OA66579 Publication Updates

DFSMSdfp CDA

04/16/2025

Andrew Wilt

This document describes the updates to the z/OS 3.1 publications as a result of the new function support delivered via OA66579.

1 Overview

The OA66579 APAR delivers new support to allow data uploaded from a z/OS sequential data set, VSAM Key Sequenced data set, or VSAM Entry Sequenced data set to keep its record boundaries while stored in a cloud object. Additionally, download of a cloud object that contains those record boundary information will be recognized, and the records will be placed into the local data set. As an additional ease-of-use enhancement, DFSMSdfp CDA will create the local data set based on the metadata tags associated with the cloud object when the local data set does not exist in the z/OS Catalog.

The GDKUTIL program is updated to support new keywords on UPLOAD FORMAT(RECORD|NONE), as well as the new metadata CDA variables. The DOWNLOAD command is updated to automatically recognize an object that has the record information via the “zos-filedata: record” metadata tag. Additional new keywords; STORCLAS, MGMTCLAS, DATACLAS, LOCSIZE, VOLUMES, UNIT, and LOCSIZE are added to allow override of original values for the data set’s SMS storage class, SMS management class, data class, amount of data in bytes used to create the data set, or disk volume for creation of the data set.

1.1 User Actions

In order to start using the new support, the user of CDA services (either through the GDKUTIL program, or an application that invokes the APIs), must take some actions.

1.1.1 Provider File Updates

The CDA sample provider files found in /usr/lpp/dfsms/gdk/samples/providers/ are updated to reflect the enablement of new support. One way to find the updates is to compare your provider file with the equivalent sample provider file. Another way is to take the sample provider file as-is, and update it with the unique values for your cloud provider.

Following is a description of updates and their associated provider file sections.

- The GETOBJECT operation needs to be able to understand metadata tags. In the GETOBJECT operation, a new responseResults array is added. A new entry is added that specifies METAHEADER as the mechanism with a descriptor of "descriptor": "<metadata-header-prefix>" . <metadata-header-prefix> is the appropriate prefix for metadata tags that are returned by the cloud object storage provider. For example, the S3 metadata-header-prefix is x-amz-meta-. For Microsoft Azure object storage, the metadata-header-prefix is x-ms-meta- .
- The GETLARGEOBJECT operation also needs to be able to understand metadata tags when performing the getSize action. In the GETLARGEOBJECT operation, a new JSON object is added to the responseResults array in the “getsize” action, a new responseResults is added in the “data” action. It specifies METAHEADER as the mechanism with a “descriptor” value of "<meta-header-prefix>" . The value of <meta-header-prefix> is described in the above bullet.
- The WRITEOBJECT operation needs to be able to create metadata tags. In the WRITEOBJECT operation, the requestParameters array must contain an entry with “mechanism”: “METAHEADER” and a “descriptor”: “<metadata-header-prefix>”
- The WRITELARGEOBJECT operation needs to understand how to create metadata tags. In the init action of the WRITELARGEOBJECT operation, the requestParameters array must contain an entry with “mechanism”: “METAHEADER” and a “descriptor”: “<metadata-header-prefix>”

1.1.2 New Keyword Exploitation

To use the new support, some updates to JCL or an application are required.

1.1.2.1 GDKUTIL invocations

- To request that the object from a z/OS data set contain information about the records during Upload, you should write JCL that invokes PGM=GDKUTIL with a SYSIN that specifies UPLOAD FORMAT(RECORD). This will cause DFSMSdfp CDA to gather metadata information about the data set and imbed record prefix areas between each user data record from the data set.
- To download an object with the imbedded record prefixes, and have the data reconstituted into a data set, no additional changes are needed as long as the object has the metadata tag that ends with "zos-filedata" and the value is "record", and your provider file has the METAHEADER updates for GETOBJECT and GETLARGEOBJECT. DFSMSdfp CDA will recognize the metadata tag and place the records into the output data set.
    - If you know that the object has imbedded record prefix areas in the data, but either your provider file doesn't have the METAHEADER description in the GETOBJECT/GETLARGEOBJECT operations, or the object does not have the "zos-filedata:record" metadata tag, you can specify the FORMAT(RECORD) keyword to cause DFSMSdfp CDA to perform the record reconstitution.
    - If you know that the object has the imbedded record prefix data, but do not want it to be reconstituted, you may specify FORMAT(NONE) to download the object as-is without any further processing.
    - If you are downloading the object to a different system from which it came, and the SMS environment is different, you can use the STORCLAS keyword to specify a new storage class name to use when creating the data set. You can specify STORCLAS() with nothing in the parentheses to override the metadata storage class name, possibly converting the data set to be non-SMS managed. (Of course, the ACS routines and other factors in your environment may cause different results.)
    - Likewise, you can use the MGMTCLAS and DATACLAS keywords to change or nullify the respective class names.
    - You can use the VOLUMES keyword to specify a volume serial (VOLSER) to use when creating the data set.
    - Additionally, you may wish to override the UNIT name associated with the volume you wish the data set to be created on. You can use the UNIT keyword to specify the unit name to be included on the DYNALLOC call when creating the data set.
    - Sometimes, you may wish to override the size that CDA uses to calculate how big of a data set to create. You can specify the LOCSIZE keyword with the number of bytes that the data set should hold in its primary allocation.
- The LIST command is updated to recognize two new keywords, which enable redirection of output to an alternate DD, as well as tailoring the date format for the objects listed.
    - LISTOUTDD(<ddname>) – Specify the <ddname> you want the list output to be written to.
    - LISTDATEFMT('<format_string>') – Using the variables documented for the strftime() function: https://www.ibm.com/docs/en/zos/3.1.0?topic=functions-strftime-convert-formatted-time#strfti , you can tailor the date format for each object listed. All text between the single-quotes is included.

1.1.2.2 Application usage

In order to use the new functionality from an application, new optional parameters are required.

- The GDKWRITE API is updated to recognize new optional parameters for the GDK_PATHDATALOCATION source type.
    - "cloudformat" can be passed with a value of "record" or "none". "records" indicates that you want the data set records to be separated by imbedded record prefix areas, which indicate a length of the user record. "none" is the default and indicates that no record prefixes should be imbedded in the cloud object.

- When cloudformat:record is requested, metadata tags with the keys listed in the table below will be attached to the cloud object as appropriate for the data set.

Metadata key	Value
zos-lrecl	Set to the logical record length of the data set
zos-recfm	Set to the record format of the data set
zos-blksize	Set to the blocksize of the data set
zos-dsorg	Set to the data set organization of the data set.
zos-dataclas	Set to the DATACLAS for the data set
zos-mgmtclas	Set to the SMS Management Class for the data set
zos-storclas	Set to the SMS Storage Class for the data set
zos-secondary	Set to the secondary allocation amount in the format <nnnn><type> where type is CYLS, TRKS, BLKS
zos-vs-account	Set to the value of the ACCOUNT for the VSAM data set if it exists.
zos-vs-buffspace	Set to the value of the BUFFSPACE for the VSAM data set if not default.
zos-vs-bwotype	Set to the backup-while-open value for the VSAM data set.
zos-vs-dcisize	Set to the VSAM cluster Data Component CISIZE
zos-vs-icisize	Set to the Indexed VSAM cluster Index Component CISIZE.
zos-eattr	Set to the Extended Attribute value for the VSAM data set.
zos-vs-erase	Set to the ERASE setting for the VSAM data set
zos-vs-freespace	Set to the VSAM data set FREESPACE value
zos-vs-keylabel	Set to the VSAM data set KEYLABEL name if it exists
zos-vs-keys	Set to the Primary Key field information for the VSAM data set.
zos-vs-log	Set to the value of the LOG keyword for the VSAM data set.
zos-vs-logreplicate	Set to the value of the LOGREPLICATE keyword for the VSAM Data set.
zos-vs-logstreamid	Set to the LOGSTREAMID value for the VSAM data set if it exists.
zos-vs-owner	Set to the OWNER value for the VSAM data set if it exists.
zos-vs-recordsize	Set to the RECORDSIZE value for the VSAM data set.
zos-vs-spanned	Set to SPANNED when the VSAM data set was defined as having spanned records.
zos-volumes	Set to a hyphen separated list of volume serials for the data set.

- The GDKGET API is updated to recognize a new optional parameter.
  - If the “cloudformat” optional parameter is not passed, then the existence of the “zos-filedata” metadata tag with the value “record” will be used to decide whether to parse imbedded record prefixes from the object data.
  - “cloudformat” can be passed with a value of “record” or “none”. “records” indicates that the object data contains imbedded record prefixes which indicate data set records. The data set records will be written to the local data set. “none” is the default and indicates that no record prefixes should be imbedded in the cloud object.

        - When cloudformat:record is requested or the zos-filedata tag is found, metadata tags with the keys listed in the table below will be recognized and used to create the local data set when the local data set is not cataloged.
    - "storclas" can be passed to override the value found from the object metadata tags. The string can be simply the empty string, which indicates that no storclas name should be passed on the allocation.
    - "mgmtclas" can be passed to override the value found from the object metadata tags. The string can be simply the empty string, which indicates that no mgmtclas name should be passed on the allocation.
    - "dataclas" can be passed to override the value found from the object metadata tags. The string can be simply the empty string, which indicates that no dataclas name should be passed on the allocation.
    - "volumes" can be passed to override the volume serials used on the create request. This should be a comma-separated list of volume serials.
    - "unit" can be passed to override the unit name passed on the DYNALLOC request. The string can be simply the empty string, which indicates that no UNIT name should be passed. When not specified, the UNIT name used is SYSALLDA.
    - "LocalSize" can be passed to override the number of bytes used to calculate the primary allocation amount of the data set.

The GDKWRITE API is updated for the GDK_PATHDATALOCATION source type to utilize the ability to request that data uploaded from a z/OS data set have record prefix areas saved so that the data set can be reconstituted on a z/OS system from the cloud object. New CDA metadata variables are recognized, and additionally automatically used when uploading a data set using the new format:record functionality.

The GDKGET API is updated for the GDK_PATHDATALOCATION source type to recognize an object as containing the record prefixes.

CDA symbol	Meaning
GDK_DSORGE	• PDSE for PDS/E dataset • PDS for PDS dataset • PS_LARGE when is a large format data set. (DS1LARGE) • PS_EXT when is extended format. (DS1STRP) • PS when data set is basic format sequential. • VSAMESDS when data set is Entry Sequenced VSAM • VSAMKSDS when data set is Key Sequenced VSAM
GDK_VSACCOUNT	32 bytes of accounting information and user data for a VSAM data set. Resolves to the value originally specified on the ACCOUNT() parameter when the VSAM data set was defined.
GDK_VSBUFFSPACE	The BUFFERSPACE value for the VSAM data set.
GDK_VSBWOTYPE	The backup-while-open (BWO) value for the VSAM data set. Will resolve to TYPECICS, TYPEIMS, or NO.
GDK_VSDCISIZE	The size of the control interval for the VSAM data set data component.
GDK_VSICISIZE	The size of the control interval for the VSAM data set index component.
GDK_EATTR	The Extended Attribute value for the VSAM data set. Resolved values are: OPT or NO
GDK_VSERASE	Indicator whether the cluster's components are to be erased when its entry in the catalog are deleted. Resolved values are: ERASE or NOERASE.

GDK_VSFREESPACE	Value of the free space to be set aside after the initial load of the VSAM data set. Values are: cinnn-cannn
GDK_VSKEYLABEL	The key label name associated with the VSAM data set.
GDK_VSKEYS	Primary key field information for the VSAM data set. Resolved values are in the form of len_nnn-off_nnn ,where len_nnn is the length of the key, and off_nnn is the zero based offset in the record that the primary key starts at.
GDK_VSLOG	Value from the LOG keyword for the VSAM data set. Resolved values are: NONE, UNDO, and ALL.
GDK_VSLOGREPLICATE	Value for the VSAM data set eligibility for VSAM replication. Resolved values are: LOGREPLICATE, and NOLOGREPLICATE.
GDK_VSLOGSTREAMID	Name for the forward recovery log stream for the VSAM data set.
GDK_VSOWNER	Resolves to the cluster's owner userid.
GDK_VSRECORDSIZE	The average and maximum lengths of the records in the data component. Values are in the format: avg_nnn-max_nnn, where avg_nnn is the average record size, and max_nnn is the maximum record size.
GDK_VSREUSE	Resolves to the indicator whether the VSAM cluster can be opened again and again as a reusable cluster. Values are: REUSE or NOREUSE
GDK_VSSHAREOPTIONS	The share options value for the VSAM data set. Values are in the format: reg_n-sys_n, where reg_n is the crossregion option value, and sys_n is the crosssystem option value. i.e. 1-3
GDK_VSSPANNED	Attribute of the VSAM data set indicating whether it can contain records that cross control interval boundaries. Resolved values are: SPANNED, or NON_SPANNED
GDK_VSPREFORMAT	Attribute of the VSAM data set indicating whether the control areas of the VSAM data component should be preformatted during loading. Resolved values are: SPEED, or RECOVERY.
GDK_VOLSER	This will resolve to a hyphen separated list of volume serials for the current data set. e.g. VOL1-VOL002-VOLA1E

2 Publication Updates

2.1 z/OS MVS Programming: Callable Services for High Level Languages

MVS Programming: Callable Services for High Level Languages

SA23-1377

Part 11. Cloud Data Access Services is updated as follows:

2.1.1 Chapter 26. Cloud Data Access(CDA) API basics

Chapter 26. Cloud Data Access (CDA) API basics – Elements of Cloud Data Access is updated to add the following new bullets:

Upload a z/OS data set, imbedding record prefixes in between data records.

Download an object with imbedded record prefixes, writing data records to the z/OS data set.

Create a z/OS data set based on metadata tags from the object.

Chapter 30. Cloud Data Access callable services is updated as follows:

2.1.1.1 GDKGET — Retrieve a cloud object

The Optional Parameters table is updated to add the following rows:

cloudformat	Character	This can have a value of “record” or “none”. “records” indicates that you know that the object contains imbedded record prefixes separating the record data. “none” indicates that no processing of imbedded record data should be performed. When an object has the metadata tag indicating cloudformat:record, or it is requested, metadata tags with the keys listed in the table below will be recognized and used to create the local data set when the local data set is not cataloged.
storclas	Character	String containing a value to override the name found from the object metadata tags. The string can be simply the empty string, which indicates that no STORCLAS name should be passed on the allocation.
mgmtclas	Character	String containing a value to override the name found from the object metadata tags. The string can be simply the empty string, which indicates that no MGMTCLAS name should be passed on the allocation.
dataclas	Character	String containing a value to override the value found from the object metadata tags. The string can be simply the empty string, which indicates that no DATACLAS name should be passed on the allocation.
volumes	Character	String containing a comma separated list of volume serials to override the volume serials used on the create request.
unit	Character	String containing a value to override the unit name passed on the DYNALLOC request. The string can be simply the empty string, which indicates that no UNIT name should be passed. When not specified, the UNIT name used is SYSALLDA.
LocalSize	Character	String that contains a number indicating the number of bytes that the primary allocation of the data set should hold.

Metadata tags recognized and used to create a local data set

Metadata key	Value
`zos-lrecl`	Set to the logical record length of the data set
`zos-recfm`	Set to the record format of the data set
`zos-blksize`	Set to the block size of the data set
`zos-dsorg`	Set to the data set organization of the data set.
`zos-dataclas`	Set to the DATACLAS for the data set
`zos-mgmtclas`	Set to the SMS Management Class for the data set
`zos-storclas`	Set to the SMS Storage Class for the data set

zos-secondary	Set to the secondary allocation amount in the format <nnnn><type> where type is CYLS, TRKS, BLKS
zos-vs-account	Set to the value of the ACCOUNT for the VSAM data set if it exists.
zos-vs-buffspace	Set to the value of the BUFFSPACE for the VSAM data set if not default.
zos-vs-bwotype	Set to the backup-while-open value for the VSAM data set.
zos-vs-dcisize	Set to the VSAM cluster Data Component CISIZE
zos-vs-icisize	Set to the Indexed VSAM cluster Index Component CISIZE.
zos-eattr	Set to the Extended Attribute value for the VSAM data set.
zos-vs-erase	Set to the ERASE setting for the VSAM data set
zos-vs-freespace	Set to the VSAM data set FREESPACE value
zos-vs-keylabel	Set to the VSAM data set KEYLABEL name if it exists
zos-vs-keys	Set to the Primary Key field information for the VSAM data set.
zos-vs-log	Set to the value of the LOG keyword for the VSAM data set.
zos-vs-logreplicate	Set to the value of the LOGREPLICATE keyword for the VSAM Data set.
zos-vs-logstreamid	Set to the LOGSTREAMID value for the VSAM data set if it exists.
zos-vs-owner	Set to the OWNER value for the VSAM data set if it exists.
zos-vs-recordsize	Set to the RECORDSIZE value for the VSAM data set.
zos-vs-spanned	Set to SPANNED when the VSAM data set was defined as having spanned records.
zos-volumes	Set to a hyphen separated list of volume serials for the data set.

Return and reason codes

Return code	Constant Name	Explanation
145	GDK_ICONV_ERROR	Conversion error, converting from one code page to another code page.
148	GDK_DECOMPRESSION_INIT_FAILURE	An error occurred while initialising the stream to perform decompression.
149	GDK_DECOMPRESSION_FAILURE	An error occurred while decompressing the data.
151	GDK_NO_MEMORY_AVAILABLE	Could not allocate memory for decompressing the data.
152	GDK_INVALID_COMP_PARMS	Invalid decompress or compression parameter passed.
159	GDK_INVALID_METADATA	An invalid metadata string was passed. Examine the log for ERROR messages relating to

		which metadata symbol was invalid for the operation.
160	GDK_INVALID_CF_PARMS	“cloudformat” was passed, but the value was invalid.
161	GDK_UNSUPPORTED_DS	The requested data set is of a type that DFSMSdfp CDA does not support for processing.

2.1.1.2 GDKWRITE – Send a cloud object

In the description of the optionalParmStructPtr, the Optional Parameters table is updated to add the following rows:

cloudformat	character	This can have a value of “record” or “none”. “records” indicates that you want the data set records to be separated by record prefixes as mapped by IGGRPFX, which indicate a length of the user record. “none” is the default and indicates that no record prefixes should be imbedded in the cloud object. Note: This optional parameter only applies when the datalocation parameter is GDK_PATHDATALOCATION, and the path name is for a z/OS Data Set. z/OS UNIX files are not supported.

Add a new table that describes the IGGRPFX mapping:

Offset	Length	Symbol	Description (IGGRPFX)
0	**1**	**RPFZ00**	**Reserved**
1	**3**	**RPFXLLL**	**Length of record that follows this prefix**

CDA symbol Table update:

CDA symbol	Meaning
GDK_DSORGE	• PDSE for PDS/E dataset • PDS for PDS dataset • PS_LARGE when is a large format data set. (DS1LARGE) • PS_EXT when is extended format. (DS1STRP) • PS when data set is basic format sequential. • VSAMESDS when data set is Entry Sequenced VSAM • VSAMKSDS when data set is Key Sequenced VSAM
GDK_VSACCOUNT	32 bytes of accounting information and user data for a VSAM data set. Resolves to the value originally specified on the ACCOUNT() parameter when the VSAM data set was defined.
GDK_VSBUFFSPACE	The BUFFERSPACE value for the VSAM data set.
GDK_VSBWOTYPE	The backup-while-open (BWO) value for the VSAM data set. Will resolve to TYPECICS, TYPEIMS, or NO.
GDK_VSDCISIZE	The size of the control interval for the VSAM data set data component.

GDK_VSICISIZE	The size of the control interval for the VSAM data set index component.
GDK_EATTR	The Extended Attribute value for the VSAM data set. Resolved values are: OPT or NO
GDK_VSERASE	Indicator whether the cluster's components are to be erased when its entry in the catalog are deleted. Resolved values are: ERASE or NOERASE.
GDK_VSFREESPACE	Value of the free space to be set aside after the initial load of the VSAM data set. Values are: cinnn-cannn
GDK_VSKEYLABEL	The key label name associated with the VSAM data set.
GDK_VSKEYS	Primary key field information for the VSAM data set. Resolved values are in the form of len_nnn-off_nnn ,where len_nnn is the length of the key, and off_nnn is the zero based offset in the record that the primary key starts at.
GDK_VSLOG	Value from the LOG keyword for the VSAM data set. Resolved values are: NONE, UNDO, and ALL.
GDK_VSLOGREPLICATE	Value for the VSAM data set eligibility for VSAM replication. Resolved values are: LOGREPLICATE, and NOLOGREPLICATE.
GDK_VSLOGSTREAMID	Name for the forward recovery log stream for the VSAM data set.
GDK_VSOWNER	Resolves to the cluster's owner userid.
GDK_VSRECORDSIZE	The average and maximum lengths of the records in the data component. Values are in the format: avg_nnn-max_nnn, where avg_nnn is the average record size, and max_nnn is the maximum record size.
GDK_VSREUSE	Resolves to the indicator whether the VSAM cluster can be opened again and again as a reusable cluster. Values are: REUSE or NOREUSE
GDK_VSSHAREOPTIONS	The share options value for the VSAM data set. Values are in the format: reg_n-sys_n, where reg_n is the crossregion option value, and sys_n is the crosssystem option value. i.e. 1-3
GDK_VSSPANNED	Attribute of the VSAM data set indicating whether it can contain records that cross control interval boundaries. Resolved values are: SPANNED, or NON_SPANNED
GDK_VSPREFORMAT	Attribute of the VSAM data set indicating whether the control areas of the VSAM data component should be preformatted during loading. Resolved values are: SPEED, or RECOVERY.
GDK_VOLSER	This will resolve to a hyphen separated list of volume serials for the current data set. e.g. VOL1-VOL002-VOLA1E

Return and Reason Codes

Return code	**Constant Name**	**Explanation**
145	GDK_ICONV_ERROR	Conversion error, converting from one code page to another code page.

148	GDK_COMPRESSION_INIT_FAILURE	An error occurred while initialising the stream to perform compression.
149	GDK_COMPRESSION_FAILURE	An error occurred while compressing the data.
151	GDK_NO_MEMORY_AVAILABLE	Could not allocate memory for compressing the data.
152	GDK_INVALID_COMP_PARMS	Invalid compression or compLevel parameter passed.
155	GDK_FULLY_QUALIFIED_DS_NOT_PASSED	The dataLocation name that is a z/OS data set is not a fully qualified data set name.
156	GDK_COULD_NOT_ALLOCATE_DATASET	An error occurred while CDA was trying to create a new data set. An ERROR level message will describe the S99ERROR or S99INFO values that describe the error. Details about the error codes can be found in “Interpreting error reason codes from DYNALLOC”. If the error was for a VSAM data set, refer to the IDCxxxxx messages that are included in the output. DEBUG level logging may provide additional information.
157	GDK_PDS_OR_PDSE_NOT_FOUND	The specified data set name for a PDS or PDSE data set was not found in the Catalog.
160	GDK_INVALID_CF_PARMS	“cloudformat” was passed, but the value was invalid.
161	GDK_UNSUPPORTED_DS	The requested data set is of a type that DFSMSdfp CDA does not support for processing.

2.2 z/OS DFSMSdfp Utilities

SC23-6864-60

Chapter 2: GDKUTIL (Cloud Object Utility) Program, is updated. In the SYSIN Statement table, make the following changes:

In the Commands section, update the description for the DOWNLOAD command to read as follows:

Commands
Download Requests the data in the object named by the OBJNAME DD for the provider reference on the PROVIDER keyword be written to the z/OS UNIX file, or Data Set/member referenced on the LOCAL or LOCNAME DD. If the LOCAL DD is used, and DISP=OLD is used, the data will be overwritten. If DISP=MOD is used, the data will be appended to the end. The request will fail if DISP=SHR is used.

If LOCNAME is used and the z/OS UNIX file or data set exists, its data will be overwritten. Likewise, if an error occurs before data is written to the data set, it may be left empty, with the previous data no longer accessible. If the z/OS UNIX file or data set does not exist and the object does not have the , it will be created using the defaults described in the fopen() defaults section of the "Input and Output" chapter of the z/OS XL C/C++ Programming Guide. These defaults may not meet your expectations for containing the data. Minimum length DOWN. **Note:** If not transferring in text mode, and the target data set is RECFM=V or RECFM=VB, all records will be the same length, except possibly the last.

In the Keywords section, add the following rows in the appropriate place:

FORMAT(RECORD|NONE)	When specified on the UPLOAD command for a z/OS data set, FORMAT(RECORD) indicates that DFSMSdfp CDA should imbed record prefixes into the user data so that the cloud object may be downloaded to z/OS while re-creating the records in the data set. Metadata will be associated with the object to identify the object as containing the imbedded record prefix data. FORMAT(NONE) is the default. The DOWNLOAD command does not require specification of this keyword, since DFSMSdfp CDA will recognize the zos-filedata:record metadata tag, and process the data accordingly. However, if you know that the data within an object contains the imbedded record prefixes, but the object does not have the metadata tag, you may specify FORMAT(RECORD) to force processing of the imbedded record prefix data. Likewise, if you want to override automatic processing of the imbedded record prefix data, you may specify FORMAT(NONE). When creating a data set, the metadata for the object must contain at least the zos-recfm and zos-lrecl tags. Note: z/OS UNIX files are not supported for UPLOAD with FORMAT(RECORD).
STORCLAS(<storage_class>)	This keyword may be specified on the DOWNLOAD command, and allows you to override the SMS storage class name used when creating the data set when it doesn't exist. This keyword may be specified with nothing between the parentheses to indicate that no name should be used when creating the data set.
MGMTCLAS(<management_class>)	The MGMTCLAS keyword is recognized with the DOWNLOAD command, and allows an override of the SMS management class name used when creating the data set when it doesn't exist. This keyword may be specified with nothing between the parentheses to indicate that no name should be used when creating the data set.
DATACLAS(<data_class>)	The DATACLAS keyword is recognized with the DOWNLOAD command, and allows an override of the data class name used when creating the data set when it doesn't exist. This keyword may be specified with nothing between the parentheses to indicate that no name should be used when creating the data set.
VOLUMES(<volume_list>)	The VOLUMES keyword is recognized with the DOWNLOAD command and allows an override of the

	volume serials used when creating the local data set. If more than one volume is passed, the volume serials should be separated by commas. If the local data set is to be a sequential data set, only the first volume serial is used. If the local data set is to be a VSAM data set, all volume serials will be passed when creating the data set.
UNIT(<unit_name>)	The UNIT keyword is recognized with the DOWNLOAD command, and allows an override to the unit name passed on the DYNALLOC request when creating a non-VSAM data set. The string can be simply the empty string, which indicates that no UNIT name should be passed. When not specified, the UNIT name used is SYSALLDA.
LOCSIZE(<bytes>)	The LOCSIZE keyword may be specified to override the size of the data set created. The size specified is in number of bytes of data to be written to the data set.
LISTOUTDD(<ddname>) LISTDD(<ddname>)	The output from a LIST command may be directed to a different DD statement using the LISTOUTDD keyword. Specify the name of a DD statement for the step where the object list information should be written to. The record length of the data should allow for the full records to be written. (Maximum object name length is 1024 characters.) If the named DD does not exist, SYSPRINT will be used for the LIST output.
LISTDATEFMT('<format>')	The format of the date fields in the LIST output may be tailored to your needs using the LISTDATEFMT keyword. Use syntax conforming to the strftime() conversion specifiers to indicate how the object date should be formatted. https://www.ibm.com/docs/en/zos/3.1.0?topic=functions-strftime-convert-formatted-time#strfti All text between the single quote characters is considered the format text.

The CDA Symbols table is updated to add the following entries:

CDA symbol	Meaning
GDK_DSORGE	• PDSE for PDS/E dataset • PDS for PDS dataset • PS_LARGE when is a large format data set. (DS1LARGE) • PS_EXT when is extended format. (DS1STRP) • PS when data set is basic format sequential. • VSAMESDS when data set is Entry Sequenced VSAM • VSAMKSDS when data set is Key Sequenced VSAM
GDK_VSACCOUNT	32 bytes of accounting information and user data for a VSAM data set. Resolves to the value originally specified on the ACCOUNT() parameter when the VSAM data set was defined.
GDK_VSBUFFSPACE	The BUFFERSPACE value for the VSAM data set.
GDK_VSBWOTYPE	The backup-while-open (BWO) value for the VSAM data set. Will resolve to TYPECICS, TYPEIMS, or NO.
GDK_VSDCISIZE	The size of the control interval for the VSAM data set data component.

GDK_VSICISIZE	The size of the control interval for the VSAM data set index component.
GDK_EATTR	The Extended Attribute value for the VSAM data set. Resolved values are: OPT or NO
GDK_VSERASE	Indicator whether the cluster's components are to be erased when its entry in the catalog are deleted. Resolved values are: ERASE or NOERASE.
GDK_VSFREESPACE	Value of the free space to be set aside after the initial load of the VSAM data set. Values are: cinnn-cannn
GDK_VSKEYLABEL	The key label name associated with the VSAM data set.
GDK_VSKEYS	Primary key field information for the VSAM data set. Resolved values are in the form of len_nnn-off_nnn ,where len_nnn is the length of the key, and off_nnn is the zero based offset in the record that the primary key starts at.
GDK_VSLOG	Value from the LOG keyword for the VSAM data set. Resolved values are: NONE, UNDO, and ALL.
GDK_VSLOGREPLICATE	Value for the VSAM data set eligibility for VSAM replication. Resolved values are: LOGREPLICATE, and NOLOGREPLICATE.
GDK_VSLOGSTREAMID	Name for the forward recovery log stream for the VSAM data set.
GDK_VSOWNER	Resolves to the cluster's owner userid.
GDK_VSRECORDSIZE	The average and maximum lengths of the records in the data component. Values are in the format: avg_nnn-max_nnn, where avg_nnn is the average record size, and max_nnn is the maximum record size.
GDK_VSREUSE	Resolves to the indicator whether the VSAM cluster can be opened again and again as a reusable cluster. Values are: REUSE or NOREUSE
GDK_VSSHAREOPTIONS	The share options value for the VSAM data set. Values are in the format: reg_n-sys_n, where reg_n is the crossregion option value, and sys_n is the crosssystem option value. i.e. 1-3
GDK_VSSPANNED	Attribute of the VSAM data set indicating whether it can contain records that cross control interval boundaries. Resolved values are: SPANNED, or NON_SPANNED
GDK_VSPREFORMAT	Attribute of the VSAM data set indicating whether the control areas of the VSAM data component should be preformatted during loading. Resolved values are: SPEED, or RECOVERY.
GDK_VOLSER	This will resolve to a hyphen separated list of volume serials for the current data set. e.g. VOL1-VOL002-VOLA1E

Reserved metadata tag keys when FORMAT(RECORD) is specified:

Metadata key	Value
`zos-lrecl`	Set to the logical record length of the data set
`zos-recfm`	Set to the record format of the data set
`zos-blksize`	Set to the block size of the data set
`zos-dsorg`	Set to the data set organization of the data set.
`zos-dataclas`	Set to the DATACLAS for the data set

zos-mgmtclas	Set to the SMS Management Class for the data set
zos-storclas	Set to the SMS Storage Class for the data set
zos-secondary	Set to the secondary allocation amount in the format <nnnn><type> where type is CYLS, TRKS, BLKS
zos-vs-account	Set to the value of the ACCOUNT for the VSAM data set if it exists.
zos-vs-buffspace	Set to the value of the BUFFSPACE for the VSAM data set if not default.
zos-vs-bwotype	Set to the backup-while-open value for the VSAM data set.
zos-vs-dcisize	Set to the VSAM cluster Data Component CISIZE
zos-vs-icisize	Set to the Indexed VSAM cluster Index Component CISIZE.
zos-eattr	Set to the Extended Attribute value for the VSAM data set.
zos-vs-erase	Set to the ERASE setting for the VSAM data set
zos-vs-freespace	Set to the VSAM data set FREESPACE value
zos-vs-keylabel	Set to the VSAM data set KEYLABEL name if it exists
zos-vs-keys	Set to the Primary Key field information for the VSAM data set.
zos-vs-log	Set to the value of the LOG keyword for the VSAM data set.
zos-vs-logreplicate	Set to the value of the LOGREPLICATE keyword for the VSAM Data set.
zos-vs-logstreamid	Set to the LOGSTREAMID value for the VSAM data set if it exists.
zos-vs-owner	Set to the OWNER value for the VSAM data set if it exists.
zos-vs-recordsize	Set to the RECORDSIZE value for the VSAM data set.
zos-vs-spanned	Set to SPANNED when the VSAM data set was defined as having spanned records.
zos-volumes	Set to a hyphen separated list of volume serials for the data set.

New examples are added to the **GDKUTIL Examples** section as follows:

Example 15: Upload a data set, keeping record information

In this example, the GDKUTIL utility is used to upload a z/OS data set to object storage, keeping the record boundaries intac.

```
//CRBUCKET EXEC PGM=GDKUTIL,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSIN DD *
  UPLOAD FORMAT(RECORD)
  PROVIDER(IBMCOS)
/*
//OBJNAME  DD  *
   /bucket-name/multi01/dir1/CavesofTerrorpg18970.txt
/*
//LOCNAME  DD  *
 BACKUPS.DFSMSDSS.D140325.DUMP
/*
```