, or combined, to provide improved data management, and storage efficiency.

The next two sections introduce the concepts behind hardware and software tiering.

1.2 Hardware layers overview

The hardware layers consist of the following storage areas:

- Custom Flash technologies
- Solid-state disk (SSD)
- Enterprise
- Nearline
- Tape

Storage media that is used in mainframe systems has changed dramatically, from drives that can store a few megabytes, to current drives that can store terabytes, or flash technologies that provide increased performance gains over spinning disk drives.

You can configure a DS8880 with a mix of flash cards, SSD, Enterprise, and Nearline volumes with IBM Easy Tier®. This approach enables the system hardware to constantly monitor data extents (or track group, for small extents).

It identifies data temperature based on the number of accesses that are requested within a period, and takes the appropriate actions to maintain efficiency. An extent can become *hot*. In this case, hot means that the I/O workload of the extent's data is higher, compared to other extents in the same extent pool and in the same tier. In this case, it is moved to a higher performance tier. When the data becomes cold, meaning it is being less frequently accessed than other data, it is demoted to slower, cost-efficient volumes.

Figure 1-1 shows a logical volume with hot and cold extents that are allocated on high-performance and cost-efficient hardware.

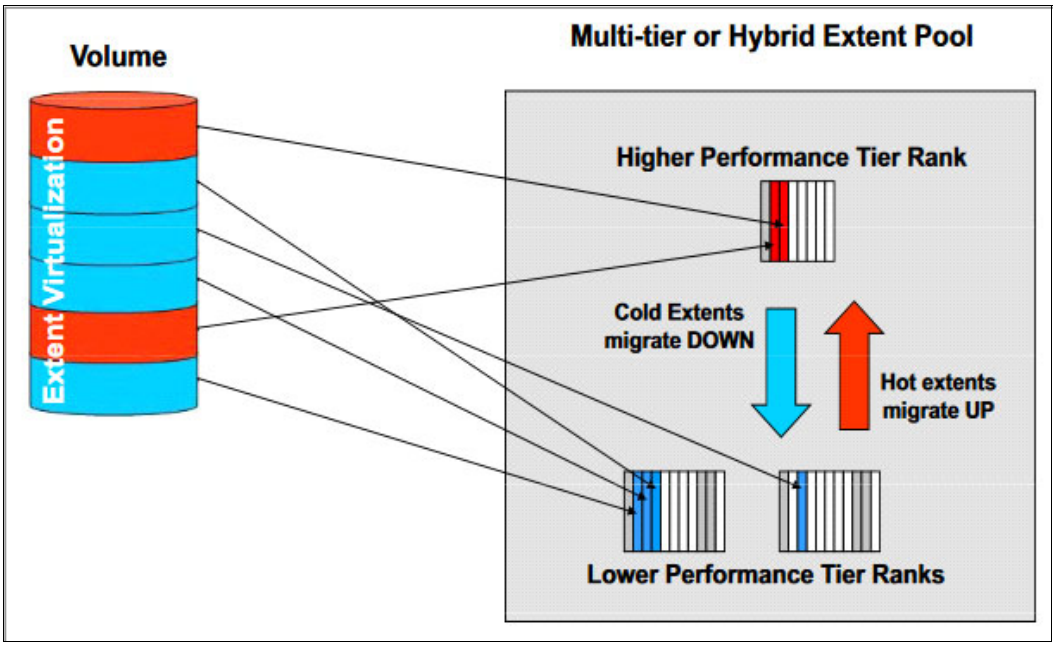


Figure 1-1 Hot and cold extents

IBM Easy Tier monitors data within a single tier and monitors ranks to ensure that one rank is not overloaded while others are idle. This function is called auto-rebalance, or intra-tier rebalancing, and is triggered every 6 hours, if required.

Auto-rebalance migrates extents across ranks within a storage tier to achieve these goals:

- A balanced workload distribution across the ranks and to avoid hotspots.
- Consequently, reduced performance skew within a storage tier and the best available I/O performance from each tier.

The extents are migrated only within the same extent pool. So to take advantage of this feature, at least two ranks must be defined in an extent pool.

Another layer that is available under storage hardware management includes offline media. Offline media includes any media that cannot be directly accessed by a computer, unless it is made available to the system.

Virtual and physical tapes are considered offline media, and can also be part of hardware storage layers. Although the direct access storage device's (DASD) hardware cannot directly migrate data to tapes, you can use software tiers to accomplish this migration. Also, Virtual Tape Libraries use disk storage to emulate tapes, which are offloaded to physical tapes when the cache is full. This process is known as *pre-migration*.

When the requested data is stored in physical tapes, they are mounted and the virtual tape content is loaded into the Virtual Tape Library and made available for z/OS read. The process of recovering data from physical tapes to cache is also known as *recall*.

1.3 Software layers

The software layers consist of the following storage areas:

- Level 0
- Migration Level 1
- Migration Level 2

Unlike hardware tiers, there are two types of storage devices available from a z/OS perspective: disk and tape. It is necessary to have a storage management product, such as DFSMShsm, to enable the use of software tiers. DFSMShsm can manage data by migrating, recalling, backing up, and recovering data sets as required.

DFSMShsm can manage storage tiers by using the SMS Management Class construct to identify data sets that can be moved to other tiers based on time since last access or creation. The following tiers are available on DFSMShsm:

- Primary volumes (L0)

  These SMS or non-SMS DASD volumes store online data, and can be directly accessed by TSO users, Jobs, or applications. These volumes are managed by DFSMShsm, but are not owned by it.

- Migration Level 1 (ML1) volumes

  These non-SMS DASD volumes contain data that is migrated from L0 volumes based on Management Class attributes. The data in these volumes is owned by DFSMShsm and cannot be directly read by users or applications. If a read/write operation is required, the data set is recalled to L0 volumes before they can be read. To use a volume as ML1, an ADDVOL command must be included on DFSMShsm parmlib or dynamically added. If dynamically added, this configuration is reset during DFSMShsm restart.

- Migration Level 2 (ML2) volumes

  This second level of DFSMShsm migration often is designated for large data sets or long retention periods. These volumes are a set of non-SMS tapes (physical or virtual) or low-performance DASD volumes that are owned by DFSMShsm. The data in these volumes cannot be directly read by users or applications unless they are recalled to L0 volumes first.

Class Transition

A class transition is a change in the object's management class or storage class. Class transition was introduced in z/OS V2R1, which enables DFSMShsm to also manage and migrate data set laterally within L0 volumes. By implementing the use of class transition, you can create pools with different performance levels and move your data between these volumes as they age.

Newly created data sets might require high-performance levels. These performance requirements can decrease as the data ages, but the data still must be accessed. In this case, migrating the data to ML1 volumes is not an option, because this data is still required by applications. However, leaving this data on high-performance DASD for an extended period reduces the ROI on the high-performance DASD and might deny access for other data with high-performance needs.

Using class transition provides a balanced approach to managing the change in performance needs of data by allowing DFSMShsm to migrate data sets between different Storage Groups. It uses Management Class attributes to define the following migration policies:

- Time since creation

- Time since last use
- Periodic transition

Class transitioning can be started through Primary, Interval, or on-demand migrations. It can also be started by using a user-issued command. An example of this command would be:

```
HSEND MIGRATE DSN(/) TRANSITION
```

After it is started, it references Management Class policies to select the data sets for transition. If it is eligible, DFSMShsm starts ACS routines to assign a new Storage Class, Management Class, or Storage Group. If the Storage Group changes, DFSMShsm attempts to move the data set to the new pool.

Figure 1-2 shows the sample implementation of class transition. The data is first allocated in high-performance DASD, and then it will transition to cost-efficient devices as the data ages and the data's performance requirement drops. Later, you can migrate the data to even more cost-efficient levels (ML1/ML2).

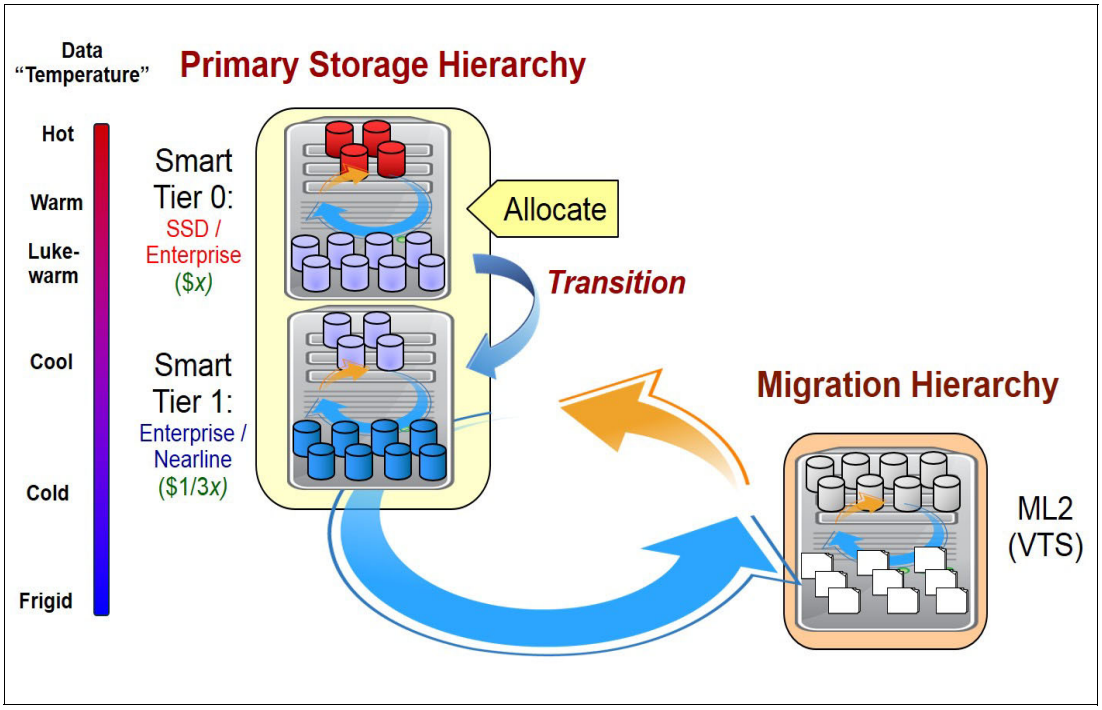


Figure 1-2 Smart tiers within the primary hierarchy

The tiering capability adds depth to a storage management strategy. If your data storage consists of a "flat" structure, such as being held in a single large DASD pool, your options for application quality are limited. A multi-tiered structure (as shown in Figure 1-2) provides opportunities to enrich your business applications through the following qualities of service:

- Higher availability levels
- Performance improvements through data positioning
- High-quality data management through organized constructs and tier migration

1.4 DFSMShsm behavior without TCT

DFSMShsm is the z/OS component responsible for performing the Information Lifecycle Management task. As such, it is responsible for moving data between the different storage

tiers, including both Online and Offline media types, like DASD and Tape respectively, based on pre-defined policies and data access availability needs.

Figure 1-3 on page 8 shows how the data flows between these distinct technologies, through DFSMShsm, and the challenges related to life cycle management.

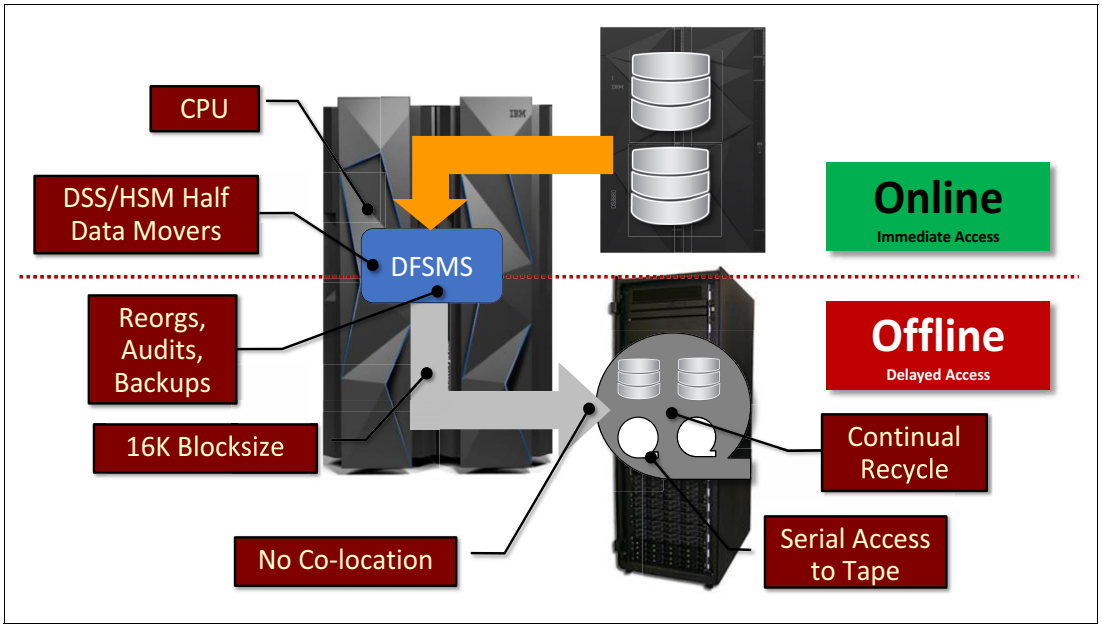


Figure 1-3 DFSMShsm data movement between DASD and Tape technologies

To move the data between the tiers, DFSMS uses HSM and DSS Data Movers to read the data from the source storage and write the data to the target storage, via IBM FICON® connectivity. Online media access format is different than Offline media access format.

During the processing of the data movement from disk to tape, DSS reads the data from the source media, passes it to HSM, which converts the data into 16K blocks, and writes the data to tape. This movement of data from disk to tape flows through the mainframe, and consumes extra CPU cycles.

As you can imagine, to process all these operations the system will use IBM Z CPU cycles that could be used for other important workloads, such as business applications.

Other challenges of migrating data to tape include the lack of colocation for the data sets, meaning data with different retention will be placed in the same tape, which will also create the need of a recycle process to increase tape usage efficiency. The serial access to tape also prevents the recall of multiple data sets that are stored on the same tape. Otherwise, the system tries to run those data sets concurrently.

1.5 Introducing TCT

Transparent Cloud Tiering for IBM DS8880 was introduced to help customers to use IBM Z resources more efficiently. With its integration with z/OS through DFSMShsm, it allows clients to reduce CPU utilization by eliminating constraints that are tied to original Tape methodologies.

This is accomplished by enabling direct data movement from IBM DS8880 to cloud object storage, without the need of the data having to go through the host. DFSMS communicates with DS8880 through a REST API interface and issues commands for the DS8880 to move the data directly to/from the Cloud, as shown in the Figure 1-4 on page 9.

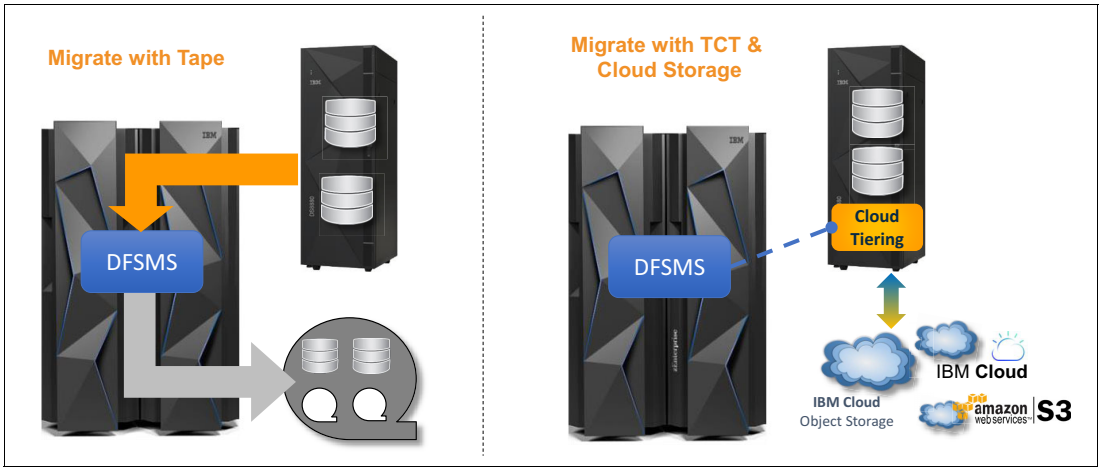


Figure 1-4 TCT data movement layout

In this way, TCT offloads all the actual data movement processing-related workload from z/OS. Significant CPU savings result, as compared with the traditional data movement methods, especially when considering large data sets.

The CPU savings expected are achieved by reducing or eliminating CPU processing for the following tasks:

- Tape recycle
- Dual (DSS and HSM) data movement
- Moving data through CPU
- Reblocking data to 16K blocks

This also gives organizations flexibility to choose the most appropriate Offline media option, depending on cost, performance, and availability requirements.

2

Cloud Overview

In this chapter, we introduce you to cloud concepts and explain how IBM Transparent Cloud Tiering (TCT) enables cloud integration with IBM DS8880 systems that run z/OS environments. You get a basic understanding of what a cloud is in the context of TCT, through a short description of Cloud Storage versus Cloud Computing.

2.1 What defines cloud in the context of TCT?

The cloud is a combination of several different solutions, components and services. It consists of different layers, including but not limited to:

- Application Layer: where applications can be hosted and run, and can also take advantage of pre-coded software APIs that you can integrate to create new applications.
- Infrastructure Layer: where entire systems can be hosted. An Infrastructure Layer can be composed of a mix of cloud and traditional infrastructures, interconnected between each other.

The Infrastructure Layer is composed of three main classes of components:

- Storage Layer
- Network Layer
- Compute Layer

Within the Storage Layer, we have three Storage types:

- Block Storage
- File Storage
- Object Storage

TCT uses Object Storage architecture for storing data sets. We will cover this in more detail in this chapter.

In the Figure 2-1 we provide a comprehensive diagram showing the different layers and where TCT integrates with the cloud:

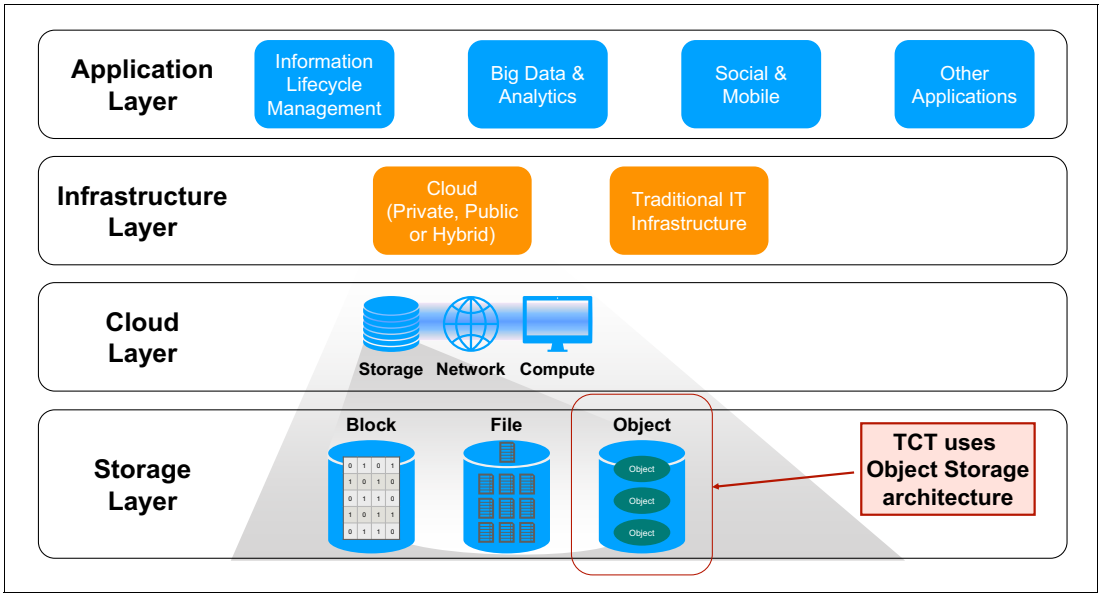


Figure 2-1 TCT in the context of the Cloud

2.2 Compute cloud versus storage cloud

In short, the compute cloud provides the necessary components to run the applications for processing resources, and is where you can deploy and run your chosen software, including operating systems and applications.

On the other hand, the storage cloud holds the data and caters for operations like backing up and archiving data. As mentioned earlier in this chapter, TCT uses Object Storage architecture to store the data it manages in the cloud.

2.3 Types of Storage

Before you use storage clouds, you must understand what a gateway is, its function, and how the data is managed between the mainframe and the cloud.

The following types of architectures (see Figure 2-2) can be used for storing data, where each type has its advantages:

- Block and File: This architecture is used on mainframes and other operating systems to store data. It has the advantages of being faster, IOPS-centric, flash-optimized, and allows various approaches.
- Object Storage: This architecture provides larger storage environments, or cool/cold data, which can scale to petabytes of data while being cloud-compatible.

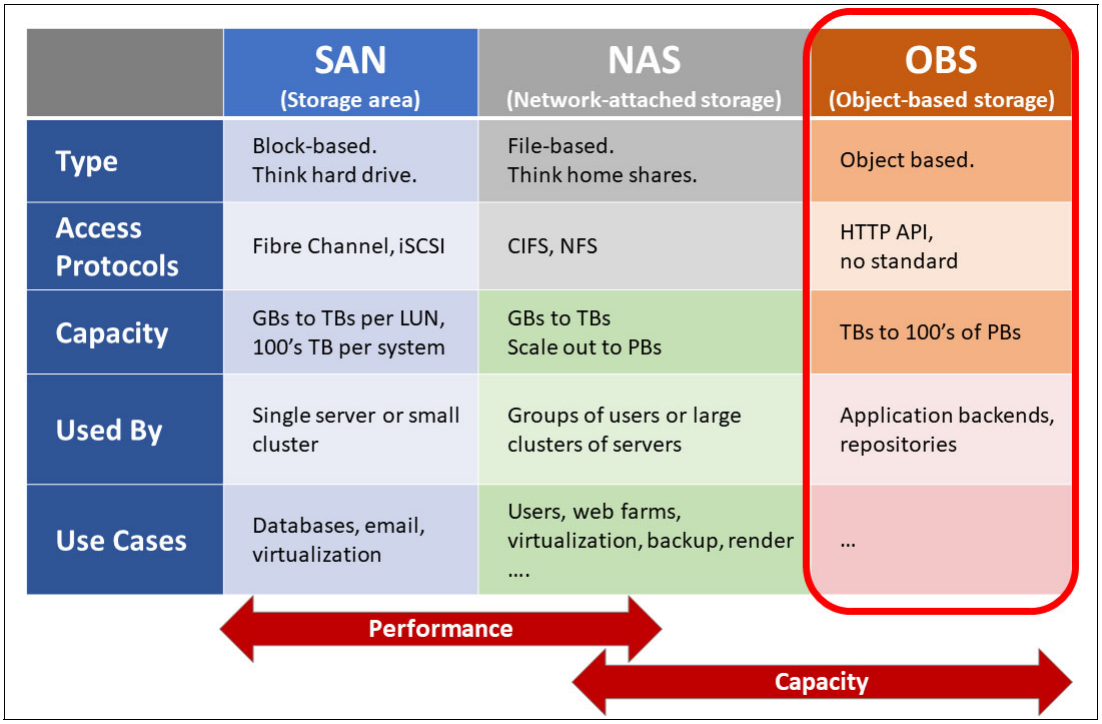


Figure 2-2 Types of storage

Having a storage cloud that uses object storage has several benefits. A storage cloud significantly reduces the complexity of storage systems by simplifying data scaling within a single namespace. Also, the REST protocol is used for communication between the server and the client. The use of high-density, low-cost commodity hardware turns storage clouds into a scalable, cost-efficient storage option.

The Transparent Cloud Tiering function of IBM DS8880 provides a method to convert the block to Object Storage without additional hardware on the LAN.

On storage clouds, the data is managed as objects, unlike other architectures that manage data as a block of storage, as it is done on mainframes. For this reason, the communication

between mainframe systems and storage cloud is done by an application responsible for converting cloud storage APIs, such as SOAP or REST, to block-based protocols, such as iSCSI or Fibre Channel, when necessary.

Therefore, the storage cloud can be considered an auxiliary storage option for mainframe systems to be used by applications, such as these:

- DFSMShsm to migrate and recall data sets
- DFSMSdss to store data that is generated by using the DUMP command

2.4 Cloud Storage delivery models

Cloud delivery models refer to how a cloud solution is used by an organization, where the data is located, and who operates the cloud solution. There are multiple delivery models that can deliver the capabilities needed in a cloud solution.

The cloud delivery models are as follows:

- Public cloud
- Private cloud
- Hybrid cloud

These delivery models can be integrated with traditional IT systems and other clouds. They are divided into two categories:

- *On premise:* Consists of a private cloud infrastructure at your organization's location.
- *Off premise:* Consists of a cloud infrastructure being hosted in a cloud service provider's location.

Public Cloud

A *public cloud* is a solution in which the cloud infrastructure is available to the general public or a large industry group over the internet. The infrastructure is not owned by the user, but by an organization that provides cloud services. Services can be provided at no cost, as a subscription, or as a pay-as-you-go model.

There is another delivery model option known as *community cloud*, or *multi-tenant cloud*, which typically consists of a public cloud that is shared among multiple organizations, to lower costs. For ease of understanding, this book treats this delivery model as part of the public cloud category.

Private Cloud

A *private cloud* is a solution in which the infrastructure is provisioned for the exclusive use of a single organization. The organization often acts as a cloud service provider to internal business units that obtain all of the benefits of a cloud without having to provision their own infrastructure. By consolidating and centralizing services into a cloud, the organization benefits from centralized service management and economies of scale.

A private cloud provides an organization with some advantages over a public cloud. The organization gains greater control over the resources that make up the cloud. In addition, private clouds are ideal when the type of work that is being done is not practical for a public cloud because of network latency, security, or regulatory concerns.

A private cloud can be owned, managed, and operated by the organization, a third party, or a combination of the two. The private cloud infrastructure is provisioned on the organization's premises, but it can also be hosted in a data center that is owned by a third party.

Hybrid Cloud

As the name implies, a *hybrid cloud* is a combination of various cloud types (public, private, and community). Each cloud in the hybrid mix remains a unique entity, but is bound to the mix by technology that enables data and application portability.

The hybrid approach allows a business to use the scalability and cost-effectiveness of a public cloud without making available applications and data beyond the corporate intranet. A well-constructed hybrid cloud can service secure, mission-critical processes, such as receiving customer payments (a private cloud service) and secondary processes, such as employee payroll processing (a public cloud service).

IBM Cloud Object Storage and TCT

IBM Cloud Object Storage (COS) offers all the delivery model options described above. The table below provides a summary of each option, with its capabilities:

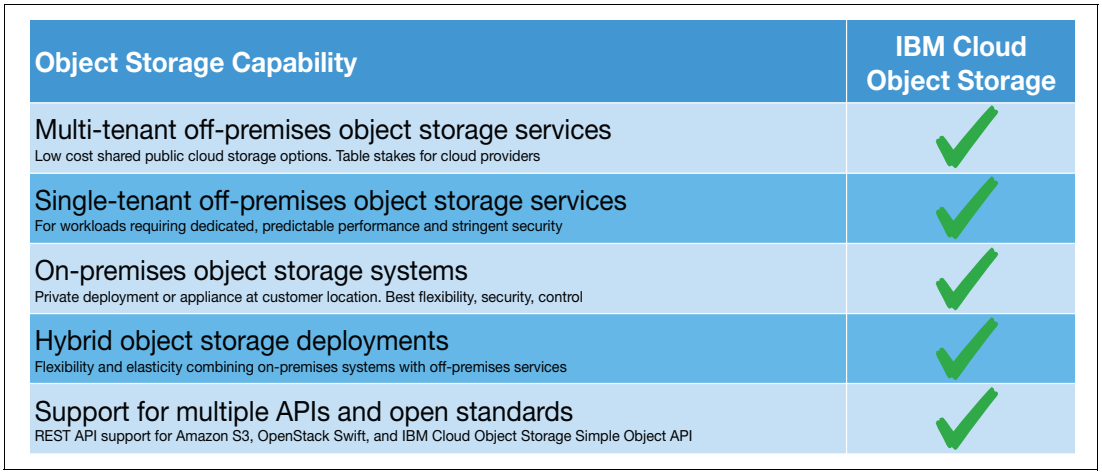


Figure 2-3 IBM Cloud Object Storage capabilities

IBM Transparent Cloud Tiering solution provides an integration of the IBM DS8880 storage system, when running in z/OS environments, with a Cloud Object Storage infrastructure, that can be any of the options described above.

For detailed information about the IBM Cloud Object Storage service offering, refer to the *Cloud Object Storage as a Service: IBM Cloud Object Storage from Theory to Practice - For developers, IT architects and IT specialists*, SG24-8385 Redbooks publication or go to the IBM Cloud Object Storage website at this link:

`http://www.ibm.com/cloud/object-storage`

For other IBM Cloud Storage solutions, refer to the *IBM Private, Public, and Hybrid Cloud Storage Solutions*, REDP-4873 Redbooks publication or go to the IBM Cloud website at the link:

`http://www.ibm.com/cloud/solutions/`

2.5 Object Storage hierarchy

Data that is written to a cloud by using Transparent Cloud Tiering is stored as objects and organized into a hierarchy. The hierarchy consists of accounts, containers, and objects. An account can feature one or more containers and a container can include zero or more objects.

2.5.1 Storage cloud hierarchy

The storage cloud hierarchy consists of the following entities:

- Account
- Containers
- Objects

Each entity plays a specific role on data store, list, and retrieval by providing a namespace, the space for storage, or the objects. There also are different types of objects, data, and metadata.

A sample cloud hierarchy structure is shown in Figure 2-4. Each storage cloud component is described next.

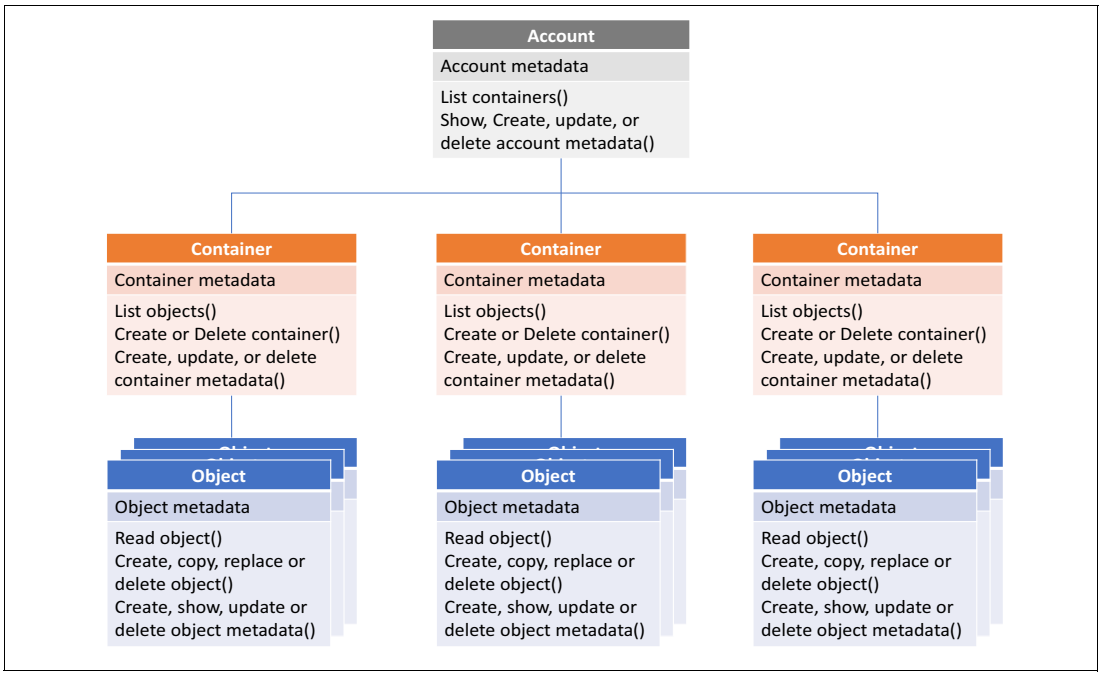


Figure 2-4 Cloud hierarchy

Account

An account is the top level of the hierarchy and is created by the service provider, but owned by the consumer. Accounts can also be referred to as *projects* or *tenants* and provide a namespace for the containers. An account has an owner that is associated with it, and the owner of the account has full access to all the containers and objects within the account.

The following operations can be done from an account:

- List containers
- Create, update, or delete account metadata
- Show account metadata

Containers

Containers (or buckets) are similar to folders in Windows or UNIX, and provide an area to organize and store objects, container-to-container synchronization, quotas, and object versioning. One main difference is that containers cannot be nested. That is, no support is available for creating a container within another container. Container names can be 256 bytes.

Access to objects within a container are protected by using read and write Access Control Lists (ACLs). There is no security mechanism to protect an individual object within a container. After a user is granted access to a container, that user can access all of the objects within that container.

The following operations are supported for containers:

- List objects
- Create container
- Delete container
- Create, update, or delete container metadata
- Show container metadata

Objects

As of this writing, there is a 5 GB limit to the size of an object due to an Openstack Swift restriction. Objects larger than 5 GB must be broken up and stored by using multiple segment objects. After all of the segment objects are stored, a *manifest* object is created to piece all of the segments together. When a large object is retrieved, the manifest object is supplied and the Object Storage service concatenates all of the segments and returns them to the requester. For most sizes greater than 100MB, the system does "multi-part uploads." By creating multiple parts, the system does parallel recall and greater recall efficiency is achieved.

The following operations are supported for objects:

- Read object
- Create or replace object
- Copy object
- Delete object
- Show object metadata
- Create, update, or delete object metadata

The objects can also have a defined, individual expiration date. The expiration dates can be set when an object is stored and modified by updating the object metadata. When the expiration date is reached, the object and its metadata are automatically deleted.

However, the expiration does not update information about the z/OS host; therefore, DFSMShsm and DFSMSdss do not use this feature. Instead, DFSMShsm handles the expiration of objects.

User-created backups (backups that are created outside of DFSMShsm to the cloud) must be managed by the user. Therefore, a user must go out to a cloud and manually delete backups that are no longer valid. At present, this process is not recommended.

2.5.2 Metadata

In addition to the objects, metadata is recorded for account, container, and object information. Metadata consists of data that contains information about the stored data. Some metadata

information might include data creation and expiration date, size, owner, last access, and other pertinent information.

The difference between data and metadata is shown in Figure 2-5.

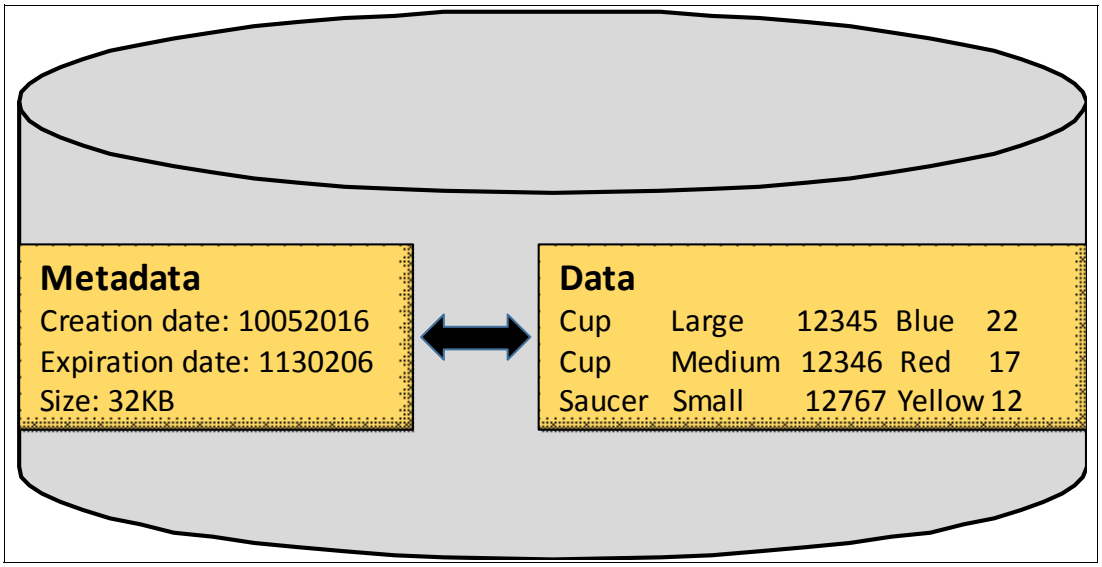


Figure 2-5 Data and metadata differences

3

Transparent cloud tiering

In this chapter, we extend the tiering process a step further. We add another tier, which is the cloud.

Good storage management practices are based on one principle in particular. You can configure that physical space in logical pools that are dynamically reconfigured to increase or decrease the storage capacity that is available for use. This reconfiguration should also be transparent to the user.

Storage Cloud on mainframes introduces a new storage tier that provides extended storage capacity at a lower cost while making the data available from different locations.

This chapter includes the following topics:

3.1 FICON and TCP/IP data movement

Traditional data movement that DFSMShsm does during availability and space management operations are performed over the FICON protocol. The data is usually transferred between the DASD controller, the CPU, and a tape controller. With the introduction of transparent cloud tiering, and the possibility of using Object Storage clouds, the data is transferred to the cloud using TCP/IP protocols.

With transparent cloud tiering, the DS8000 is responsible for communicating and transferring/receiving the data from the storage cloud through an ethernet link. The metadata on the other hand, can be transferred by the host system or the DS8000, depending on the cloud type used.

3.1.1 Data flow on Swift cloud type

When a Swift type is used to communicate with the cloud, the DS8000 is responsible for archiving and retrieving the actual data to the cloud. The host communicates with the DS8000 through the FICON - Fibre Channel protocol, and sends the metadata directly to the cloud through a TCP/IP connection, as shown on Figure 3-1.

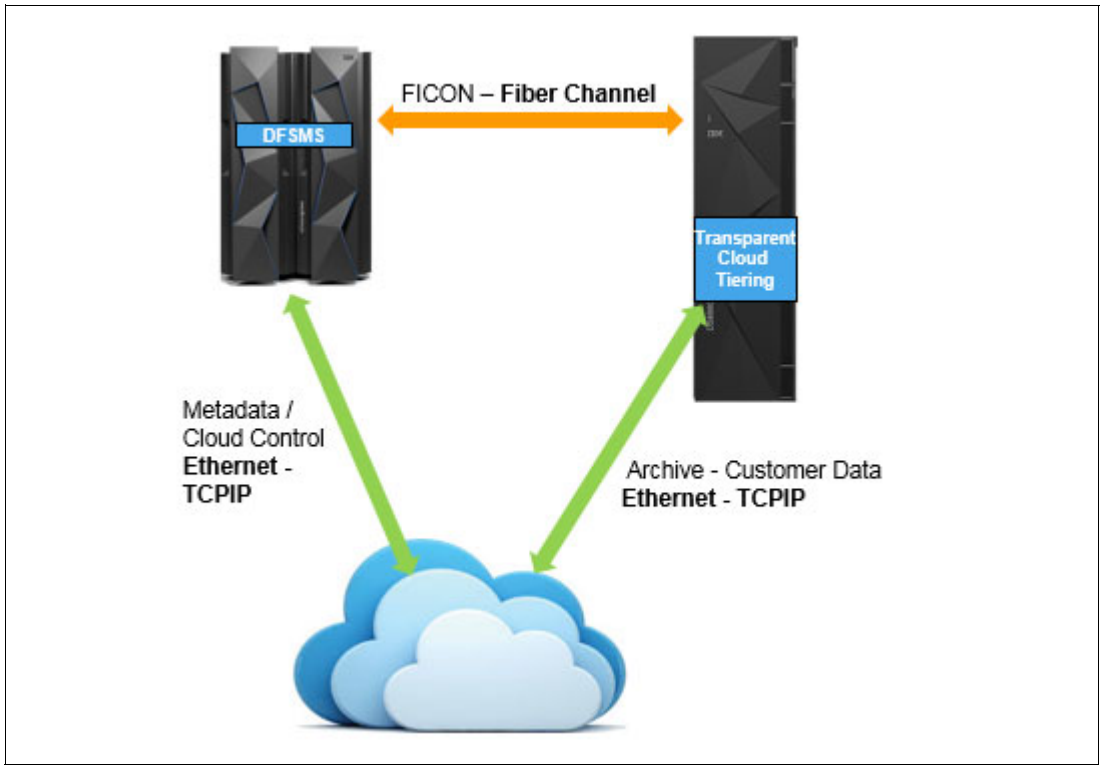


Figure 3-1 Cloud communication with Swift API

The SMS configuration will also have the cloud information, including the cloud endpoint, username, password and so on. DFSMShsm also uses this service to control the cloud. For example, DFSMShsm uses the service to create and list containers and objects, and to run utility commands for reporting and auditing.

3.1.2 Data flow on S3 and IBM COS types

The integration between z/OS and S3 and IBM COS types is a little different from the one seen on Swift. While the actual data is still transferred by the DS8000, the z/OS host no longer communicates with the cloud. Instead, the host communicates with the DS8000 through an Ethernet TCP/IP link to send the metadata to the DS8000, which then stores the metadata in the cloud, as shown in Figure 3-2.

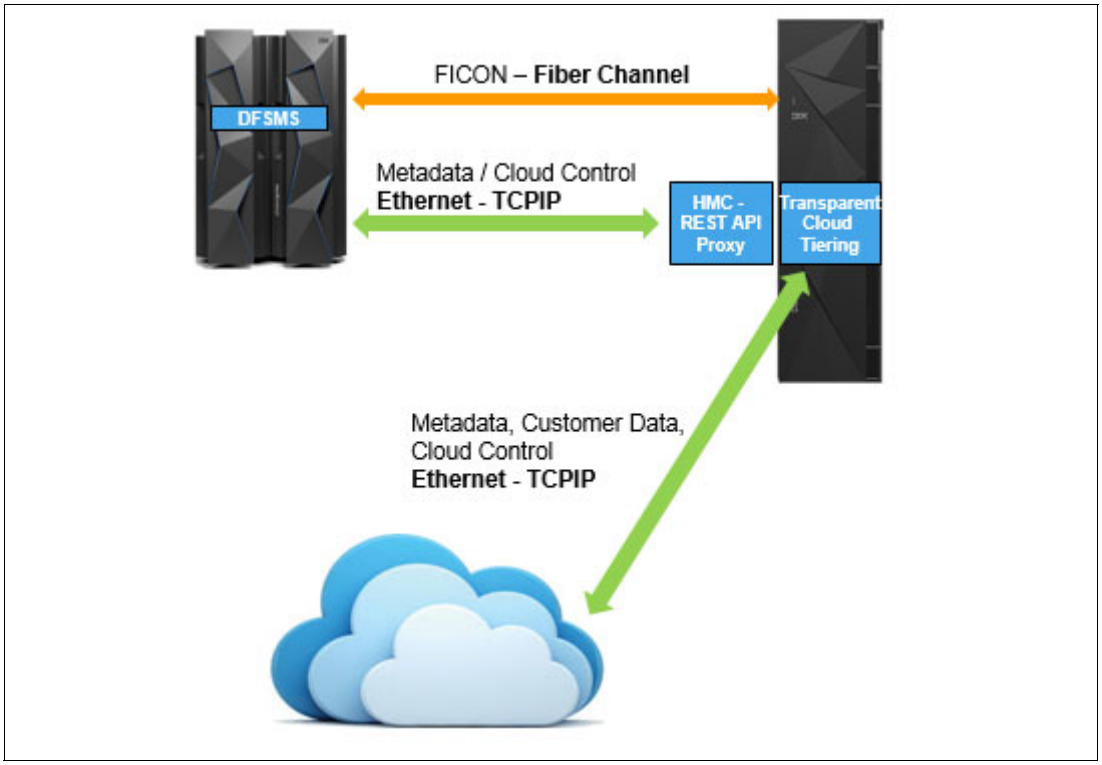


Figure 3-2 Cloud communication with S3 and IBM COS APIs

With the S3 and IBM COS Cloud Types, metadata and cloud control commands now flow to a REST API proxy which is located in the HMC of the DS8880. A user ID is necessary to connect to the DS8000, which can be defined either locally, or in an LDAP server, if it is used for authentication.

Also, the SMS is configured such that the endpoint, user and password are concatenated to form the HMC URL. Also, the username and password of local/LDAP user is created on the HMC. With S3 and IBM COS, all data (metadata and data) is transferred to the cloud through the DS8000.

3.1.3 Bandwidth considerations

As previously mentioned, the data and metadata is transferred to the cloud using the ethernet TCP/IP protocol. If you use S3 or IBM COS, all this information is sent by the DS8000. So it is important to plan for bandwidth availability to prevent bottlenecks.

At the time of the writing of this book, the DS8000 has two 1Gbit ethernet ports. You can use them to connect to the network through Central Electronic Complex (CEC), resulting in a total of 4Gbits maximum bandwidth available per DS8000.

When storing or retrieving data from the cloud, the system optimally uses the ethernet card of the CEC that owns the Logical Subsystem (LSS) associated with the request. This practice might lead to unbalanced migrations and recalls if the majority of the data is on a specific LSS.

Defining your storage groups with volumes from LSS that belong to both storage facility images can reduce the chances of having an unbalanced link usage.

3.2 Storing and retrieving data by using DFSMS

From a z/OS perspective, the storage cloud is an auxiliary storage option, but unlike tapes, it does not provide a block-level I/O interface. Instead, it provides only a simple HTTP get-and-put interface that works at the object level.

When data sets are stored in the cloud, the number of objects that are created varies. The determining factors are the data set attributes (multi-volume, data set size, and VSAM), and the metadata that is created. For each volume that a data set is on, a new object is created within the container. One or more metadata objects are also created in the same container.

DFSMSdss can be used to store logical dumps and restore data sets from the cloud. It is also used as the data mover by DFSMShsm when data is migrated to or recalled from the cloud. In each case, the number of objects that is created can vary, based on the following factors:

- The number of volumes the data set is on. For each volume the data set is stored on, a new object is created.
- The size of the data set. As of this writing, there is no object maximum size and multi-part-uploads for almost any size of data sets are now being done.
- VSAM data sets. When VSAM data sets are migrated to cloud, each component has its own object, meaning a key-sequenced data set (KSDS) has at least one object for the data component and another for the Index. The same concept is applied to alternative indexes.

Also, several metadata objects are created to store information about the data set and application. Table 3-1 lists some objects that are created as part of the DFSMSdss dump process.

Table 3-1 Created objects

Object name	Description
Objectprefix/HDR	Metadata object that contains ADRTAPB prefix.
objectprefix/DTPDSNLnnnnnnn	n = list sequence in hexadecimal. Metadata object that contains a list of data set names successfully dumped. **Note:** This object differs from dump processing that uses OUTDD where the list consists of possibly dumped data sets. For Cloud processing, this list includes data sets that were successfully dumped.
objectprefix/dsname/DTPDSHDR	Metadata object that contains data set dumped. If necessary, this object also contains DTCDFATT and DTDSAIR.

Object name	Description
objectprefix/dsname/DTPVOLDnn/desc/META	Metadata object that contains attributes of the data set dumped: ► desc = descriptor ► NVSM = NONVSAM ► DATA = VSAM Data Component ► INDX = VSAM Index Component ► nn = volume sequence in decimal, 'nn' is determined from DTDNVOL field inDTDSHDR
objectprefix/dsname/DTPSPHDR	Metadata object that contains Sphere information. If necessary, this object also contains DTSAIXS, DTSINFO, and DTSPATHD. Present if DTDSPER area in DTDSHDR is ON.
objectprefix/dsname/DTPVOLDnn/desc/EXTENTS	Data object. This object contains the data that is found within the extents for the source data set on a per volume basis: ► desc = descriptor ► NVSM = NONVSAM ► DATA = VSAM Data Component ► INDX = VSAM Index Component
objectprefix/dsname/APPMETA	Application metadata object that is provided by application in EIOPTION31 and provided to application in EIOPTION32.

After the DSS metadata objects are stored, the data objects are stored by using DS8880 transparent cloud tiering. The data object consists of the extents of the data set that are on the source volume. This process is repeated for every source volume where the data set is stored.

After all volumes for a data set are processed (where DSS successfully stored all the necessary metadata and data objects), DSS stores the application metadata object. DFSMSdss supports one application metadata object for each data set that is backed up.

Because data movement is offloaded to the DS8880, a data set cannot be manipulated as it is dumped or restored. For example, DFSMSdss cannot do validation processing for indexed VSAM data sets, compress a PDS on RESTORE, nor reblock data sets while it is being dumped.

At the time of this writing, compression and encryption are not done on DS8000 during data migration. Your data might be compressed or encrypted when it is allocated on the DS8000. Such data is offloaded to cloud in its original condition: compressed or encrypted. (Compression or encryption is typically done by zEDC or pervasive encryption.)

If you attempt to create an object prefix that exists within a container, DFSMSdss fails the DUMP to prevent the data from being overwritten.

Note: At the time of this writing, there are some considerations required when implementing transparent cloud tiering on systems with mirroring or copy services. For more information about this topic, refer to 3.3, “Data replication and copy services with transparent cloud tiering” on page 25.

3.2.1 Disaster Recovery considerations

Having a working disaster recovery solution is vital to maintaining the highest levels of system availability. These solutions can range from the simplest volume dump to tape and tape movement management, to high availability multi-target PPRC and IBM HyperSwap® solutions. The use of transparent cloud tiering, and storing data in cloud storage might affect your disaster recovery plan, and needs to be carried out during implementation.

Some of the steps required to recover your migrated data after a disaster include, but are not limited to:

- Network connectivity

  Make sure your disaster recovery has network access to the cloud environment. This might include configuring proxy, firewall, and other network settings to secure your connection.

- Cloud configuration

  Your disaster recovery DS8000 must be configured with the information necessary to access the cloud storage, including certificates to allow SSL connections. You might also have to set up your z/OS to connect to the cloud environment, depending on your configuration.

- User ID administration

  You might also need to create the userid and password on your disaster recovery DS8000 if you use S3 or IBM COS clouds. Update your z/OS to connect to the new DS8000, and have the user id defined in your storage.

- Bandwidth

  Keep in mind that during a disaster, a large amount of data set recalls might be requested, such as migrated image copies, and other data sets used only for disaster recovery (DR) purposes. If these data sets are stored in the cloud, make sure to have enough bandwidth available in your recovery site to avoid recovery delays related to network issues.

Note: The use of replication and copy services can affect the implementation of the transparent cloud tiering and the migration/recall of data. For more information about this topic, refer to 3.2.2, “Migration considerations”.

3.2.2 Migration considerations

The use of DFSMShsm to migrate and recall your data from the storage cloud can significantly reduce the number of CPU cycles required to manage the lifecycle of your data.

The following migration considerations can guide clients who are looking towards implementing Transparent Cloud Tiering and configuring the DFSMShsm to use it:

- Already migrated data

  After you configure the DFSMShsm, you might want to move some of your data from other migration tiers, such as disk ML1, or tapes ML2. You need to keep in mind that currently there is no command or automated process to move migrated data directly to the cloud.

  If you want to move data already migrated to the cloud, you need to first recall your data, and then migrate it to the cloud, again.

- Recalling data from the cloud

  As you expand your environment to use cloud services, it is likely that more data will be migrated and recalled from the cloud. When you request a recall for a data set, the ACS

routines are called, and the volume selection is done. Only cloud-capable volumes can be selected as target volumes. If no volumes in such condition exists, the recall fails.

Note: If you have more than one DS8000 attached to your system, make sure both have access to migrate and recall data from the cloud.

3.3 Data replication and copy services with transparent cloud tiering

You can use different technologies to provide data replication and copy services. The solutions can include flashcopying volumes, synchronous mirroring, asynchronous mirroring, or any combination of them. The solutions that you deploy on your system affect the implementation of transparent cloud tiering.

More information about the transparent cloud tiering compatibility is provided next:

- Two-site Metro Mirror

  The transparent cloud tiering is capable of handling data that is part of a two-site Metro Mirror relationship, with or without HyperSwap capability.

- IBM FlashCopy®

  Volumes with FlashCopy relationships are currently not eligible for transparent cloud tiering. Any FlashCopy relationship must be stopped before the volume is eligible for use by transparent cloud tiering. Future code updates are planned to remove this constraint.

- Extended Remote Copy (XRC) and Global Mirror (GM) sessions

  At the time of the writing of this book, you cannot use transparent cloud tiering on active XRC and GM sessions. Future releases are planned to enable the use of transparent cloud tiering on systems that use XRC and GM.

- Multi-Target Peer-to-Peer Remote Copy (PPRC) and Metro Global Mirror (MGM)

  Transparent cloud tiering currently does not support the use of multiple site replication. Future code updates are planned to remove this constraint.

3.4 Storage cloud communication

The communication between the mainframe and the cloud should be done by using a Representational State Transfer RESTful interface. REST is a lightweight, scalable protocol that uses the HTTP standard. A toolkit was included by using an APAR and is available on z/OS V2R1 and above. It supports secure, HTTPS communication between endpoints by using the Secure Sockets Layer (SSL) protocol or the Transport Layer Security (TLS) protocol.

The z/OS Web Enablement Toolkit provides the following sets of APIs that are necessary to establish communication:

- HTTP/HTTPS protocol enabler
- Java Script Object Notation (JSON) parser

Note: JSON is an open standard for exchanging data between two endpoints. It supports only a few simple data types and organizes the data in a set of {name,value} pairs.

The JSON parser is a set of APIs that allow applications on z/OS to search for specific names and get the corresponding values from the output that is returned by a server.

The HTTP/HTTPS protocol enabler allows applications on z/OS to connect to a server, build an HTTP request, submit the request, receive a response, and disconnect from a server. Multiple requests can be sent while a connection to a server is active.

The responses from a web server can be returned in different formats. The application that issues the requests specifies the format that is required for the return responses.

An HTTP response can be returned as plain text, which is considered an unstructured string of bytes. It also can be returned in a more structured format that is called JSON. The toolkit supplies a set of APIs to handle the structured data that is returned as a response to an HTTP request. This set is called the JSON parser.

DFSMS then uses the z/OS Web Enablement Toolkit to communicate directly with an Object Storage server to do the following tasks:

- Create containers: An application can create containers in an account by using the PUT method.
- Delete containers: An application can delete containers from an account by using the DELETE method.
- List objects in a container: An application can list objects within a particular container by using the GET method.
- Store objects: An application can store data as an object in an object store by using the PUT method.
- Retrieve objects: An application can retrieve an object from an object store by using the GET method.
- Delete objects: An application can delete objects from a container using the DELETE method.

The Web Enablement Toolkit is used to create, list, and delete containers and store and retrieve metadata. Transparent Cloud Tiering is responsible for storing and retrieving data from the cloud.

Figure 3-3 shows the relationship between the DFSMS, Web Enablement Toolkit, Transparent Cloud Tiering, and the cloud.

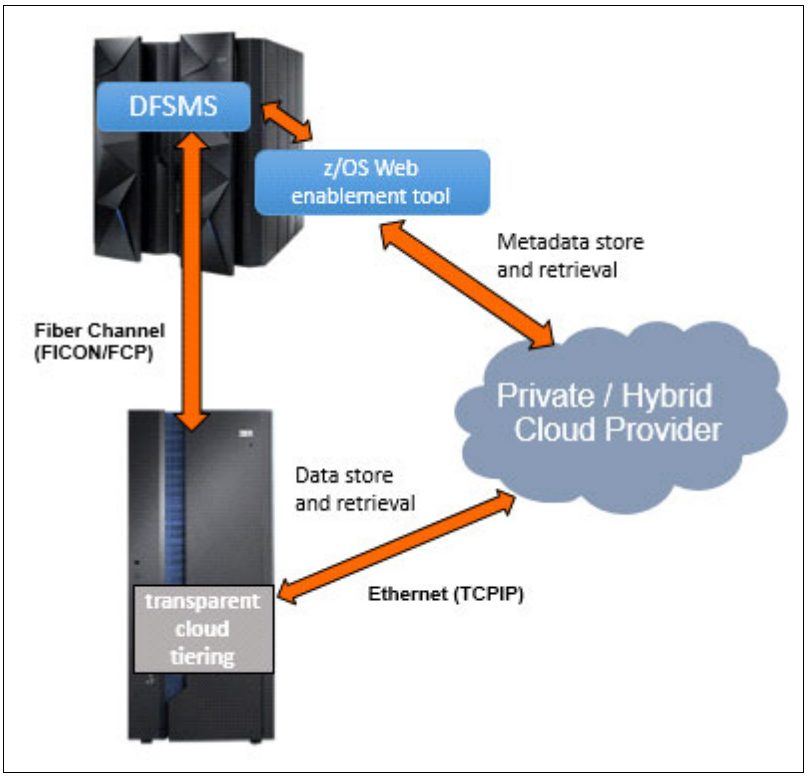


Figure 3-3 Cloud communication paths

The security in communication between the mainframe and the cloud is provided by using a user ID and password combination. Although the password is not included in DFSMS Cloud constructs, it is used by DFSMShsm and DFSMSdss to do store and recovery tasks from the cloud.

DFSMShsm stores an encrypted version of the password in its control data sets (CDSs) for use when migrating and recalling data. For manual DUMP and RESTORE operations, DFSMSdss requires the user ID and password to be included on JCL.

Note: Any users with access to the user ID and password to the cloud have full access to the data from z/OS or other systems perspective. Ensure that only authorized and required personnel can access this information.

For more information about security and user ID and password administration for DFSMShsm, see Chapter 8, “Using automatic migration” on page 71.

3.5 Selecting data for storage cloud

When you decide to implement a storage cloud, you also must plan for who can use this cloud and the type of data that you want to store. Defining correct data to be offloaded to cloud gives you more on-premises storage to allocate to other critical data.

A described in this chapter, the cloud should be considered an auxiliary storage option within a z/OS system, meaning that no data that requires online or immediate access should be moved to the cloud. Also, only Simplex volumes are eligible to have their data sets moved to the cloud.

Because no Object Storage data is cataloged or automatically deleted (except for DFSMShsm-owned objects), it is suggested that you proceed with caution when deciding which users can dump and restore their data sets from cloud. If any users decide to use the cloud, they must manually do housekeeping and delete the storage objects and containers that are created by them.

The area of DFSMShsm operations is a principal use case for storage cloud, for these reasons:

- DFSMShsm maintains information on containers and objects in its control data sets.
- It can retrieve or expire in its control data sets.
- It can automatically retrieve and expire the objects that are associated with data sets that are migrated to cloud storage.

The latest APARs for auto-migration allow for deleting empty containers.

Part 2

Cloud setup and use

In this part, we show you how we set up a cloud and how we used the new functionality to communicate with the cloud and to send data and retrieve data from it.

This part includes the following chapters:

4

Requirements

In this chapter, we describe the requirements necessary before you run Transparent Cloud Tiering, including DS8880, network and z/OS environment.

4.1 What do I need before running TCT?

To be able to make use of Transparent Cloud Tiering, you need to check the following requirements:

- Ethernet connections on DS8880
- z/OS Levels
- DS8880 Release Level
- Cloud APIs support
- Authentication Information from Cloud Service Provider or Administrator
- TLS/SSL considerations

4.1.1 Ethernet connections on DS8880

Transparent Cloud Tiering services use up to two 1Gb ethernet ports in each of the DS8880 processors (also known as central processor complexes, CECs).

Ethernet connectivity is also required from the mainframe to either the DS8880 or the Object Storage cloud server, depending on the cloud type chosen, as discussed in the 3.1, “FICON and TCP/IP data movement” section of this book. The Ethernet card is physically located in location code P1-C10 or P1-C11 (depending on the model), with Ports T1 and T2 used for the Hardware Management Console (HMC) to processor node communication. Ports T3 and T4 (bottom two ports circled in the Figure 4-1 below) are empty and typically covered by a plastic port covering. Remove the plastic covering and insert the RJ45 cable into either or all available ports.

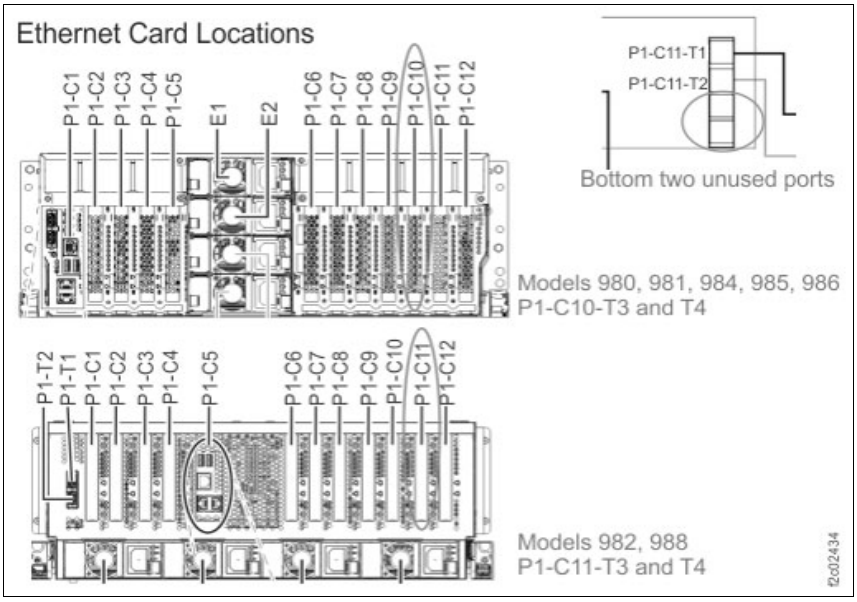


Figure 4-1 Ethernet ports to be used for TCT

4.1.2 z/OS Level

To set up cloud configuration you must have z/OS V2R1 with PTF/APAR OA51622, z/OS V2R2 with PTF/APAR OA50667, or higher.

The details of these APARs can be found at the following web pages:

http://www-01.ibm.com/support/docview.wss?uid=isg1OA51622

http://www-01.ibm.com/support/docview.wss?uid=isg1OA50667

DFSMShsm automatic migration support is available on z/OS V2R2 and above with PTF for APAR OA52913. Coexistence support is also available on z/OS V2R1 with PTF for APAR OA52913.

Use the IBM.Function.DFSMSCloudStorage fix category to identify PTFs associated with the DFSMS TCT support.

https://www-03.ibm.com/systems/z/os/zos/features/smpe/fix-category.html

You can also check the IBM Transparent Cloud Tiering for DS8880 Knowledge Center to verify if there are other specific Software requirements, at the link below:

https://www.ibm.com/support/knowledgecenter/en/ST5GLJ_8.3.0/com.ibm.storage.ssic.help.doc/f2c_requirements_trans_cloud_tiering.html

4.1.3 DS8880 Release Level

To set up the cloud configuration, DS8880 must have at least the release 8.2.3 – Bundle 88.23.19.0 Microcode and DSCLI.

To check your current DS8880 microcode level, issue the **lsserver -l** DSCLI command as shown in the example below:

```
dscli> lsserver -l
Date/Time: December 11, 2017 10:57:56 AM MST IBM DSCLI Version: 7.8.31.118 DS: -
ID Image ID Image Name     Power Control SFI State  LIC Version OS Version Bundle Version
==========================================================================================
00 1        SF75DMD30ESS01                0 online 7.8.31.118  7.1.4.402  88.31.41.0
01 1        SF75DMD30ESS11                0 online 7.8.31.118  7.1.4.402  88.31.41.0
dscli>
```

Figure 4-2 Display DS8880 Release on DSCLI

To display DS8880 release on DSGUI select Actions --> Properties as show Figure 4-3

Figure 4-3 Display DS8880 Release on DSGUI

4.1.4 Cloud APIs support

With the R8.2.3 of the DS8880 code, Transparent Cloud Tiering only supports the OpenStack Swift API to connect to object storage systems. R8.3 DS8880 code introduces support for S3 and IBM Cloud Object Storage in the S3 API.

- For the Swift API, z/OS communicates directly with the Cloud for exchanging metadata.
- For S3 and IBM Cloud Object Storage, all the data flows directly from the DS8880 to the Cloud.

Note: DFSMS does not natively support clouds that provide an S3 API. However, DFSMS can use the DS8880 to transparently tier data to an object storage cloud that provides a S3-compatible API. In Chapter 5, “Configuring the IBM DS8880 for TCT” on page 37 of this book, we will detail how to configure the DS8880 and DFSMS to enable transparent cloud tiering to an S3 cloud.

4.2 Authentication information

The following account information must be provided by your Cloud Service Provider or Administrator:

- Username
- Password
- Endpoint
- Tenant (if Swift)
- Port Number
- Certificates (if using SSL/TLS)

4.2.1 User name and Password

The cloud administrator supplies a user name and a password for each tenant. At the time of the writing of this book, a single user ID and password are used for all z/OS and DS8000 access to the cloud. For security purposes, the password is suppressed from the logs when configuring DFSMShsm to use the cloud, and it is stored in an encrypted form on DFSMShsm MCDS.

Note: Great care needs to be taken with this password. Because a single user ID and password are used, anyone with access to this password can access the cloud directly. This access gives the user the power to read, update, or delete the data in the cloud, potentially compromising data integrity, or making DFSMShsm unable to Recall or Restore the data from this cloud account. IBM suggests that a Security Administrator who is managing the Cloud Storage passwords be the individual who also manages the password for DFSMShsm, to protect this method of access to the cloud data sets.

4.2.2 Endpoint

The endpoint is the location or URL that the DS8880 and DFSMS use when authenticating with the Cloud object storage system.

When a swift-keystone authentication method is used, the endpoint must contain the version number of the identity API to use. As of the writing of this book, only the version 2 API is

supported. For example, if the provider endpoint is **`https://dallas.ibm.com`**, the endpoint should be configured as **`https://dallas.ibm.com/v2.0`**.

4.2.3 Tenant (if Swift)

This is the name or project name that identifies your object store environment. This name needs to be something meaningful to your organization's environment, for example, possible tenant names could be *production, development, test,* and so on. This name is usually provided by the cloud administrator.

4.2.4 Port Number

To have access to the endpoint connection you need to have the remote port number to which to connect, instead of the default HTTP or HTTPS port. The maximum length for the port number is five characters, ranging from 0 to 65535. You must also ensure that this port is open on the local network firewalls.

The port number is also provided by the cloud administrator. If using DS8880 as a proxy to connect to your cloud service provider, you should use port 8452 when defining your SMS cloud configuration.

4.2.5 Certificates (if using SSL/TLS)

The first level of encryption-based security provides secure communications between the DS8880 system, DFSMS, and the cloud service provider. The standard protocol, Transport Layer Security (TLS), protects these connections by encrypting authentication data that is transferred between DFSMS, DS8880 systems, and the cloud service provider. Secure communications are mandatory for these connections and require that public certificates are exchanged between the cloud service provider, DFSMS, and the DS8880 systems.

Note: SSL/TLS is only used to encrypt the authentication data between DFSMS, the DS8880, and the cloud object storage. Currently, data is not encrypted in flight. If you are using Pervasive Encryption to encrypt data sets from the host, that data remains encrypted.

For the swift-keystone cloud interface type that uses SSL/TLS to encrypt the authentication path, certificates are required to maintain a chain of trust between DFSMS, the DS8880, and the object store. If you use self-signed certificates, only the SysCA option is required. If you use a Certificate Authority (CA), the root CA and intermediate CA can be provided when configuring your cloud. These items point to a privacy-enhanced mail (PEM) file type that you can import into the DS8880 system and DFSMS. A PEM file format supports multiple digital certificates, including a certificate chain.

In the Chapter 5, "Configuring the IBM DS8880 for TCT" on page 37 and Chapter 6, "Configuring DFSMS" on page 45 of this book we demonstrate how you can configure the DS8880 system and DFSMS to use certificates for secure communications.

4.3 TLS/SSL considerations

DFSMS and IBM DS8880 will send account information (user names and passwords) over HTTP connection. To ensure that the information is encrypted, we highly recommend

establishing a secure HTTP connection between the z/OS host, the IBM DS8880 system, and the object storage cloud server.

The supported SSL/TLS versions to be used when making HTTP requests are: TLSV12, TLSV11, TLSV1, SSLV3.

There are two types of authentication:

- Server authentication: The z/OS host and/or DS8880 verifies the identity of the object storage cloud server.
- Mutual authentication: The z/OS host and/or DS8880 verifies the identity of the object storage cloud server, and the object storage cloud verifies the identity of the z/OS host and/or DS8880.

4.3.1 External CA versus self-signed certificates with DS8880 and DFSMS/RACF

Determining the type of certificate to use for secure communications sessions and the method to generate the certificate is challenging. Self-signed certificates and digital certificates issued by certificate authorities offer advantages and disadvantages.

The Table 4-1 compares the advantages and disadvantages of self-signed and CA-signed certificates:

Table 4-1 External certificate versus self-signed certificate

Type of Certificate	Advantages	Disadvantages
Self-signed certificate	No cost	Requires you to distribute your certificate, minus the private key, to each trading partner in a secure manner
	Easy to generate	Difficult to maintain; anytime the certificate is changed, it must be distributed to all clients
	Self-validated	Not validated by a third-party entity
	Efficient for small number of trading partners	Inefficient for large number of trading partners
CA-signed certificate (External)	Eliminates having to send your certificate to each trading partner	Trading partners must download digital CA-signed certificate used to verify the digital signature of trading partner public keys.
	No changes are required on the trading partner's system if you recreate the CA digitally-signed certificate using the same CA	Must be purchased from third-party vendor

For the Swift API, swift-keystone cloud interface is used to encrypt authentication credentials and connect DFSMS and IBM DS8880 to a cloud storage target. The authentication is done using root or system certificates with either Secure Sockets Layer or Transport Layer Security, SSL/TLS. For S3 or IBM Cloud Object Storage, the DS8880 system will have the certificate files to communicate with the cloud storage target and will communicate with DFSMS through a REST API proxy that runs in the HMC of the DS8880. The configuration steps to allow this communication will be detailed in the next chapters of this book.

5

Configuring the IBM DS8880 for TCT

This chapter describes how to configure the IBM DS8880 to support Transparent Cloud Tiering.

5.1 Configuring the IBM DS8880 for TCT

To access the cloud services, your DS8880 hardware must be configured to communicate with the cloud by using Ethernet connections. This configuration includes defining the following the components:

- Ethernet configuration
- Cloud configuration
- Configuring DS8880 User for REST API Proxy

The DS8880 features four Ethernet ports in each Central Electronic Complex (CEC), where two ports can be used to connect to the cloud services and the remaining two are used for the Hardware Management Console (HMC). You can connect one or two Ethernet cables to each CEC to provide a failover option and increase the overall bandwidth that is available for use. If you plan to use Swift cloud interface, the host will be responsible for storing metadata in the cloud, and will also require an Ethernet connection to the cloud server.

5.1.1 Ethernet configuration

When configuring your DS8880, you first must set up the Ethernet port configuration. This configuration is required to enable network connectivity. After you connect your Ethernet cables to the hardware, you can use the **lsnetworkport** command from the DS8880 Command Line Interface DSCLI to display any current Ethernet configuration. Example 5-1 shows a sample output from the **lsnetworkport** command.

Example 5-1 Output from lsnetworkport command

```
Date/Time: November 29, 2017 10:22:04 AM MST IBM DSCLI Version: 7.8.31.118 DS:
IBM.2107-75CFY71
ID      IP address  Subnet Mask  Gateway Primary DNS    Secondary DNS   State
I9813   0.0.0.0     0.0.0.0      0.0.0.0  0.0.0.0       0.0.0.0         Offline
I9814   0.0.0.0     0.0.0.0      0.0.0.0  0.0.0.0       0.0.0.0         Offline
I9B13   0.0.0.0     0.0.0.0      0.0.0.0  0.0.0.0       0.0.0.0         Offline
I9B14   0.0.0.0     0.0.0.0      0.0.0.0  0.0.0.0       0.0.0.0         Offline
```

Note: There is no way to delete a network port at this time. You can only clear the IP address or just leave it.

Ensure that you have the correct IP addresses, subnet mask, and DNS information available while you are configuring your hardware. Then, use the **setnetworkport** command to define the network settings, as shown on Example 5-2.

Example 5-2 Defining the network settings

```
setnetworkport -ipaddr 10.0.1.2 -subnet 255.255.255.0 I9814
setnetworkport -ipaddr 10.0.1.3 -subnet 255.255.255.0 I9B14
```

After the configuration is complete, issue a new **lsnetworkport** command to confirm that the network was properly configured. Example 5-3 shows the new network configuration.

Example 5-3 Verifying the network configuration

```
Date/Time: November 29, 2017 10:26:20 AM MST IBM DSCLI Version: 7.8.31.118 DS:
IBM.2107-75CFY71
ID      IP address  Subnet Mask   Gateway   Primary DNS    Secondary DNS   State
```

```
I9813   0.0.0.0     0.0.0.0       0.0.0.0   9.0.000.10     0.0.0.0         Offline
I9814   10.0.1.2    255.255.255.0 0.0.0.0   9.0.000.10     0.0.0.0         Online
I9B13   0.0.0.0     0.0.0.0       0.0.0.0   9.0.000.10     0.0.0.0         Offline
I9B14   10.0.1.3    255.255.255.0 0.0.0.0   9.0.000.10     0.0.0.0         Online
```

With the Ethernet configuration complete, you can proceed to the cloud configuration process.

5.1.2 Cloud configuration

Now that your network is properly configured and functional, you can configure the access to the cloud storage. When you are defining your cloud settings, you should have all cloud-related information that is required to set up the connection, including the cloud endpoint, tenant (if applicable), user ID, and password.

Use the **mkcloudserver** command to define a new cloud to your hardware. The following parameters must be supplied when the **mkcloudserver** command is used:

- Endpoint

  A URL that describes the place to authenticate to the Object Storage system.

- Type

  Transparent Cloud Tiering supports the following types of Object Storage protocols and authentication mechanisms:

  - swift: This keyword describes a decrypted (HTTP) communication path.
  - swift-keystone: This keyword uses SSL/TLS to encrypt authentication credentials across the TCP/IP network.
  - ibmcos: Specifies IBM Cloud Object Storage (COS) for data protection through backup and recovery.
  - aws-s3: Specifies that Amazon Simple Storage Service (Amazon S3) is to allow the DS8000 system to authenticate and connect to S3 storage.

  Note: It is not recommended that you use swift to authenticate to an object store because user name and password credentials are sent in the clear form.

- Tenant (typically only for the OpenStack Swift cloud type)

  The cloud administrator supplies a tenant name that describes the tenant of the object store because Object Storage is traditionally a multi-tenant system.

- User name and password

  The cloud administrator supplies a user name and a password for each tenant.

- RootCA, IntermCA, SysCA

  For the swift-keystone type that uses SSL/TLS to encrypt the authentication path, certificates are required to maintain a chain of trust between the DS8880 and the object store. If you use self-signed certificates, only the SysCA option is required. If you use a certificate authority (CA), the root CA and intermediate CA can be provided in the **mkcloudserver** command. These items point to a PEM file type that you can import into the DS8880 system.

Note: The full path from where you run the DSCLI command must point to the PEM file to be imported.

- Cloud Name

  You can provide a unique name to describe the object store that is being configured. The cloud name that is used in the hardware configuration should be the same that is used when the cloud is defined on DFSMS.

- Location

  The location parameter specifies the location of the cloud server and is only valid with types ibmcos or aws-s3.

When you run the **mkcloudserver** command, the ability of the DS8880 and the object store to communicate is verified. Running the command also verifies that the data path is accessible and encryption certificates are valid.

Configuring a Swift cloud server

Example 5-4 shows a sample **mkcloudserver** command that was issued from the DSCLI interface to configure a Swift cloud server.

Example 5-4 Configuring a Swift cloud server

```
dscli> mkcloudserver -type swift-keystone -tenant tenant -username username
-pw password -endpoint https://ibmcloud.ibm.com:5000/v2.0/ -rootcaloc
/home/ssl_cacert.pem -intermcaloc /home/user/ssl_cacert.pem -syscaloc
/home/user/ssl_cert.pem ibmcloud
```

Configuring an IBM COS or S3 cloud server

Example 5-5 shows a sample **mkcloudserver** command that was issued from the DSCLI interface to configure an IBM COS cloud server.

Example 5-5 Configuring an IBM COS cloud server

```
dscli> mkcloudserver –type ibmcos -username ITSOUser –pw password –endpoint –nossl
–endpoint http://ibmcloud.internal.ibm.com/ -loc tucson ibmcloudname
```

Listing a cloud server configuration

You can also list the cloud configuration on your hardware. Use the **lscloudserver** command to list cloud information. Sensitive information, such as user ID and password are not displayed in the output. The Example 5-6 displays a sample output from the **lscloudserver** command.

Example 5-6 Listing cloud information

```
dscli> lscloudserver

Date/Time: November 30, 2017 2:24:31 PM MST IBM DSCLI Version: 7.8.31.118 DS: -
name       node type            tenant   endpoint
ibmcloud   0 swift-keystone     test     https://ibmcloud.ibm.com:5000/v2.0/
ibmcloud   1 swift-keystone     test     https://ibmcloud.ibm.com:5000/v2.0/
```

Updating or removing a cloud server configuration

If you need to update any cloud settings, the existing configuration must be first deleted. Then, the new configuration can be defined. Use the **rmcloudserver** command to remove the existing cloud configuration. When the command is issued, a prompt message will be displayed requesting a confirmation if the cloud server configuration is to be removed. If you want to continue with the removal enter **y**, as shown in the Example 5-7:

Example 5-7 Removing a cloud server configuration

```
dscli> rmcloudserver ibmcloud

Are you sure you want to delete cloud server ibmCloud? [y/n]:y
The cloud server ibmcloud successfully deleted.
```

After deleting the old configuration, you can add a new configuration using the **mkcloudserver** command.

Note: At the time of this writing, only a single cloud can be configured at a specific time. If a new cloud must be configured, the current setting must be deleted before the new cloud definition is used, using the **rmcloudserver** command.

5.1.3 Configuring DS8880 User for REST API Proxy

As mentioned earlier in this book, starting with the R8.3 DS8880 code level, it is possible to use the DS8880 system as a proxy for the mainframe. This capability enables support for S3 and IBM COS cloud interfaces for Transparent Cloud Tiering.

To allow this, a user ID needs to be created in the DS8880 that will communicate with the cloud servers, and this user ID will be used by the DFSMShsm to authenticate on the DS8880 system using REST API interface. The REST API proxy service is automatically enabled when the DS8880 Code level is upgraded to R8.3 or higher. No other tasks are required to enable the communication other than configuring the network and the user ID that will be used by DFSMShsm.

The Example 5-8 shows how to create a local user ID in the DS8880 using DSCLI:

Example 5-8 Creating a DS8880 user ID for DFSMShsm to connect

```
dscli> mkuser -pw REDBOOKS -group monitor itsouser

Date/Time: December 13, 2017 8:56:41 AM MST IBM DSCLI Version: 7.8.31.118 DS: -

CMUC00133I mkuser: User itsouser successfully created.
```

When a user ID is created on the DS8880, the initial password used in the **mkuser** command is temporary and expired. It will need to be changed during the first logon by the security administrator, either through the DS Storage Manager GUI or DSCLI, to the final password that will later be used by DFSMShsm. After the user ID is created and the password is changed, you will be able to connect the DFSMShsm to the DS8880 using the steps described in the next chapter of this book.

Attention: The user ID created for the DFSMShsm to connect to the DS8880 will follow the security rules defined in the Authentication Policy of the DS8880. It means that, depending on your policy rules, this password can expire after a certain number of days. To avoid connectivity issues with TCT, change the password of this user ID both in the DS8880 and the DFSMShsm before the expiration date or modify the expiration date policy to never expire.

Configuring LDAP authentication

Optionally, you can use LDAP to create the user ID that will be used by DFSMShsm to connect to the DS8880 REST API Proxy interface. Starting with the DS8880 code R8.1, the Hardware Management Console (HMC) of the DS8880 includes the IBM Copy Services Manager (CSM) software preinstalled on it. CSM as installed on the HMC, or acquired separately, includes a lightweight build of the IBM WebSphere® Liberty server code, used to authenticate CSM users through a Lightweight Directory Access Protocol (LDAP). The same integrated LDAP support can be used for remote authentication of DS8000 users. Furthermore, if you want to take advantage of only the CSM LDAP client for DS8000 LDAP authentication, the CSM license and CSM activation are not required.

You can configure the DS8880 so that both local and LDAP authentication methods can be used concurrently. The repositories (user registry) for local users and LDAP users are different.

Important: After you configure the LDAP authentication, users from the local user registry are still able to authenticate. We suggest that at least one user from the local user registry retain the administrator role. This user serves as a backup if a loss in communication to the LDAP servers occurs.

The step-by-step instructions to configure LDAP on the DS8880 system can be found in the *IBM DS8880 Integrated Copy Services Manager and LDAP Client on the HMC*, REDP-5356 publication.

External CA versus self-signed certificates and REST API Proxy

The 4.2.5, “Certificates (if using SSL/TLS)” on page 35 section of this book describes how to secure the communication between the DS8880 system, DFSMS and the cloud service provider. You can use encryption-based security with certificates that are exchanged during the authentication process, which can be External CA (signed by a third-party Certificate Authority) or self-signed certificates. Also, if using S3 or IBMCOS cloud types, DS8880 uses a REST API Proxy interface to communicate with DFSMShsm. To do so, DFSMShsm connects to the DS8880 using its HTTPS interface, so the certificate used by the DS8880 must be added to the IBM RACF® for the DFSMShsm authentication to be successful.

We will demonstrate how to configure both the self-signed and External CA certificate options in the DS8880 next. The steps required to configure RACF with the certificates are described in the Chapter 6, “Configuring DFSMS” on page 45.

Note: To work with certificate files on the DS8880 you will need a user ID with Administrator role in the DS8880.

Creating a self-signed certificate on the DS8880

To create a self-signed certificate on the DS8880, you may use the DS Storage Manager GUI, by following the steps below:

1. Log on to the DS Storage Manager GUI.
2. Select **Settings** > **Security.**
3. Access the “Communications Certificate” tab.
4. Click **Create Self-signed Certificates**.
5. Enter the information requested regarding your organization.
6. Click **Create.** A warning message is displayed stating that the HMC will be rebooted and any users connected will be automatically logged off.
7. Click **Yes** to continue with certificate creation. After creation, the certificate is automatically loaded at the HMC.

This self-signed certificate must be uploaded to the z/OS host that will connect to the DS8880. First, download the certificate to your workstation using an *openssl* command, as shown in the Example 5-9. (In the example, **“**`DS8000-HMC-IP`**”** is the IP address of the DS8880 HMC that is configured as the REST API Proxy server):

Example 5-9 Downloading the DS8880 HMC self-signed certificate

```
openssl x509 -in <(openssl s_client -connect DS8000-HMC-IP:8452 -prexit
2>/dev/null) -text -out certificate.pem
```

Note: Certain operating systems support the openssl command natively. Others may require a client to be installed to support the openssl command.

The procedure to upload the certificate from your workstation to the z/OS host is demonstrated in the section 6.1.1, “Uploading the Certificate files to the z/OS host” on page 46 of this book.

Creating and using an External CA certificate on the DS8880

To create an External CA certificate on the DS8880, you may also use the DS Storage Manager GUI to create a Certificate Signing Request (CSR) file that will be signed by the third-party Certificate Authority in the creation of the CA certificate, using the following steps:

1. Log on to the DS Storage Manager GUI.
2. Select **Settings** > **Security**.
3. Access the “Communications Certificate” tab.
4. Click **Create Certificate Signing Requests**.
5. Enter the HMC DNS host name and the information regarding your organization.
6. Click **Create**. The CSR file is created, and you receive a Web-browser download window to specify the destination path to be used to save the file in your computer.
7. Save the file, and send it to the Certificate Authority to generate the certificate.

After the Certificate Authority generates and sends you the final certificate file, you must import it to the DS8880. You can do this through the DS Storage Manager GUI, using the steps below:

1. Log on to the DS Storage Manager GUI.
2. Select **Settings** > **Security**.
3. Access the “Communications Certificate” tab.
4. Click **Import Existing Certificates**. A message window prompts you to select the certificate file from your local computer.
5. Navigate to the folder in your computer where the file is stored, select the certificate file, and click **Import**. A warning message states that the HMC will be rebooted and any users connected will be automatically logged off.
6. Click **Yes** to import and load the certificate at the HMC.

6

Configuring DFSMS

This chapter describes how to configure the z/OS DFSMS component to use Transparent Cloud Tiering. In this book, we used the Interactive Storage Management Facility (ISMF) ISPF interface to define the cloud to DFSMS.

6.1 Adding digital certificates to RACF

In this section, we will show all the steps required to add the certificate files to RACF if using secure communication methods for TCT.

6.1.1 Uploading the Certificate files to the z/OS host

Take these steps to prepare to send the certificate files to the host:

1. Decide which of the following certificate types you will use:
   – External certificate (from a third-party Certificate Authority) or
   – Self-signed certificate
2. Receive this certificate from the certificate administrator.

To upload the certificate files to your z/OS host, you must meet the following conditions:

- Certificate files are to be uploaded to z/OS as RECFM=VB
- Must be cataloged
- Cannot be a PDS or PDS member

Example 6-1 shows how you can upload the certificate files to a z/OS host:

Example 6-1 Uploading certificates to the z/OS host

```
ITSOUser-MBP:ssl_certs itsouser$ ftp my.zoshost.com
Connected to my.zoshost.com.
220-FTP1 IBM FTP CS V2R3 at my.zoshost.com, 17:47:54 on 2017-08-06.
220 Connection will close if idle for more than 5 minutes.
Name (my.zoshost.com:workstation_user): zosuser
331 Send password please.
Password:
230 IBMUSER is logged on.  Working directory is "IBMUSER.".
Remote system type is MVS.
ftp> site RECFM=VB
200 SITE command was accepted

ftp> put carootcert.pem INTROOT.PEM
local: carootcert.pem remote: INTROOT.PEM
229 Entering Extended Passive Mode (|||1037|)
125 Storing data set IBMUSER.INTROOT.PEM
100% |***********************************|  1434      606.22 KiB/s    --:-- ETA
250 Transfer completed successfully.
1434 bytes sent in 00:00 (11.65 KiB/s)

ftp> put caintermediatecert.pem INTRMED.PEM
local: caintermediatecert.pem remote: INTRMED.PEM
229 Entering Extended Passive Mode (|||1038|)
125 Storing data set IBMUSER.INTRMED.PEM
100% |***********************************|  1838      787.24 KiB/s    --:-- ETA
250 Transfer completed successfully.
1838 bytes sent in 00:00 (14.63 KiB/s)
ftp>
```

6.1.2 Adding External CA certificates to RACF

After sending the certificate files to the host you must add them to RACF. This is done using the RACDCERT RACF command. For the user to be able to run this command, there are also a few other security requirements to be satisfied:

- The RACDCERT command needs to be authorized under the AUTHCMD list in IKJTSOxx parmlib member.
- The user ID that will issue the RACDCERT command also needs to be authorized on RACF to do so.

You may check the details of the RACF authorization requirements for the RACDCERT command in the IBM Knowledge Center page, at the link:

https://www.ibm.com/support/knowledgecenter/en/SSLTBW_2.1.0/com.ibm.zos.v2r1.icha700/cracd.htm

The Example 6-2 below shows how you can add the certificate data set to RACF, using the RACDCERT command:

Example 6-2 Adding a certificate data set to RACF

```
RACDCERT CERTAUTH ADD(<certificate dataset>) WITHLABEL('Cloud Certificate') TRUST
```

Also, after adding the certificate file to the RACF, the DIGTCERT class needs to be refreshed for the new configuration to take effect, as shown in the Example 6-3:

Example 6-3 Refreshing the DIGTCERT class

```
SETROPTS RACLIST (DIGTCERT) REFRESH
```

When you use an external CA, the certificate might be signed by either a root CA or an intermediate CA.

If the certificate was signed by a root CA, you will only need to add the cloud Root CA certificate data set to the RACF, as shown in the Example 6-4.

Example 6-4 Adding only the Root CA certificate to RACF

```
RACDCERT CERTAUTH ADD('IBMUSER.INTROOT.PEM') WITHLABEL('Cloud Root CA') TRUST
SETROPTS RACLIST (DIGTCERT) REFRESH
```

If the certificate was signed by an intermediate CA, you must add both the root CA certificate and the intermediate CA certificate data sets to RACF, as demonstrated in the Example 6-5:

Example 6-5 Adding Root CA and Intermediate CA certificates on RACF

```
RACDCERT CERTAUTH ADD('IBMUSER.INTROOT.PEM') WITHLABEL('Cloud Root CA') TRUST
RACDCERT CERTAUTH ADD('IBMUSER.INTRMED.PEM') WITHLABEL('Cloud Intermediate CA')
SETROPTS RACLIST (DIGTCERT) REFRESH
```

6.1.3 Adding self-signed certificates to RACF

To add a self-signed certificate data set to RACF, you must use the SITE option of the RACDCERT RACF command, as demonstrated in the Example 6-6.

Example 6-6 Adding site self signed certificates on RACF

```
RACDCERT SITE ADD('IBMUSER.INTROOT.PEM') WITHLABEL('Self-Signed-Cert') TRUST
SETROPTS RACLIST (DIGTCERT) REFRESH
```

6.2 Creating a Cloud Construct using ISMF

To support the use of storage clouds, the new DFSMS cloud construct was added to ISMF panels. This new construct allows you to define the parameters that are necessary to connect to the clouds. The new Cloud option on ISMF menu is accessible only when you have access to the administrator mode on ISMF panels.

Example 6-7 shows the new Cloud option that is available from main ISMF menu panel. Depending on your terminal configuration, it might be necessary to scroll down to view this new option.

Example 6-7 New Cloud option on ISMF panel

```
                ISMF PRIMARY OPTION MENU - z/OS DFSMS V2 R2

                                                                More:   -
 4  Data Class                - Specify Data Set Allocation Parameters
 5  Storage Class             - Specify Data Set Performance and Availability
 6  Storage Group             - Specify Volume Names and Free Space Thresholds
 7  Automatic Class Selection - Specify ACS Routines and Test Criteria
 8  Control Data Set          - Specify System Names and Default Criteria
 9  Aggregate Group           - Specify Data Set Recovery Parameters
 10 Library Management        - Specify Library and Drive Configurations
 11 Enhanced ACS Management   - Perform Enhanced Test/Configuration Management
 C  Data Collection           - Process Data Collection Function
 G  Report Generation         - Create Storage Management Reports
 L  List                      - Perform Functions Against Saved ISMF Lists
 P  Copy Pool                 - Specify Pool Storage Groups for Copies
 R  Removable Media Manager   - Perform Functions Against Removable Media
 S  Cloud
```

Select the **S** option (Cloud) to open the Cloud Application Selection panel (see Example 6-7). In this panel, you can list, display, define, or alter a cloud. As shown in Example 6-8 on page 48, we are defining (option **3**) a cloud that is named IBMREDBOOKS:

Example 6-8 Defining the IBMREDBOOKS cloud

```
                    CLOUD APPLICATION SELECTION

To perform Cloud Operations, Specify:
  CDS Name  . . . . . . . 'TCTRBOOK.SCDS'
                          (1 to 44 character data set name or 'Active' )
  Cloud Name  . . . . . . IBMREDBOOKS                    (For Cloud List,
                          fully or partially specified or * for all)

Select one of the following options:
  3  1. List    - Generate a list of Clouds
     2. Display - Display a Cloud
     3. Define  - Define a Cloud
     4. Alter   - Alter a Cloud
```

```
If List Option is chosen,
   Enter "/" to select option      Respecify View Criteria
                                   Respecify Sort Criteria

Command ===>
```

The cloud definition process consists of two panels. The first panel is shown in Example 6-9:

Example 6-9 First cloud definition panel

```
                                CLOUD DEFINE                  Page 1 of 2

SCDS Name  . . : TCTRBOOK.SCDS
Cloud Name . . : IBMREDBOOKS

To DEFINE Cloud, Specify:

Description   IBM REDBOOK DEMO CLOUD


Provider  . . SWIFT-KEYSTONE    (SWIFT or SWIFT-KEYSTONE)

Identity  . . test:tester




Command ===>
```

The following fields are available for definition in the first cloud definition panel:

- SCDS Name: The name of the SCDS where the cloud construct is stored.
- Cloud Name: The cloud name that is used by DFSMShsm and DFSMSdss when communicating with the cloud. The same cloud name must be defined on the DS8000.
- Description: A brief description of the cloud you are defining. You can include some information about the service provider, service expiration date, or availability. Up to 120 characters can be used in description.
- Provider: Specifies the type of cloud provider. At the time of this writing, only SWIFT and SWIFT-KEYSTONE options are available. For S3 and IBMCOS cloud providers, DFSMS will communicate with the DS8880 using the REST API Proxy interface, which uses SWIFT for this communication.
- Identity: Specifies the credentials that are used when authenticating with the cloud. Up to 256 characters can be used (upper and lowercase, numbers, and “@”, “#”, “$”, “_”, “:” special characters).

Move to the second definition panel by using the **DOWN** command. The second panel is shown in Example 6-10.

Example 6-10 Second cloud definition panel

```
                                CLOUD DEFINE                  Page 2 of 2

SCDS Name  . . : TCTRBOOK.SCDS
```

```
Cloud Name . . : IBMREDBOOKS

To DEFINE Cloud, Specify:

Endpoint . . . . https://ibmredbooks.demo.ibm.com




Port Number  . . 5000      (0 to 65535)
SSL Version  . . TLSV12    (TLSV12, TLSV11, TLSV1, SSLV3 or blank)
SSL Key  . . . . *AUTH*/*



Command ===>
```

The following fields are available for definition in the second cloud definition panel:

- Endpoint: Identifies the Uniform Resource Identifier (URI) that is used when authenticating with the cloud. Up to 256 characters can be used (upper and lowercase, numbers, and several special characters).
- Port Number: Specifies the remote port number to which to connect. Possible values are 0 - 65535. If connecting to the DS8880 HMC REST API Proxy, you must use port 8452.
- SSL Version: Defines the lowest acceptable SSL version that is used when connecting to the cloud.
- SSL Key: Specifies the name of the keystore to be used. DFSMS supports the use of SAF key ring name or a PKCS #11 token. If you will use CA certificates, you must specify *AUTH*/*. If you will use a self-signed certificate, you must specify *SITE*/*.

After the cloud configuration is completed and saved, the SCDS that contains the cloud definition must be activated.

Activating the new configuration does not automatically connect z/OS to the cloud. Each application that is trying to access the cloud is required to provide the password to store and retrieve data. The DS8880 must also be configured to access the cloud before the connection can be established.

To activate the SCDS, go to main ISMF menu and select option **8 Control Data Set**, as shown in Example 6-11.

Example 6-11 Selecting the Control Data Set option

```
                ISMF PRIMARY OPTION MENU - z/OS DFSMS V2 R2

                                                                More:     +
 0  ISMF Profile             - Specify ISMF User Profile
 1  Data Set                 - Perform Functions Against Data Sets
 2  Volume                   - Perform Functions Against Volumes
 3  Management Class         - Specify Data Set Backup and Migration Criteria
 4  Data Class               - Specify Data Set Allocation Parameters
 5  Storage Class            - Specify Data Set Performance and Availability
 6  Storage Group            - Specify Volume Names and Free Space Thresholds
 7  Automatic Class Selection - Specify ACS Routines and Test Criteria
 8  Control Data Set         - Specify System Names and Default Criteria
```

```
 9  Aggregate Group           - Specify Data Set Recovery Parameters
 10 Library Management        - Specify Library and Drive Configurations
 11 Enhanced ACS Management   - Perform Enhanced Test/Configuration Management
 C  Data Collection           - Process Data Collection Function
 G  Report Generation         - Create Storage Management Reports
 L  List                      - Perform Functions Against Saved ISMF Lists
 Selection or Command ===>
```

The CDS Application Selection panel is displayed, as shown in Example 6-12. Select option **5. Activate the CDS** to activate the configuration.

Example 6-12 Activate the CDS

```
                          CDS APPLICATION SELECTION

 To Perform Control Data Set Operations, Specify:
   CDS Name . . 'TCTRBOOK.SCDS'
                             (1 to 44 Character Data Set Name or 'Active')

 Select one of the following Options:
   5  1. Display       - Display the Base Configuration
      2. Define        - Define the Base Configuration
      3. Alter         - Alter the Base Configuration
      4. Validate      - Validate the SCDS
      5. Activate      - Activate the CDS
      6. Cache Display - Display CF Cache Structure Names for all CF Cache Sets
      7. Cache Update  - Define/Alter/Delete CF Cache Sets
      8. Lock Display  - Display CF Lock Structure Names for all CF Lock Sets
      9. Lock Update   - Define/Alter/Delete CF Lock Sets
 If CACHE Display is chosen, Enter CF Cache Set Name . . *
 If LOCK Display is chosen, Enter CF Lock Set Name . . . *
 Command ===>
```

Place a forward-slash in the **Confirm Activate Request** panel.

You can validate your Source Control Data Set (SCDS) by using option **4 Validate the SCDS** before you make it the active CDS.

An alternative way of activating the CDS is by using the **SETSMS SCDS**(*dsname*) command.

The DFSMShsm also needs permission to list the keyrings it has access to. It can be allowed by granting READ access to the user ID that is used by the DFSMShsm started task to the IRR.DIGTCERT.LIST and IRR.DIGTCERT.LISTRING profiles, as shown in the Example 6-13:

Example 6-13 Setting access to DSHSM proc on RACF

```
PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ID(DFHSM) ACCESS(READ)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID(DFHSM) ACCESS(READ)
SETROPTS RACLIST(FACILITY) REFRESH
```

6.3 Configuring the DFSMShsm environment

Before you can use storage cloud for migration, it is necessary to configure DFSMShsm to communicate to the cloud. We suggest that you complete the DFSMShsm cloud configuration

after the DS8880 configuration is complete, as described in the Chapter 5, “Configuring the IBM DS8880 for TCT” on page 37. Also, DFSMS cloud construct (see Chapter 6.2, “Creating a Cloud Construct using ISMF” on page 48) should be set up because DFSMShsm uses these structures to establish a cloud connection.

If the structures are not set up, the configuration commands fail. The result of the command to connect to a cloud that is not defined in the DFSMS constructs is shown in Example 6-14.

Example 6-14 DFSMShsm missing constructs messages

```
                              ISPF Command Shell
Enter TSO or Workstation commands below:

===> HSEND SETSYS CLOUD(NAME(IBMREDBOOKS) CCREDS)



Place cursor on choice and press enter to Retrieve command
=>
=>
=>
COMMAND REQUEST 00000074 SENT TO DFSMSHSM
ARC1584I SETSYS CLOUD - NAME IBMREDBOOKS NOT FOUND
ARC0100I SETSYS COMMAND COMPLETED
***
```

To configure DFSMShsm to use a cloud storage, you must issue a **SETSYS CLOUD** command from the TSO command screen. The command allows you to perform the following actions:

- Connect to a new cloud
- Refresh cloud credentials
- Delete a cloud from DFSMShsm CDSs

It is possible to have up to seven cloud definitions stored on DFSMShsm CDS. If it is needed to set up a new one after having seven cloud definitions already configured, you must first delete one definition before creating the new one. To identify the clouds currently configured to DFSMSMhsm, you can issue the command shown on Example 6-15, and search for configured clouds. The first cloud name starts on byte x’45C’:

Example 6-15 Displaying Cloud information

```
HSEND FIXCDS S MHCR DISPLAY


+0440  00000000 00000000 00000000 00000000 00000000 00000000 00000000 E2F3D7D9
 *                           IBMR*
+0460  D6E7E840 40404040 40404040 40404040 40404040 40404040 40400007 80000000
 *EDBOOKS                              *
+0480  B70D8B65 BDBAF129 841BDF9F AB0ACD01 4AD6B572 436A844E 9CBCE8C6 E8B3BDF5
 *      1           0        YFY  5*
+04A0  954F1E0E 266F6578 0EFDEF51 94584767 492957AF B0B93A62 1937EBF7 744CE463
 *                            7  U *
+04C0  B0EDCDD7 93854BB0 0B0109C5 62727A38 00000000 00000000 00000000 00000000
 *   P  .    E                        *
+04E0  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
```

The **SETSYS CLOUD** command that is shown in Example 6-16 defines the cloud to the DFSMShsm, attempts to connect, and if it's successful, stores the cloud related information into the MCDS.

Example 6-16 Defining the cloud to DFSMShsm

```
HSEND SETSYS CLOUD(NAME(IBMREDBOOKS) CLOUDCREDENTIALS)
```

When this command is issued, a WTOR prompts you to supply the cloud password. The message identifier that is related to the WTOR is ARC1585A. Example 6-17 shows the WTOR waiting for reply on system log.

Note: DFSMShsm activity is quiesced until the WTOR receives a reply.

Example 6-17 WTOR generated by SETSYS CLOUD command

```
 ARC0300I IBMUSER  ISSUED===>SETSYS CLOUD(NAME(IBMREDBOOKS) CCREDS)
*0029 ARC1585A ENTER PASSWORD FOR CLOUD IBMREDBOOKS
 R 29 SUPPRESSED
 IEE600I REPLY TO 0029 IS;SUPPRESSED
 ARC0100I SETSYS COMMAND COMPLETED
```

The cloud password can contain upper and lower case characters; therefore, we recommend that you reply to this WTOR from the system console or the SDSF System Command Extension as show Example 6-18. Replying from the SDSF SYSLOG forces a reply to upper case, which can result in a failed authentication.

Example 6-18 Case sensitive password

```
                      System Command Extension

===> 29,PaSswOrd
```

During the configuration, the connection to the cloud is tested. If it cannot be established, an error message is returned to the user with the information related to the connection error, as shown in Example 6-19.

Example 6-19 Failure to connect to the cloud message

```
ARC1581I UNEXPECTED HTTP STATUS 401 DURING A GET FOR URI
ARC1581I (CONT.) https://ibmredbooks.demo.ibm.com/auth/v1.0 ERRTEXT HTTP/1.1
ARC1581I (CONT.) 401 Unauthorized
ARC0100I SETSYS COMMAND COMPLETED
***
```

This message indicates that some of the authentication information entered is wrong, or that the password entered might be expired on the DS8880, as explained in the section 5.1.3, "Configuring DS8880 User for REST API Proxy" on page 41 of this book. You can use the **SETSYS CLOUD** command to refresh the cloud settings, change the password, or remove the cloud from DFSMShsm control records. The following sample **SETSYS CLOUD** commands are supported:

- **SETSYS CLOUD(NAME(xxxxx) REMOVE)**: Use this command to remove the cloud "xxxxx" from DFSMShsm control data sets.

- **SETSYS CLOUD(NAME(xxxxx) REFRESH)**: Use this command to refresh the cloud “xxxxx” credentials to DFSMShsm CDS, including the password that was used to connect to the cloud.
- **SETSYS CLOUD(NAME(xxxxx) PASSWORD)**: Use this command to create a WTOR requesting the cloud password. After the new password is suppressed, it is encrypted and stored on DFSMShsm CDS.

6.4 Controlling access to the Cloud features

At the time of this writing, a single user ID and password are used to connect to a cloud.

Note: Any user with access to the Cloud credentials has full access to all storage objects in the cloud, including the capability to update, move, or delete data.

We strongly suggest that you protect your cloud password and avoid the use of DFSMSdss jobs to move your data to cloud because any user with access to your JCL (from SDSF panels or your JCL libraries) can spot the cloud password.

Use DFSMShsm to manage the migration of your data to and from the cloud. DFSMShsm stores an encrypted version of the password in its Control Data Sets (CDSs), which users cannot access.

There are IBM RACF facility class profiles that are available to protect and control which users are allowed to use the **DUMP** and **RESTORE** commands, along with **CLOUD**, **CONTAINER**, or **OBJECTPREFIX** keywords. Only users with READ access to these profiles can use these commands.

Note: If the profiles are not defined, any user can use DFSMSdss to store data and retrieve data from a cloud if they know the cloud credentials.

6.4.1 Controlling access to DFSMSdss

Define SAF resources to control access to the CLOUD keyword on the DFSMSdss DUMP and RESTORE commands. Typically, the following FACILITY class profiles are defined with a universal access of NONE:

- STGADMIN.ADR.DUMP.CLOUD applies to logical dump
- STGADMIN.ADR.RESTORE.CLOUD applies to logical restore

The Example 6-20 shows a sample job that can be used to define these FACILITY class profiles on RACF:

Example 6-20 Define SAF resources to control access to the CLOUD keyword

```
//STEPR009 EXEC PGM=IKJEFT01
//SYSTSPRT DD  SYSOUT=*
//SYSTSIN  DD  *
   RDEFINE FACILITY STGADMIN.ADR.DUMP.CLOUD    UACC(NONE)
   RDEFINE FACILITY STGADMIN.ADR.RESTORE.CLOUD UACC(NONE)

   SETROPTS     RACLIST(FACILITY) REFRESH
/*
```

6.4.2 Controlling access to DFSMShsm

To control the access to DFSMShsm, you must set up the following tasks:

- Enable the CP Assist for Cryptographic Functions
- Define DFSMShsm to z/OS UNIX System Services
- Define SAF resources to control access to the CLOUD

Enable the CP Assist for Cryptographic function

Ensure that the IBM Z feature code 3863 (CP Assist for Cryptographic Functions) is enabled.

This feature code enables clear key DES and TDES instructions on all CPs. For more information, see the *Getting Started with z/OS Data Set Encryption*, SG24-8410 IBM Redbooks publication.

Define DFSMShsm to z/OS UNIX System Services

Define DFSMShsm to z/OS UNIX System Services as a superuser. Also, the DFSMShsm RACF user ID must have a default RACF group which has an OMVS segment with a group ID (GID). This user ID must also have an OMVS segment with the following parameters: UID(0) HOME('/')

Define SAF resources to control access to the cloud

The sample job in Example 6-21 defines SAF resources control access to the CLOUD keyword on the HMIGRATE end user command in DFSMShsm, and grants READ access to the STGADMIN.ARC.ENDUSER.HMIGRATE.CLOUD FACILITY class profile.

Example 6-21 Granting READ access to DFSMShsm to the migrate task

```
//STEPS001 EXEC PGM=IKJEFT01
//SYSTSPRT DD  SYSOUT=*
//SYSTSIN  DD  *
     RDEFINE FACILITY STGADMIN.ARC.ENDUSER.HMIGRATE.CLOUD UACC(NONE)
     PERMIT STGADMIN.ARC.ENDUSER.HMIGRATE.CLOUD CLASS(FACILITY) -
             ID(HSMUSER) ACCESS(READ)
     SETROPTS     RACLIST(FACILITY) REFRESH
/*
```

Part 3

Operation and Usage

In this part, we show you how we set up a cloud and how we used the new functionality to communicate with the cloud and to send data and retrieve data from it.

This part includes the following chapters:

7

DFSMShsm

In this chapter, we describe the changes that made to DFSMShsm to support the cloud tier.

This chapter includes the following topics:

7.1 Cloud use overview

DFSMShsm can use storage clouds by using DFSMSdss as a data mover to migrate and recall data sets from the cloud. The use of a cloud to migrate data sets can reduce your DASD and tape requirements, and provide disaster recovery capability for migrated data sets.

DFSMShsm tracks migrated data sets and their related objects in cloud storage in its control data sets (CDSs). By using the DFSMS Cloud construct to connect to the cloud (along with CDS information), DFSMShsm can manage data on cloud storage the same way it manages offline data.

Unlike tape devices, the space that is left by a deleted object is returned to free space and can be used by other objects as they are created. This feature eliminates the need for recycling containers on storage cloud, which reduces the MIPS that is related to **RECYCLE** processing and the window that is necessary to run DFSMShsm tasks. It also increases the availability of migrated data as RECYCLE processing holds HSM tapes during the recycle operation.

Also, the option of having an off-premise cloud increases the options that are available for disaster recovery. If your cloud is unaffected after a disaster, you can recover your production systems in an alternative location (including DFSMShsm CDSs) and define DFSMS Cloud construct to connect to your cloud. This ability gives you access to all data sets that are migrated to the cloud and have their related CDS records.

After a recovery situation, we suggest that you run an **AUDIT** against CDS and Cloud to report and correct any mismatches between them.

7.2 Cloud container management

DFSMShsm can create cloud containers to store the objects. By default, the container name adheres to the syntax that is shown in Example 7-1.

Example 7-1 Default DFSMShsm container name

```
SYSZARC.<HSMplexname>.MIG.yyyyddd
```

By default, a container is created every 7 days. Creating a container every few days allows you to better manage the objects within a container, list migrated data, and **AUDIT** the cloud.

If you decide an alternative value is required for the frequency of container creation that is based on listing performance considerations, you can change the default value from 7 days to shorter or larger amounts. Example 7-2 shows the **PATCH** command to change the default value from 7 to 10 days.

Example 7-2 PATCH to change container creation frequency

```
PATCH .MCVT.+50F X'09'
```

Just as DFSMShsm can automatically create containers when required, it can also delete empty and no longer used containers owned by DFSMShsm. You can configure DFSMShsm to perform empty container deletion as part of the Secondary Space Management tasks by using the new EMPTYCONTAINERDELETION(x) keyword on DFSMShsm SETSYS MAXSSMTASKS command.

The Example 7-3 shows a sample SETSYS command to allow 1Cloud Storage containers processing task. This is the default value. To prevent DFSMShsm from deleting empty containers, set the EMPTYCONTAINERDELETION to 0.

Example 7-3 Setting container deletion task

```
SETSYS MAXSSMTASKS(EMPTYCONTAINERDELETION(1))
```

If you disable automatic deletion of containers by DFSMShsm, you will need to create your own process to manage empty containers.

7.3 Object management

DFSMShsm can automatically create and delete objects from cloud storage by using DFSMSdss as the data mover. For each data set, DFSMSdss is started to migrate or recall the data by using transparent cloud tiering.

Storage objects that are created by DFSMShsm follow a new data set naming convention, which is similar to the naming convention that is used to ML1 migrate data sets. The object naming convention is shown in Example 7-4.

Example 7-4 DFSMShsm object naming convention

```
INSTPFX.HMIG.TCCCCHH.USER1.USER2.?YDDD
```

The naming convention consists of the following parts:

- `INSTPFX` is an installation defined prefix.
- `TCCCCHH` is a form of how HSM expresses the time, where `CCCC` is the number of hundredths of seconds since the beginning of the hour and compressed into four alphanumeric digits. `HH` is the hour. When there is a conflict, `T` can change to be from `U - S` (starting from `T` and wrapping around).
- `USER1.USER2` are the first two qualifiers of the data set name that is being migrated.
- `?YDDD` is the Julian Date where is `A - F` is for decade; for example, `2000 - 2060`.

How DFSMSdss handles the data depends on the request that is performed by DFSMShsm (a migration or recall process). These processes are described next.

7.3.1 Migration

When a migration process is started, DFSMShsm calls DFSMSdss to perform the data movement. HSM is responsible for passing to DSS the data set name, along with the Cloud constructs, including cloud name, account, container, and object prefix. DFSMSdss then communicates with the DS8880 passing information that is related to the tracks that should be moved to the cloud, along with cloud-related information.

The metadata is stored in the cloud directly by the host for Swift clouds, or DS8000 for S3 and IBM Cloud Object Storage (COS) clouds. DFSMSdss returns control to DFSMShsm after all data is moved to the cloud or after any failures during the process.

During the **`DUMP`** process, any data sets that are larger than 5 GB are broken up in 5 GB segments. **`VALIDATE`** processing is skipped for VSAM data sets.

7.3.2 Recall

During a recall request, DFSMShsm sends to DFSMSdss the data set name to be restored, along with the cloud attributes. DFSMSdss issues a request to the DS8880 for the objects that should be retrieved from the Cloud. Metadata is retrieved by the host for Swift clouds, and by DS8000 for S3 and IBM COS clouds.

At retrieval time, object segments (for data sets larger than 5 GB) are grouped, and data set extents are reduced when possible. During this phase, no REBLOCKing function is performed.

The storage objects can be deleted or retained during the recall process, if your HSMplex is configured to support fast subsequent migration.

7.4 Fast Subsequent Migration

When data sets are migrated to ML2, they are stored on tapes until the data set expires or is recalled. If a recall occurs, the data set is not physically deleted from the tape, but the CDS records are marked as invalid. With Fast Subsequent Migration, the recalled data set can be reconnected to the tape, which eliminates the need to rewrite the tape data.

Use of the storage cloud also allows you to reconnect recalled data sets to the cloud objects, which prevents a new migration, and thus reduces the network traffic to the cloud.

A new **SETSYS** command option (see Example 7-5) is available to include in your DFSMShsm parmlib to allow the reconnect.

Example 7-5 Set up fast subsequent migration

```
SETSYS CLOUDMIGRATION(RECONNECT(ALL))
```

7.5 Migration update and considerations

Data can be migrated to the cloud either by command, or during the automatic space management. The next topics will explain in more detail how you can manage your data for automatic and manual selection for migration.

7.5.1 Command-driven migration

A **CLOUD** parameter is available from the **MIGRATE** or **HMIGRATE** commands to target data sets to cloud. Example 7-6 shows a sample **HSEND MIGRATE** command with the **CLOUD** keyword.

Example 7-6 HSEND MIGRATE command cloud option

```
HSEND MIGRATE DSN(youdsname) CLOUD(yourcloud)
```

Note: The **CLOUD** parameter is mutually exclusive with **MIGRATIONLEVEL1**, **MIGRATIONLEVEL2**, and **CONVERT** parameters. Also, **COMPACT**, **COMPACTPERCENT**, **COMPACT(ALL)**, **CONVERSION(REBLOCKTOANY)**, and **CONCURRENT SETSYS** values are not used when migrating to the cloud.

To migrate a data set to the cloud, it must be SMS-managed. The types of data sets that can be migrated to the cloud, along with migration and recall restrictions, are listed in Table 7-1.

Table 7-1 Data set migration to cloud eligibility and considerations

Data set type	Can it be migrated to the cloud?	Comments
Non-SMS	N	Only SMS-managed data sets can be migrated to the cloud at the time of this writing.
Sequential	Y	
Extended Format	Y	
Extended format multi-volume	N	VSAM restrictions for HURBA=HARBA (used = allocated), and Multi-layer VSAM (volume count > stripe count) cannot be migrated.
Multi-extents sequential/partitioned	Y	Extent reduction is performed at recall time if possible.
Multi-volume sequential/partitioned	Y	
Multi-stripe sequential	Y	If SMS cannot provide enough volumes to keep the stripe count, the recall fails.
VSAM	Y	VALIDATE is not performed during migration.
Multi-extent VSAM	Y	VALIDATE is not performed during migration.
Multi-volume VSAM	Y	VALIDATE is not performed during migration.
VSAM with IBM AIX® and PATHs	Y	VALIDATE is not performed during migration.
Data sets in volumes with FlashCopy, XRC, Global Mirror, Metro Global Mirror, Multi-Target PPRC	N	As of this writing, only volumes with simplex or two-site Metro Mirror, with or without HyperSwap, are cloud-capable.
Data sets spanning more than 26 volumes	Y	An object cannot be restored to more than 26 volumes.
Multi-volume data sets spanning multiple DS8880s	Y	Cannot be restored in volumes spanning DS8880s.

The **HSEND MIGRATE** command that is issued from option ISPF panels to migrate a multi-extent sequential data set is shown in Example 7-7. There is no need to supply account, container, object prefix, or cloud credentials because they are handled by DFSMShsm.

Example 7-7 HSEND MIGRATE issued from ISPF panel

```
Menu  Options  View  Utilities  Compilers  Help
sssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
DSLIST - Data Sets Matching TCTRBOOK                                 Row 1 of 1
Command ===>                                                  Scroll ===> PAGE

Command - Enter "/" to select action                 Message           Volume
-------------------------------------------------------------------------------
```

```
HSEND MIGRATE DSN(/) cloud(IBMREDBOOKS)                                 D9S3SC
***************************** End of Data Set list ****************************
```

After the migration process is complete, the catalog entry is updated to reflect the new location of the data set. In ISPF, to differentiate data sets that are migrated to cloud from data sets on ML1 or ML2, a new **MIGRATC** volume serial number is used for cloud migrated data sets, as shown in Example 7-8.

Example 7-8 MIGRATC volume entry

```
Menu  Options  View  Utilities  Compilers  Help
sssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
DSLIST - Data Sets Matching TCTRBOOK                              Row 1 of 5
Command ===>                                                  Scroll ===> PAGE

Command - Enter "/" to select action                Message          Volume
------------------------------------------------------------------------------
         TCTRBOOK.ACDS.BKP                                           *VSAM*
         TCTRBOOK.ACDS.BKP.DATA                                      USRPAK
         TCTRBOOK.MIG.MYCLOUD                                        MIGRATC
         TCTRBOOK.SCDS                                               *VSAM*
         TCTRBOOK.SCDS.DATA                                          WRK132
***************************** End of Data Set list ****************************
```

A new device type of `x'00018000'` is used to identify data sets that are migrated to the cloud. The ICF catalog entry is shown in Example 7-9. Regardless of the migration tier, migrated data sets are always cataloged by using volser `MIGRAT`.

Example 7-9 Device type for data set migrated to the cloud

```
NONVSAM ------- TCTRBOOK.MIG.CLOUD
     IN-CAT --- SYS1.MVSRES9.MASTCAT
     HISTORY
        DATASET-OWNER-----(NULL)     CREATION--------2017.347
        RELEASE----------------2     EXPIRATION------0000.000
        ACCOUNT-INFO-----------------------------------(NULL)
      SMSDATA
        STORAGECLASS ---SCS3G016     MANAGEMENTCLASS--MCLASS2
        DATACLASS --------(NULL)     LBACKUP ---XXXX.XXX.XXXX
      VOLUMES
        VOLSER------------MIGRAT     DEVTYPE------X'00018000'     FSEQN---------
---------0
      ASSOCIATIONS--------(NULL)
      ATTRIBUTES
```

The information that is necessary to recall the data set from the cloud is stored in HSM CDSs. You can issue a **FIXCDS DISPLAY** command to view the created CDS records. A sample **FIXCDS** command format to display an MCD record is shown in Example 7-10.

Example 7-10 FIXCFDS command to view a CDS record

```
HSEND FIXCDS D dsname DISPLAY
```

The command output for a cloud-migrated data set is shown in Figure 7-11. Areas are highlighted to show recognizable eye catchers within the CDS record.

Example 7-11 Output of FIXCFDS command

```
MCH=   02580000 D394BE12 05EA36D3 D394BE12 05EA36D3
 *    L      LL      L            *
+0000  6CC3D3D6 E4C48001 00340000 0117347F 00000000 0117347F 00000000 00000000
 * CLOUD                         *
+0020  09523956 0117347F 40006D10 00908000 00000014 00000412 00000000 00010000
 *                               *
+0040  C4F9E2F3 E2F30200 00000000 3030200F 00000000 00000000 00010000 00000000
 *D9S3S3                         *
+0060  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
 *                               *
+0080  00000000 00000000 00000000 00000000 00000000 00000000 0000000F C4C6C8E2
 *                          DFHS*
+00A0  D44BC8D4 C9C74BE3 F3F6F5F2 F0F94BE3 C3E3D9C2 D6D6D24B D4C9C74B C1F7F3F4
 *M.HMIG.T365209.TCTRBOOK.MIG.A734*
+00C0  F7404040 40404040 00000000 00000000 00000000 00000000 00000000 00000000
 *7                              *
+00E0  00000000 00004040 40404040 40404040 40404040 40404040 40404040 40404040
 *                               *
+0100  40404040 0008E2C3 E2F3C7F0 F1F64040 40404040 40404040 40404040 40404040
 *      SCS3G016                 *
+0120  40404040 0007D4C3 D3C1E2E2 F2404040 40404040 40404040 40404040 40404040
 *      MCLASS2                  *
+0140  40404040 00000000 00000000 00800000 00000000 00000000 00000000 00000000
*                               *
+0160  00000000 00008000 00050000 00000000 00000000 03020200 00000000 00000050
 *                               *
+0180  A2404040 4000C400 00000000 0008E2C7 F9E2C8D9 D2F30000 00000000 00000000
 *      D       SG9SHRK3          *
+01A0  00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
 *                               *
+01C0  00000000 00000000 0007C3C2 D4D9C5C4 C3D6D6D2 E2404040 40404040 40404040
 *          IBMREDBOOKS *
+01E0  40404040 40404040 E2E8E2E9 C1D9C34B C1D9C3D7 D3C5E7F0 4BD4C9C7 4BF2F0F1
 *        SYSZARC.ARCPLEX0.MIG.201*
+0200  F7F3F4F4 40404040 40404040 40404040 40404040 00000005
*7344                            *
```

The **FIXCDS** command output includes the regular data set information (such as SMS constructs) and the cloud-related information (including the cloud name) and the container where the data set is stored.

7.5.2 Automatic migration

The DFSMShsm can also migrate data sets to the cloud during the automatic space management, including primary space management, interval migration, or on-demand migration.

To support automatic migration functions, new parameters were included in DFSMShsm parmlib and SMS management class constructs. These changes are described in more detail in Chapter 8, “Using automatic migration” on page 71.

7.5.3 CPU utilization considerations

Because the data movement to and from the cloud is performed directly by the DS8000, it is expected to have a variation on CPU utilization by DFSMShsm. You can create reports to estimate possible CPU savings from implementing cloud migration. The pre implementation reporting is discussed in more detail in Chapter 9, “Operational integration and reporting considerations” on page 75

7.6 Recall considerations

The RECALL process is automatically triggered when access to the data set is requested or the **HSEND RECALL** command is issued. There are no changes to the **RECALL** command.

During the **RECALL**, the ACS routines are called to define the Storage Class, Management Class, and Storage Group for the data set. The data set can also be extent-reduced if possible. Example 7-12 shows the recalled data set from the **HSEND MIGRATE** command issued, where the recalled data set has a single extent.

Example 7-12 Recalled data set with consolidated extent

```
Menu  Options  View  Utilities  Compilers  Help
sssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
DSLIST - Data Sets Matching TCTRBOOK                                 Row 3 of 5
Command ===>                                                  Scroll ===> PAGE

Command - Enter "/" to select action                       Tracks %Used   XT
------------------------------------------------------------------------------
         TCTRBOOK.MIG.CLOUD                                   340  100     1
```

When data sets expire, all data and metadata objects that are related to the data set are automatically deleted from the storage cloud and the catalog entry is removed.

7.7 LIST command updates

The **LIST** command has updates to support the cloud. DFSMShsm also gives you the options to list the following elements:

- Cloud
- Containers
- Objects

New **CLOUD**, **CONTAINER**, and **PREFIX** parameters can be used within the **LIST** command to retrieve cloud and container content information. The **LIST** command can list DFSMShsm and non-DFSMShsm owned containers, which gives the users the chance to list user-created containers and retrieve object information.

The output from the **LIST** command can be directed to a terminal or data set. The sample command that is used to list IBMREDBOOKS cloud information is shown in Example 7-13.

Example 7-13 LIST command for a specific cloud

```
HSEND LIST CLOUD(IBMREDBOOKS)
```

The command output is shown in Example 7-14.

Example 7-14 LIST command output

```
CLOUD NAME:     IBMREDBOOKS
CONTAINERS:
SYSZARC.ARCPLEX0.MIG.2018015
SYSZARC.ARCPLEX0.MIG.2017344
SYSZARC.ARCPLEX0.MIG.2017274
SYSZARC.ARCPLEX0.MIG.2021071
SYSZARC.ARCPLEX0.MIG.2017330
SYSZARC.ARCPLEX0.MIG.2017302
SYSZARC.ARCPLEX0.MIG.2017337
SYSZARC.ARCPLEX0.MIG.2017295
SYSZARC.ARCPLEX0.MIG.2017267
SYSZARC.ARCPLEX0.MIG.2018197
SYSZARC.ARCPLEX0.MIG.2018001
SYSZARC.ARCPLEX0.MIG.2017323
SYSZARC.ARCPLEX0.MIG.2018134
SYSZARC.ARCPLEX0.MIG.2017281
ARC0140I LIST COMPLETED,       16 LINE(S) OF DATA OUTPUT
```

The output shows the cloud `IBMREDBOOKS` and lists 16 containers, such as:

- SYSZARC.ARCPLEX0.MIG.2018134
- SYSZARC.ARCPLEX0.MIG.2017281

DFSMShsm and user-created containers data can be displayed by using the **`HSEND LIST`** command. Add the **`CONTAINER(containername)`** keyword to your **`LIST`** command. The **`HSEND LIST`** command lists the container SYSZARC.ARCPLEX0.MIG.2017281, as shown in Example 7-15.

Example 7-15 List a specific container in the IBMREDBOOKS cloud

```
HSEND LIST CLOUD(IBMREDBOOKS) CONTAINER(SYSZARC.ARCPLEX0.MIG.2017281)
```

The output from the command that is shown in Example 7-15 is shown in Example 7-16.

Example 7-16 LIST the content of a specific container

```
CLOUD NAME:     IBMREDBOOKS
CONTAINER NAME: SYSZARC.ARCPLEX0.MIG.2017281
OBJECT PREFIXES:
DFHSM.HMIG.T015821.TCT.VSM.A7284
DFHSM.HMIG.T015921.TCT.S01.A7284
DFHSM.HMIG.T035921.TCT.S01.A7284
DFHSM.HMIG.T055821.TCT.VSM.A7284
DFHSM.HMIG.T065921.TCT.S01.A7284
DFHSM.HMIG.T085821.TCT.VSM.A7284
DFHSM.HMIG.T085921.TCT.S01.A7284
DFHSM.HMIG.T090309.VNDROPF.TEST.A7285
```

Listing objects is available by using **`PREFIX`** parameter. When listing objects, the cloud and container names must also be included. The **`LIST`** command with the **`PREFIX`** name is shown in Example 7-17. This option brings all of the data and metadata objects that are stored under the specific prefix.

Example 7-17 LIST objects by using the PREFIX keyword

```
HSEND LIST CLOUD(IBMREDBOOKS) CONTAINER(SYSZARC.ARCPLEX0.MIG.2017281)
PREFIX(DFHSM.HMIG.T015821.TCT.VSM.A7284)
```

The output from the **LIST** command with the **CLOUD**, **CONTAINER**, and **PREFIX** command is shown in Example 7-18 on page 68.

Example 7-18 LIST objects by PREFIX

```
CLOUD NAME:     IBMREDBOOKS
CONTAINER NAME: SYSZARC.ARCPLEX0.MIG.2017281
OBJECT PREFIX:  DFHSM.HMIG.T015821.TCT.VSM.A7284
DATASET NAME:   TCT.VSM.N06.RRDS
OBJECT NAMES:
DFHSM.HMIG.T015821.TCT.VSM.A7284/DTPDSNL00000001
DFHSM.HMIG.T015821.TCT.VSM.A7284/HDR
DFHSM.HMIG.T015821.TCT.VSM.A7284/TCT.VSM.N06.RRDS/APPMETA
DFHSM.HMIG.T015821.TCT.VSM.A7284/TCT.VSM.N06.RRDS/DTPDSHDR
DFHSM.HMIG.T015821.TCT.VSM.A7284/TCT.VSM.N06.RRDS/DTPVOLD01/DATA/EXTENTS
DFHSM.HMIG.T015821.TCT.VSM.A7284/TCT.VSM.N06.RRDS/DTPVOLD01/DATA/META
ARC0140I LIST COMPLETED,       11 LINE(S) OF DATA OUTPUT
```

For more information about each object, see Table 3-1 on page 22.

7.8 Audit

The tasks to audit DFSMShsm data are vital to keep CDS records error free, identify, report, and correct any discrepancies between the CDS records and the data sets.

Unexpected software or hardware errors during migration and recall processes might leave migrated data sets and CDS records out of sync. Regularly auditing CDS and media reduces the number of orphan or invalid data in the physical media and the CDS.

AUDIT DATASETNAMES, **LEVEL**, and **MCDS** commands can perform a cloud migrated data sets audit. If the CDS record indicates that the data set is stored in cloud storage, DFSMShsm verifies that objects corresponding to the archive are in the expected cloud and container.

DFSMShsm lists the objects in the cloud, beginning with the prefix that is stored in the MCD record. If the expected objects are found, it moves onto the next MCD record.

In addition, a new **CLOUD** parameter is available from the **AUDIT MEDIACONTROLS** command. This option allows you to audit a cloud and validate if the objects have corresponding CDS entries. If any inconsistencies are found, the **AUDIT** command reports the error back to the user.

Note: The **AUDIT** command does *not* automatically fix any identified inconsistencies because the orphan objects can be user data that is in the wrong container.

An **HSEND AUDIT** command to audit IBMREDBOOKS cloud is shown in Example 7-19.

Example 7-19 Audit a specific cloud

```
HSEND AUDIT MEDIACONTROLS(CLOUD(IBMREDBOOKS))
```

The output from the command that is shown in Example 7-19 is shown in Example 7-20, with one inconsistent entry.

Example 7-20 Example of Audit output

```
Menu  Utilities  Compilers  Help
BROWSE    TCTRBOOK.AUDIT                              Line 00000000 Col 001 080
 Command ===>                                                Scroll ===> CSR
******************************** Top of Data ********************************
-DFSMSHSM AUDIT-       ENHANCED AUDIT -- LISTING - AT 11:06:10 ON 17/12/13 FOR S
COMMAND ENTERED:
AUDIT MEDIACONTROLS(CLOUD(IBMREDBOOKS)) ODS('TCTRBOOK.AUDIT')

/* ERR 210 CDD IS NOT FOUND FOR PREFIX DUMP.4XTENTS.PDS IN CONTAINER
/* DFHSM.HMIG.T015821.TCT.VSM.A7284
- END OF -     ENHANCED AUDIT - LISTING -
******************************** Bottom of Data ********************************
```

The prefix `DUMP.4XTENTS.PDS` is not in the CDS as expected. The follow-up action is to investigate why it is missing and resolve the issue.

7.9 REPORT command

After you first implement a storage cloud to your z/OS systems, it is suggested that you create reports about key metrics for analysis, such as these:

- Number of data set migrations to the cloud,
- Amount of data transferred,
- Number of successful/failed requests, and
- Average times.

The **REPORT** command provides all of the information that is necessary to efficiently monitor your data on the cloud.

You can create daily, weekly, or monthly reports and store this information for further analysis. You also can create simple programs to process the data and return reports with data growth, percentage of data sets migrated to the cloud versus standard migration, and usage trends.

A simple **REPORT** command to display migration statistics to the cloud is shown in Example 7-21.

Example 7-21 Report command for daily migration activity

```
HSEND REPORT DAILY FUNCTION(MIGRATION(TOCLOUD))
```

It is also possible to retrieve recall specific information by issuing the **REPORT** command, as shown in Example 7-22.

Example 7-22 Report command for daily recall activity

```
HSEND REPORT DAILY FUNCTION(RECALL(FROMCLOUD))
```

Migration and recall reports display the following migration-to-the-cloud information:

- Number of data sets
- Number of tracks read and written
- Number of bytes read and written
- Number of system requests
- Number of user requests
- Failed requests
- Average age
- Average queue time
- Average wait time
- Average process time
- Average total time

A sample output from the **HSEND REPORT DAILY** command is show in Example 7-23 on page 70.

Example 7-23 Report output for daily activity

```
****************************** Top of Data ******************************
1--DFSMSHSM STATISTICS REPORT ------- AT 12:42:33 ON 2017/12/13 FOR SYST

 DAILY STATISTICS REPORT FOR 17/12/13

 STARTUPS=000, SHUTDOWNS=000, ABENDS=000, WORK ELEMENTS PROCESSED=003023
 DATA SET MIGRATIONS BY VOLUME REQUEST= 0000000, DATA SET MIGRATIONS BY
 EXTENT REDUCTIONS= 0000000 RECALL MOUNTS AVOIDED= 00000 RECOVER MOUNTS
 DATA SET MIGRATIONS BY RECONNECTION =  000000, NUMBER OF TRACKS RECONNE

                     NUMBER  ------READ--------  -----WRITTEN------  ---
    HSM FUNCTION    DATASETS TRK/BLK    BYTES    TRK/BLK    BYTES    SYS

 MIGRATION
  PRIMARY - CLOUD   0000382  00064520 001632042  00064520 001632042  000


***************************** Bottom of Data ****************************
```

SMF records are also written when data sets are migrated to the cloud. You can use SMF records to create more specific reports that are based on users, high-level qualifiers, and other information. For more information about the SMF records and how to create reports, see Chapter 9, “Operational integration and reporting considerations” on page 75.

8

Using automatic migration

In this chapter, automatic migration function and cloud usage are described.

The automatic migration is performed as part of primary space management, interval, or on-demand migration tasks. The ability to automatically select data that is eligible for cloud migration is vital to maximize CPU savings related to data migration in DFSMShsm.

This chapter includes the following topics:

8.1 SMS support for automatic migration

To enable the DFSMShsm automatic migration to cloud, it is necessary to define the policies to be used to define if a data set will be migrated to existing ML2 volumes, or the defined cloud storage configured to your systems.

The definitions about data set migration are included in the SMS Management Class construct, and therefore new fields are included to allow storage administrators to define conditions that are required for a data set to be eligible for cloud migration.

8.1.1 Management Class updates

The data sets cannot be automatically migrated from ML2 to cloud the same way that ML1 data sets can be migrated to ML2. So, the management class was updated to create the rules for deciding the migration level tier.

The following fields are now available on Management Class panels:

- Size LTE

  The “Size Less than or equal” field shows the low data set size threshold in tracks. It will be used to take the action described on “Action LTE” field.

- Action LTE

  The value in the Action LTE column shows which action to perform if the data set size is less than or equal to Size LTE. The possible values are:

  - **NONE**: No action is taken.
  - **ML1**: Target migration level is ML1.
  - **ML2**: Target migration level is ML2 regardless of the values for LEVEL 1 DAYS NON-USAGE.
  - **MIG**: Target migration level is ML1 or ML2 according to the value of LEVEL 1 DAYS NON-USAGE.
  - **TRANS**: Data Set transition.
  - **CLOUD**: Target migration level is CLOUD.

  If no value is specified, DFSMShsm will perform the migration action the same way it is performed today.

- Size GT

  The “Size Greater than” field shows the high data set size threshold in tracks. It will be used to take the action described on “Action GT” field.

- Action GT

  The value in the Action GTcolumn shows which action to perform if the data set size is greater than Size GT. The possible values are:

  - **NONE**: No action is taken.
  - **ML1**: Target migration level is ML1.
  - **ML2**: Target migration level is ML2 regardless of the values for LEVEL 1 DAYS NON-USAGE.
  - **MIG**: Target migration level is ML1 or ML2 according to the value of LEVEL 1 DAYS NON-USAGE.
  - **TRANS**: Data Set transition.

  CLOUD: Target migration level is CLOUD.

If no value is specified, DFSMShsm will perform the migration action the same way it is performed today.

- Cloud Name

  The value in the CLOUD NAME shows the name of a previously defined cloud construct for the data set migration to the cloud during automatic migration (Primary Space Management, Interval migration, and OnDemand migration).

The new fields work in a way that is very similar to the existing DFSMShsm data set migration exit (MD), which controls the migration level for data sets selected by automatic migration processing. You can configure these fields using these values in the exit as well as disable the exit.

Note: If you have the DFSMShsm MD exit on, it will override the values used in the management class construct.

8.2 Storage Group affinity enhancements

The use of cloud storage by DFSMShsm can affect the length of your space management windows, as your migration to the cloud can take longer than ML1/ML2 migrations depending on your network usage, latency, and cloud performance.

For some large systems, with several storage groups and DFSMShsm images running, storage administrators might decide to spread the workload between these HSM images. There are enhancements to the storage group affinity to allow the selection of storage groups management for specific HSM hosts.

To enable the storage group affinity, you can use the new SETSYS command to identify the storage groups that will be managed by a given HSM host. Example 8-1 shows a sample SETSYS command to create the affinity between storage group BATCH1 and the HSMIMG2 DFSMShsm image.

Example 8-1 Sample STORAGEGROUPAFFINITY command

```
F HSMIMG2,SETSYS STORAGEGROUPAFFINITY(BATCH1)
```

By setting up the storage group affinity, the specific HSM usage will manage the defined storage groups. Define this command in your ARCCMD parmlib member to save your settings across Initial Program Load (IPLs).

9

Operational integration and reporting considerations

In this chapter, we review operational integration considerations.

The Storage cloud setup is the first stage in the process of moving your data into the cloud. Use of a strong operational framework including set of instructions, housekeeping jobs, and security considerations are encouraged to ensure that you can take the best from your cloud implementation.

We encourage you to consider the suggestions that are described in this chapter and plan and implement your own operations and automation procedures that are based on your system requirements.

This chapter includes the following topics:

9.1 Pre-implementation reporting

Before implementing automatic migration to the cloud, a storage administrator might want to report on possible Central Processing Unit (CPU) savings related to offloading the migration task to the DS8000.

IBM released a package to assist storage administrators to collect and report data regarding CPU consumption related to migration, recall, and recycle tasks. You can download the package from the following link:

ftp://public.dhe.ibm.com/eserver/zseries/zos/DFSMS/HSM/zTCT/

You can use this package to extract and parse your SMF data. Then, you can generate reports and graphics to analyze the CPU savings that can be achieved with cloud migration.

9.2 Operational monitoring

After you finish configuring and activating your cloud settings, the cloud is ready for use. You then perform several tests, and they all complete with success, meaning that the hardware and software configurations are correct, and the network access to the cloud is functional.

Now, it is time to ensure that this access remains functional as long as possible, and that any errors are tracked by automation systems.

9.2.1 Monitoring cloud setting changes

The first automation process that can be set up is to identify any cloud setting changes within DFSMShsm. Whenever a change to DFSMShsm storage cloud settings fails, a new ARC1581I message is issued with a description of the error that was encountered.

You might add message handling to our automation system to be notified whenever this message is issued and provide a timely reaction to the error. The error message code that is a result of an internal server error is shown in Example 9-1.

Example 9-1 ARC1581I message

```
Display  Filter  View  Print  Options  Search  Help
 -------------------------------------------------------------------------------
 SDSF SYSLOG     33.101 3090 3090 10/10/2016 OW         4,414   COLUMNS 52- 131
 COMMAND INPUT ===>                                           SCROLL ===> CSR
0010  ARC1581I UNEXPECTED HTTP STATUS 500 DURING A POST FOR 423
0010  ARC1581I (CONT.) URI
0010  ARC1581I (CONT.) https://IBMREDBOOKS.tuc.stglabs.ibm.com/v2.
0010  ARC1581I (CONT.) 0/tokens/ ERRTEXT HTTP/1.1 500 Internal Server Error
```

9.2.2 Monitoring migration activities

You can track data set migration failures by using the ARC0279I message, which is issued when a data set migration fails. Set up automation to report on these failures based on this message.

An alternative approach is to create a REXX program to read the DFSMShsm log or system log to periodically search for ARC0279I messages. A sample migration error that is the result of a non-existing cloud definition is shown in Example 9-2.

Example 9-2 ARC0279I message

```
Display  Filter  View  Print  Options  Search  Help
 -------------------------------------------------------------------------------
 SDSF OUTPUT DISPLAY DFHSM    STC00131  DSID     2 LINE 1,361   COLUMNS 02- 81
 COMMAND INPUT ===>                                           SCROLL ===> CSR
17.00.38 STC00131  ARC0279I MIGRATION REJECTED - CLOUD NAME VOIDCLOUD NOT  770
   770             ARC0279I (CONT.) FOUND
```

Use the information from the REXX-generated reports to track users that might be trying to migrate invalid data sets, or specifying wrong cloud information. The information helps to identify what information can be included in training for those users that are new to the cloud.

9.2.3 Monitoring reconnections

If you implemented periodic checks of the HSM SETSYS configurations, include the cloud configuration information. The new ARC0444I message identifies if cloud-recalled data sets can be reconnected to cloud objects. A sample output from the **HSEND QUERY SETSYS** command to display cloud-reconnect setting is shown in Example 9-3.

Example 9-3 ARC0444I reconnection message

```
Display  Filter  View  Print  Options  Search  Help
 -------------------------------------------------------------------------------
 SDSF OUTPUT DISPLAY DFHSM    STC00131  DSID     2 LINE 120     COLUMNS 02- 81
 COMMAND INPUT ===>                                           SCROLL ===> CSR
   273             ARC0410I (CONT.) PERCENTAGE=020%, TAPEMAXRECALLTASKS=01, ML2
   273             ARC0410I (CONT.) NOT ASSOCIATED GOAL=010, RECONNECT(NONE)
06.17.32 STC00131  ARC0444I CLOUDMIGRATION RECONNECT(ALL)
06.17.32 STC00131  ARC0411I TAPESECURITY=PASSWORD, DEFERMOUNT
```

9.2.4 Other messages to consider

Other situations that can be monitored include the **DUMP** and **RESTORE** processes that are performed by DFSMSdss. Several new messages identify and describe errors during **DUMP** or **RESTORE** processing. Consider investigating the following messages to determine whether they can help build a robust operational cloud:

- ADR600E: DFSMSdss did not process the data set because of the condition code detected.
- ADR601E: DFSMSdss invokes the **ANTRQST** macro for an **MCLIST**, **STORE**, or **RETRIEVE** request and ANTRQST fails with the listed hex return code, reason code, and return information.
- ADR602E: DFSMSdss found that a backup exists with the same object prefix in the specified container.
- ADR604E: A failure occurred while trying to store an object that is related to the dump process or a data set. All related objects stored that use the object-pre fix-name are not usable because of a previous error that was encountered.

- ADR606E: A failure occurred while performing the identified z/OS Client Web Enablement Toolkit service.
- ADR607E: A failure occurred while performing the identified request.
- ADR609E: I/O errors were encountered while the indicated type of dump meta-record was being read during logical data set RESTORE processing.
- ADR610E: DFSMSdss detected an unexpected internal error during processing of an HTTP/HTTPS request.
- ADR612E: DFSMSdss encountered an error obtaining a SYSZADRO enqueue; the resource might be in use.
- ADR705E: A nonexistent storage class, management class, or cloud was specified in the `STORCLAS`, `MGMTCLAS`, or `CLOUD` keyword.

Other monitor options can also be implemented in your systems to monitor and control how clouds are used.

9.3 Operational reporting

There are different options for reporting on DFSMShsm cloud usage. Whether by using the **`HSEND REPORT`** command or SMF records, plan to have a reporting and archiving job to analyze and retain storage cloud usage.

9.3.1 Building reports

The JCL and REXX that are included in this topic are intended to show you how to extract cloud migration and recall activity from a daily report, and append the data in a CSV format to output data sets. This file might then be downloaded and imported into a spreadsheet for further analysis.

The JCL that is used to run the report by running the REXX **`RXMEMBER`** procedure is shown in Example 9-4.

Example 9-4 JCL to run the REXX procedure

```
//JOBLIST1 JOB  (XXXX),'RUN RPT',NOTIFY=&SYSUID,MSGLEVEL=(1,1),
// MSGCLASS=W
//STEP1    EXEC PGM=IKJEFT01,REGION=8M
//SYSTSPRT DD   SYSOUT=A
//HSMREPT  DD DSN=YOUR.INPUT.DATA,DISP=SHR
//CLOUDRPT DD DSN=YOUR.OUTPUT.REPORT,DISP=(NEW,CATLG),
//     LRECL=80,RECFM=FB,SPACE=(TRK,(1,1)),DSORG=PS
//SYSTSIN  DD   *
 EX 'YOUREXX.DATASET(RXMEMBER)'
```

The REXX source code is shown in Example 9-5. You can use this code as a base to develop your own specific reports.

Example 9-5 REXX source code

```
/* REXX */
"EXECIO * DISKR HSMREPT (STEM HSMREPT. FINIS)"
/* NUMBER OF MIGRATION TO CLOUD LINES */
MIG=0
```

```
/* NUMBER OF RECALLS TO CLOUD LINES */
REC=0
DO Z=1 TO HSMREPT.0
   /* GET REPORT DATE */
   IF LASTPOS('DAILY STATISTICS REPORT FOR',HSMREPT.Z) > 0 THEN
      PARSE VAR HSMREPT.Z 'DAILY STATISTICS REPORT FOR' REPDATE .
   IF LASTPOS('PRIMARY - CLOUD',HSMREPT.Z) > 0 THEN DO
   PARSE VAR HSMREPT.Z . . . NDS RTRK RBYT WTRK WBYT SYSR USRR FAIL ,
      AGE QTIME WTIME PTIME TTIME
      MIG = MIG + 1
      OUTMIG.MIG = REPDATE','NDS','RTRK','RBYT','WTRK','WBYT','SYSR,
    ||','USRR','FAIL','AGE','QTIME','WTIME','PTIME','TTIME
   END
   IF LASTPOS('CLOUD - PRIMARY',HSMREPT.Z) > 0 THEN DO
   PARSE VAR HSMREPT.Z . . . NDS RTRK RBYT WTRK WBYT SYSR USRR FAIL ,
      AGE QTIME WTIME PTIME TTIME
      REC = REC + 1
      OUTREC.REC = REPDATE','NDS','RTRK','RBYT','WTRK','WBYT','SYSR,
    ||','USRR','FAIL','AGE','QTIME','WTIME','PTIME','TTIME
   END
END
"EXECIO * DISKW CLOUDRPT (STEM HSMREPT. FINIS)"
```

Other reports can be created by using SMF records. Some suggestions of reports that can be generated include filtering migration and recall by data set high-level qualifiers, users, management class, data set size, and others. We suggest that you create at least one report that consolidates data sets by high-level qualifiers, so that you can identify the applications that are making most use of cloud resources.

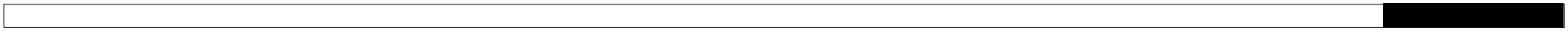
9.3.2 DCOLLECT reports

Along with the changes in DFSMShsm Control Data Sets (CDSs) and SMS constructs to enable the use of cloud storage, the DCOLLECT was also updated to reflect the extra information available.

In the DCOLLECT record type ‘MC’, cloud-related fields are also displayed, including the cloud names the management class relates to, and actions to take based on data set size during migration.

The ‘M’ records are updated to include the cloud name length, cloud name, container name, and number of objects stored.

A sample usage for this extra information includes using the DCOLLECT to gather information about the containers created and owned by DFSMShsm in the cloud, and the number of objects stored. This might be specially valuable for large cloud environments, where list commands can take an extended amount of time to complete.



(0.1”spine)
0.1”<->0.169”
53<->89 pages

SG24-8381-01

ISBN 0738442178

Printed in U.S.A.