OA66450 Direct to Cloud support Publication Updates

DFSMSdss

History

APAR	Date	Changes
OA66450	01/24/2025	Initial
OA67394	02/27/2025	Compression updates for DUMP keyword

This document describes the updates to the z/OS 3.1 DFSMSdss publication sections as a result of the DFSMSdss Direct to Cloud enhancement.

1 Overview

The OA66450 APAR introduces a new disk-storage agnostic cloud solution called Direct to Cloud. This solution allows DFSMSdss to target cloud object storage directly.

Direct to Cloud is only supported on version 3.1. If the user attempts to restore a backup created with Direct to Cloud on version 2.5, the restore will fail with ADR610E reason code 5.
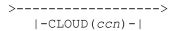
2 User Actions

To target cloud object storage instead of tape or DASD, the user must take several actions:

- Specify the required keywords for targeting CLOUD object storage (see Section 2.1)
- Configure Cloud Data Access for use (See Section 2.4)

2.1 Dump/Restore - Required Keywords

The following keywords must be included in the JCL steps for DUMP and RESTORE operations to use Direct to Cloud support.
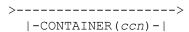
2.1.1 CLOUD

```
>------------------->
    |-CLOUD(ccn)-|
```

This keyword indicates the object storage cloud that dss will communicate with for the operation.

A provider file called *ccn.*json must exist in the required location with all of the dss required key-value pairs. For more information on the provider file, see section 2.4 of this document.

2.1.2 CONTAINER

```
>-------------------->
  |-CONTAINER(ccn)-|
```

This keyword indicates the container (also known as a bucket) that the backup will be placed in or restored from. During a dump operation, if the container does not exist, the container will be created.

The name specified can be up to 128 characters in length and will be folded to lowercase when communicating with the cloud. The allowable characters are uppercase letters A-Z, numbers 0-9, special characters $ @ # - _, and . in the non-first position.

If ccn does not start with a prefix of 'SYSZADR-', this prefix will be appended to the container name used. If the ccn does begin with the string 'SYSZADR-' or 'SYSZADR.', no prefix will be added.

2.1.3 OBJECTPREFIX

```
>------------------------>
    |-OBJECTPREFIX(opx)-|
    |------OBJPFX-------|
```

OBJECTPREFIX specifies a unique prefix that DFSMSdss is to use for all of the objects the make up a particular backup. The name can be up to 44 characters in length.

The prefix provides uniqueness amongst multiple backups in the same container. At the beginning of the backup process, DFSMSdss will determine if a backup using the same *opx* exists in the specified cloud and container. If a backup already exists with the same *opx*, then DFSMSdss will fail the backup. To overwrite a backup using the same *opx*, you can set a patch at offset x'5D'.

2.1.4 CDAPROVIDERFILE

```
>---------------------->
    |-CDAPROVIDERFILE-|
    |-----CDAPROV-----|
```

CDAPROVIDERFILE must be specified to use the direct to cloud support. If this keyword is not included, the cloud solution used revert to the default of transparent cloud tiering (TCT).

In order to verify that the direct to cloud solution was used, users can look in the listing for the following message:

```
ADR618I (R/I)-RI03 (01), CLOUD INFORMATION OBTAINED FROM CDA IS REQUESTING
DIRECT
```

Any other inserts in this message indicate that the system is attempting to use TCT rather than direct to cloud.

An installation may decide they want the direct to cloud support to be the default option in lieu of specifying the CDAPROVIDERFILE keyword. New patch byte at offset '62'x may be set to a value greater than 0. DFSMSdss will use the new support regardless of whether the keyword is specified.

For more information on the provider file, see section 2.4 of this document.

2.1.4.1 Optional Keywords for DUMP and RESTORE
The following keywords are optional when performing a direct to cloud dump or restore operation.

2.1.4.2 DEBUG
```
>------------------------------------->
    |-DEBUG-(-CLDMSG-(--MINimal-----)-|
```

```
                             |-SUMmarized-|
                             |-DETAILED---|
                             |-DTL--------|
```

You can use DEBUG as a diagnostic tool. When you specify the CLMSG subkeyword, DFSMSdss issues messages that provide details on the progress of a backup to an object storage cloud. When DEBUG(CLMSG) is not specified, MINimal is the default. Specify DEBUG(CLMSG) with one of the following sub-keywords:

- `CLMSG(MINimal)`– Neither CDA nor ZWEBP debug messages will be provided. ADR608I will be issued only for direct to cloud connections.
- `CLMSG(SUMmarized)`– Neither CDA nor ZWEBP debug messages will be provided. ADR608I will be issued for direct to cloud and TCT connections.
- `CLMSG(DETAILED)`– CDA and ZWEBP debug messages will be provided in SYSOUT.

2.1.4.3 Optional Keyword for DUMP
The following keyword is optional when performing a direct to cloud dump operation

2.1.4.4 CDACOMPRESS
```
>----------------------------------->
   |-CDACOMPRESS-|
   |---CDACOMP---|
```

When specified, DSS will request CDA to use zEDC services to compress user data prior to storing it in the cloud. During RESTORE, DSS will automatically request decompression if an object had been compressed.

2.2 CLOUDUTILS Operations

The CLOUDUTILS command can be used to list or delete elements of DFSMSdss created backups in an object-storage cloud.

The CLOUDUTILS command does not operate at the object granularity level. Use it only to manage DFSMSdss backups.

Like with DUMP and RESTORE, CLOUDUTILS will fold the container name to lowercase. CDAPROVIDERFILE must be specified to use a direct to cloud connection.

2.2.1 LIST

Specifies that the CLOUDUTILS command should perform a LIST operation.

The behavior of this keyword depends on other keywords:

- If CONTAINER is not specified, the request is to list all DFSMSdss-created containers.
- If CONTAINER is specified and OBJECTPREFIX is not specified, the request is to list all DFSMSdss dump data sets within the specified container. The list will result in object prefixes of one or more backups as specified during their dump operation.
- If OBJECTPREFIX is additionally specified, the request is to list all DFSMSdss dump data sets within the specified container that begin with the specified object prefix. The list will result in object prefixes of one or more backups as specified during their dump operation.

To list individual objects rather than dumps, use the ALL keyword.

2.2.2 DELETE

Specifies that the CLOUDUTILS command should perform a DELETE operation.

The behavior of this keyword depends on other keywords:

- If OBJECTPREFIX is not specified, the request is to delete a container. The container must be empty for the operation to succeed.
- If OBJECTPREFIX is specified, the request is to delete the DFSMSdss dump data set objects that correspond to the specified object prefix.

To delete a non-empty container, use the FORCE keyword.

2.3 CDA Setup

Direct to cloud requires a CDA environment to be set up. Instructions to set up CDA can be found here:

- https://www.ibm.com/docs/en/zos/3.1.0?topic=services-cloud-data-access-configuration
- https://www.ibm.com/docs/en/zos/3.1.0?topic=services-cloud-data-access-files

2.4 CDA Provider File

Direct to cloud requires the user invoking DFSMSdss to have an OMVS segment because a CDA provider file is necessary for this new functionality. The provider file is a JSON format file that contains information about how to connect to the cloud. It must be named in the form '*cloud_name*. json' such that the DFSMSdss command is specifies CLOUD(*cloud_name*).

2.4.1 Sample Provider Files

Sample provider files are shipped by DFSMSdss and can be found in '/usr/lpp/dfsms/dss/samples/'. For a direct to cloud provider file, file S3CLOUD.json should be used as a starting point. Please note that there will be some key-value pairs in the sample file that will need to be removed; see the following section for more information. CDA also ships sample files, however there are additional required fields that are necessary for them to be compatible with DFSMSdss.

2.4.2 Key-Value Pairs

The following keys must be specified in the provider file in order to use Direct to Cloud:

- enableDFSMSdss – This key identifies whether or not DFSMSdss can utilize this provider file. A value other than YES or the absence of this key will result in the provider file being ignored.
- host – the URI for the Cloud Object Storage provider.
    - The port and sslKey can be specified in the host name with the format: {https://}<host>{:<port>}{,<sslKey>}.
- sslVersion – the SSL version the connection will use. The accepted values are: TLSV10, TLSV11, TLSV12, or TLSV13

The following keys can be specified for direct to cloud if needed:

- port – This field can be omitted if the port is specified in the host, and if specified, the value in the host will override the value specified here. If this keyword is not specified, the CDA default port of 443 will be used.
- sslKey – the key ring to be used. If it is not included, the default value of '*AUTH*/*' will be used.

The following keys should not be specified for a direct to cloud connection: tctType, tctIdentity, and cloudName. If specified, these keys will be ignored.

2.4.3 The Supported Operations List

All of the following supported operation keys and their associated values must be present in the supported operations list to use direct to cloud:

- GETOBJECT
- GETLARGEOBJECT
- WRITEOBJECT
- WRITELARGEOBJECT
- LISTOBJECT
- DELETEOBJECT
- CREATEBUCKET
- DELETEBUCKET
- LISTBUCKETS

Since the supported operations list is quite long, it is recommended that the supported operations list be copied from the provided sample provider file located at '/usr/lpp/dfsms/dss/samples/S3CLOUD.json'.

3 Messages

The following messages are useful for debugging direct to cloud processing.

3.1 ADR617E

```
ADR617E(ttt)-mmmmm(yy), Z/OS CLOUD DATA ACCESS ENCOUNTERED AN ERROR WHILE
PERFORMING A cda_service SERVICE, RETURN CODE cda_rc

[RETURN CODE TRANSLATION: explanation]
[FOR OBJECT: object_name]
[OPERATION NAME: op_name]
[HTTP RESPONSE CODE: http_code]
[HWTHRQST RETURN CODE: hwth_code]
[HWTHRQST DIAGAREA: diagarea]
[HTTP RESPONSE BODY: response_body]
```

Explanation

A failure occurred while performing the identified Cloud Data Access service (*cda_service*). The *cda_service* completed with the specified return code (*cda_rc*).   The message may be followed by one or more of the following pieces of additional information.

- *explanation* – The CDA service return code (cda_rc) text translation.
- *object_name* – The object name that was being processed at the time of error
- *op_name* – The identified service was GDKGEN, this field identifies the operation that was requested
- *http_code* – The service performed the operation and resulted in this HTTP response code error. Value is presented in hexadecimal.
- *hwth_code* – CDA invoked the z/OS Client Web Enablement Toolkit HWTHRQST service. The service returned this error. Value is presented in hexadecimal.
- *diagarea* – This is the z/OS Client Web Enablement Toolkit HWTHRQST service DiagArea.

- *response_body* – The first 100 bytes of the HTTP response body.

System Action

This error precludes the use of cloud storage. The return code is set to 8.

Operator Response

None.

Programmer response

Refer to the identified z/OS Cloud Data Access callable service and associated return code in z/OS MVS Programming: Callable Services for High Level Languages to identify the reason for the error.

If HWTHRQST areas are provided, refer to the z/OS Client Web Enablement Toolkit section in z/OS MVS Programming: Callable Services for High Level Languages to identify the reason for the error.

3.2 ADR618I

```
ADR618I(ttt)-mmmmm(yy), CLOUD INFORMATION OBTAINED FROM {SMS | CDA} REQUESTING
{TAPE-OBJECT | SWIFT | TCT | DIRECT}
```

Explanation

The first insert indicates the following:

- SMS: DEBUG(CLMSG(SUMMARY)) or higher was specified and cloud information was obtained via SMS
- CDA: a valid CDA Provider File was found and used to obtain cloud information.

The second insert indicates the following:

- TAPE-OBJECT: The cloud protocol used to move data is TAPE-OBJECT. Both metadata and user data will be moved via TCT.
- SWIFT: The cloud protocol used to move information is SWIFT. Both metadata and user data will be moved via TCT.
- TCT: User data will be moved via TCT, and metadata will be moved via the Direct to Cloud path.
- DIRECT: Both metadata and user data will be moved via the Direct to Cloud path

System action

Processing continues.

Operator response

None.

Programmer response

None.

Source

DFSMSdss

3.3 ADR608E

```
ADR608E (ttt)-mmmmm(yy), INPUT META-DATA OBJECT objectname {NOT PRODUCED BY
DFSMSDSS | IS IN TCT FORMAT | IS IN DIRECT FORMAT}{ , UNSUPPORTED ON THIS
RELEASE}
```

Explanation

Depending on the insert, one of the following is true:

- NOT PRODUCED BY DFSMS: Either the input object is invalid for the type of RESTORE desired or the input was not produced by DFSMSdss.
- IS IN TCT FORMAT: The input object was created using TCT, but the user is attempting to restore it using a Direct to Cloud connection.
- IS IN DIRECT FORMAT: The input object was created using a Direct to Cloud connection, but the user is attempting to restore it using TCT.
- UNSUPPORTED ON THIS RELEASE: The user is attempting to restore a Direct to Cloud backup, which is not supported on release 2.5

System Action

The task ends. Processing continues with the next control statement. The return code is 8.

Operator Response

None.

Programmer Response

Take the appropriate action, depending on the error, and rerun the job.

Source

DFSMSdss

3.4 ADR582E

```
ADR582E(ttt) - mmmmm(yy), AN ERROR OCCURRED DURING CEEPIPI PROCESSING, return
code
```

Explanation

A failure occurred during Language Environment (CEEPIPI) set up necessary to use Cloud Data Access services.

Return code

The Language Environment return code

System action

The error precludes the use of cloud storage. The return code is set to 8.

Operator response

None.

Programmer response

See CEEPIPI return codes:

https://www.ibm.com/docs/en/zos/3.1.0?topic=preinitialization-ceepipi-invocation-subroutine-by-address

Source

DFSMSdss