

IBM Archive Manager for z/VM



User's Guide

Version 1 Release 1

IBM Archive Manager for z/VM



User's Guide

Version 1 Release 1

Note:

Before using this information and the product it supports, read the information in "Notices" on page 31.

| **Third Edition (November 2013)**

This edition applies to Version 1 Release 1 of IBM Archive Manager for z/VM (product number 5697-J05) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC18-9345-01.

© Copyright Rocket Software, Inc. 2003 - 2013. All Rights Reserved.

Contents

About this information	v
How to read syntax diagrams	v
How to send your comments	vi
Chapter 1. Introduction to Archive Manager	1
What's new in Archive Manager	1
What does Archive Manager do?	2
Archive Manager features and benefits	2
Supported data types	2
Supported storage media	2
Archive Manager terminology	2
Chapter 2. Requirements for archiving data	5
EDF (file-level) archive requirements	5
CKD (raw track image) archive requirements	5
SFS directory archive requirements	5
Requirements for archiving data to a staging disk	6
Chapter 3. Archive Manager commands	7
ARCHIVE	7
AMVLIST	9
HELP	13
QUERY CLASS	13
QUERY CLASSES	15
QUERY EXPIRE	16
QUERY PARCEL	17
QUERY PARCELS	20
QUERY QUOTA	21
QUERY TAG	23
RECALL	24
REMOVE	26
SET EXPIRE	27
SET TAG	28
Notices	31
Trademarks	33
Bibliography	35
Index	37

About this information

This information is designed to help system administrators administer and operate Archive Manager.

This information is intended for those persons using Archive Manager to archive and retrieve data and assumes a working knowledge of:

- The z/VM[®] operating system
- CMS (including XEDIT)

Specific changes since the previous edition of this book are indicated by a vertical bar (|) to the left of a change. Editorial changes that have no technical significance are not noted.

The Archive Manager for z/VM Web page provides current product documentation that you can view, print, and download. To locate publications with the most up-to-date information, refer to the following Web page:

<http://www.ibm.com/software/stormgmt/zvm/archive/library.html>

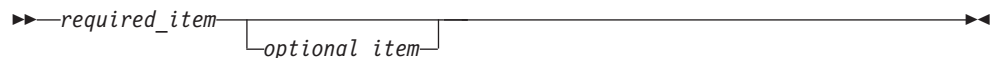
How to read syntax diagrams

The following rules apply to the syntax diagrams that are used in this information:

- Read the syntax diagrams from left to right, from top to bottom, following the path of the line. The following conventions are used:
 - The >>--- symbol indicates the beginning of a syntax diagram.
 - The ---> symbol indicates that the syntax diagram is continued on the next line.
 - The >--- symbol indicates that a syntax diagram is continued from the previous line.
 - The --->< symbol indicates the end of a syntax diagram.
- Required items appear on the horizontal line (the main path).



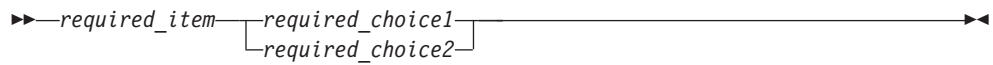
- Optional items appear below the main path.



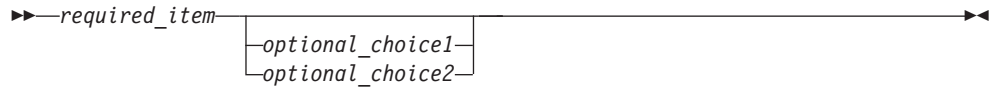
If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.



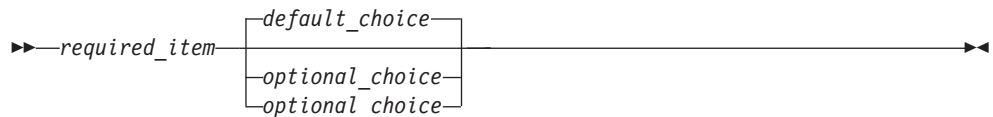
- If you can choose from two or more items, they appear vertically, in a stack. If you *must* choose one of the items, one item of the stack appears on the main path.



If choosing one of the items is optional, the entire stack appears below the main path.



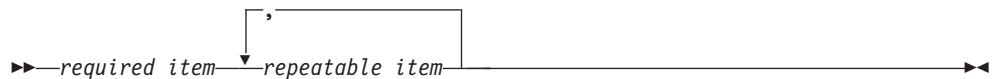
If one of the items is the default, it appears above the main path, and the remaining choices are shown below.



- An arrow returning to the left, above the main line, indicates an item that can be repeated.



If the repeat arrow contains a comma, you must separate repeated items with a comma.



A repeat arrow above a stack indicates that you can repeat the items in the stack.

- Keywords, and their minimum abbreviations if applicable, appear in uppercase. They must be spelled exactly as shown. Variables appear in all lowercase italic letters (for example, *column-name*). They represent user-supplied names or values.
- Separate keywords and parameters by at least one space if no intervening punctuation is shown in the diagram.
- Enter punctuation marks, parentheses, arithmetic operators, and other symbols exactly as shown in the diagram.
- Footnotes are shown by a number in parentheses; for example, (1).

How to send your comments

Your feedback is important in helping to provide the most accurate and high-quality information. If you have any comments about this book or any other Archive Manager for z/VM documentation, use either of the following options:

- If you have questions or comments regarding z/VM publications and product documentation, please visit: <http://www.vm.ibm.com/forms/>
- Send your comments by email to zvmtools@us.ibm.com. Be sure to include the name of the book, the part number of the book, the version of Archive Manager

for z/VM, and, if applicable, the specific location of the text you are commenting on (for example, a page number or table number).

Chapter 1. Introduction to Archive Manager

These topics introduce the functionality provided by Archive Manager including its features and benefits and how it performs processing.

Topics:

- “What's new in Archive Manager”
- “What does Archive Manager do?” on page 2
- “Archive Manager features and benefits” on page 2
- “Archive Manager terminology” on page 2

What's new in Archive Manager

This topic summarizes the technical changes for this edition.

New and changed information is indicated by a vertical bar (|) to the left of a change. Editorial changes that have no technical significance are not noted.

SC18-9348-02

- The ARCHIVE command description was updated. For more information, see “ARCHIVE” on page 7.
- The response to QUERY QUOTA was updated. For more information, see “QUERY QUOTA” on page 21.

SC18-9348-01

Information about archiving data was clarified. See Chapter 2, “Requirements for archiving data,” on page 5 for details.

- The following command descriptions, including examples, were updated:
 - ARCHIVE
 - REMOVE
 - SET EXPIRE
 - SET TAG
 - QUERY QUOTA
 - QUERY PARCELS
 - QUERY PARCEL
 - QUERY CLASSES
 - QUERY CLASS
 - QUERY EXPIRE
 - QUERY TAG

For more information, see Chapter 3, “Archive Manager commands,” on page 7.

An example was added to the RECALL command description. For more information, see “RECALL” on page 24.

What does Archive Manager do?

Archiving old or infrequently accessed data that you need to retain is an integral part of effectively managing DASD usage and maintaining data availability.

Archive Manager enables you to archive data to disk or tape and then retrieve it (or delete the data that is no longer needed).

Archive Manager features and benefits

By archiving data that you need to keep, but do not need to access on a frequent basis, you help your installation ensure efficient use of valuable DASD storage. Archiving data also assists your installation in complying with regulations that prescribe how long they must retain data, for example: for audit, payroll, or accounting purposes.

With Archive Manager you can perform the following tasks:

- Archive CMS and non-CMS data (individual files or entire minidisks) to disk or tape.
- Retrieve archived files at the file name or file type level, or by using wildcard characters to retrieve files that match specific criteria.
- Place archive files on to a staging disk before they go to tape (to archive data without having to wait for a tape mount).
- Help increase data availability by better utilizing DASD space.
- Archive, list, and recover files with mixed case.
- Perform detailed searches of files by file name, file type, user ID, disk address or directory, date created, date archived, and expiration date.
- Share access to archived files at the file level, share access to all files owned by a specific user ID, or share files with groups.

Supported data types

Archive Manager enables you to archive several types of data.

- CMS EDF minidisk (CMS formatted minidisks)
- CMS SFS (Shared File Systems)
- Raw image dumps of CKD (Count Key Data) including z/OS[®], Linux on System z[®], and z/VSE[®] volumes

Supported storage media

Archive Manager supports the following types of storage media.

- IBM[®] 3480, 3490, and 3590 storage media
- CMS files

Archive Manager terminology

Before you use Archive Manager, review the following terms and concepts.

Master archive service virtual machine (AMVARKIV)

The master Archive Manager service virtual machine verifies and processes archive requests that are initiated by clients.

Client A service virtual machine (typically, a CMS user).

Parcel A static image of archived client data and associated metadata.

Retention period

Specifies how long the data remains in the archive.

Storage class

Storage media that is owned by the master Archive Manager service virtual machine. (Parcels are stored on media that is associated with a storage class.)

Worker service virtual machine

A worker task service virtual machine that is used by the master Archive Manager service virtual machine when performing tape-oriented data handling operations.

Chapter 2. Requirements for archiving data

Archive Manager enables you to archive data, query data, and retrieve previously archived data. For archive storage purposes, any data type (CKD, EDF, or SFS) can be archived to any type of storage class (TAPE, EDF, or SFS).

Topics:

- “EDF (file-level) archive requirements”
- “CKD (raw track image) archive requirements”
- “SFS directory archive requirements”
- “Requirements for archiving data to a staging disk” on page 6

Notes:

1. You must own the source of the data that you want to archive (unless granted ADMIN privileges by an administrator).
2. Do not alter or delete the files that are selected for archive until you receive confirmation that archived objects were processed by the archive server.

EDF (file-level) archive requirements

The following requirements apply when you archive files from EDF minidisks.

- The minidisk where the files reside must be accessed by your user ID through the ACCESS command.
- At least one file on the minidisk must match the filter (*fnmask ftmask fm*) specified on the ARCHIVE command.

CKD (raw track image) archive requirements

The following requirements apply when you are archiving the image of a CKD minidisk or archiving files on CKD minidisk using the IMAGE option.

- The minidisk that contains the data that you want to archive must be accessed by your user ID through the ACCESS command.
- If the *fnmask ftmask fm* operands are specified, at least one file on the target minidisk must match the filter (*fnmask ftmask fm*) specified on the ARCHIVE command.

SFS directory archive requirements

The following requirements apply when you archive files from an SFS directory.

- The SFS directory that contains the files that you want to archive must be accessed by your user ID through the ACCESS command.
- The archive service virtual machine (AMVARKIV by default) must have SFS ADMIN authority.

For more information, see “ARCHIVE” on page 7.

Requirements for archiving data to a staging disk

Archive Manager stages archive data to disk before it writes it to its final archive location. This allows users to remove the data from their own disk sooner. Archive Manager has a copy on disk, and users do not need to wait for tape processing before they remove the file.

Note the following requirements for the staging disk:

- If you are staging to minidisk, verify that the staging area is formatted.
- If you are staging to SFS, ensure that the Archive Manager server (AMVARKIV by default) has access to the SFS directory.

Chapter 3. Archive Manager commands

The following topics describe Archive Manager commands to archive, query, and retrieve data. These commands are available to administrators and general users.

Note: The ARCHIVE and AMVLIST commands are issued as CMS commands that run in the user virtual machine.

The following commands are issued to the Archive Manager server (AMVARKIV by default) through CP SMSG. For example:

```
CP SMSG AMVARKIV CMS QUERY CMSLEVEL
```

- QUERY CLASS
- QUERY CLASSES
- QUERY EXPIRE.
- QUERY PARCEL
- QUERY PARCELS
- QUERY QUOTA
- QUERY TAG
- REMOVE
- SET EXPIRE
- SET TAG

Use the QUERY commands to obtain information about your archived data such as the storage class, parcel identifier, parcel size, and storage quota limits.

To retrieve archived data, use the RECALL command (users can recall the only the data they own). The RECALL command is intended to be issued from the panel interface (it can also be issued through CP SMSG).

ARCHIVE

The ARCHIVE command archives one or more files or non-CMS minidisks.

Syntax for CMS files (minidisk or SFS):

```
|—ARCHIVE—fnmask—ftmask—fm—(—| options |—————|
```

Syntax to archive non-CMS minidisks:

```
|—ARCHIVE—VDEV—devnum—(—| options |—————|
```

Figure 1. ARCHIVE command syntax

Authorization

- System administrators can archive any data to which they have access. For example, system administrators can archive data from minidisks or filespace

that they do not own, if they are able to obtain access to the minidisk or SFS directory before they issue the ARCHIVE command.

- General users can archive only the data that they own.

Note: The ARCHIVE command requires a read/write minidisk or SFS directory at file mode A.

Operands

fnmask

The name of the file (1-8 characters) to archive. To specify a subset of file names to be included in the archive operation, use wildcard characters.

ftmask

The type of file (1-8 characters) to archive. To specify a subset of file types to be included in the archive operation, use wildcard characters.

fm

The file mode of the file to archive (the minidisk or SFS directory must be accessed). Specify either a single file mode letter, letter and number (the valid numerical range is 0-6), or letter and asterisk (*).

VDEV *devnum*

The minidisk at virtual address *devnum* is subject to ownership requirements, but the actual minidisk content can be CMS or non-CMS data. (Archive Manager does not check to determine whether any CMS files are present on the minidisk.) This allows clients to archive non-CMS minidisks, such as Linux file system minidisks.

Note: See the *z/VM CMS Commands and Utilities Reference* for information about specifying wildcard characters.

Options

CLASS *classname*

The storage class name (*classname*) to receive the parcel.

Note: If CLASS is omitted, the storage class name of DEFAULT is used.

FILESPACE

(Applicable to CMS SFS directories only.) Specify FILESPACE to request an archive of the entire filespace that is associated with file mode *fm*. The FILESPACE option processes the entire SFS filespace, with file selection subject to *fnmask* *ftmask*.

The FILESPACE option differs from the SUBDIRS option. The SUBDIRS option performs a "drill-down" search of the file space, from the directory that is associated with *fm*.

IMAGE

(Applicable to CKD minidisks only.) Store a minidisk as a CKD image instead of as a set of files.

Notes:

1. You must specify *fnmask* and *ftmask* to preserve the syntax.
2. The IMAGE option is only valid when *fmspec* is associated with a minidisk. Currently, only CKD DASD is supported.

RETAIN *ndays* | **DEFAULT**

When you archive data, specify the number of days you want to retain the data in the archive using the RETAIN option on the ARCHIVE command.

RETAIN specifies how long to retain the archived data. Configuration of the destination storage class determines the default retention period and the valid range for *ndays*. The specific archive parcel is retained by the archive server for the number of days you specify.

Note:

- If you do not specify a retention period, Archive Manager uses the default retention period for the storage class that is associated with the data to be archived.
- Use the SET EXPIRE command to modify the retention period after you archive the data.

SUBDIRS

(Applicable to SFS directories only.) Specifying SUBDIRS indicates that subdirectories and their contents, in addition to the contents of the directory referenced by *fn*, are archived.

Note: If you omit the SUBDIRS option, only the contents of the referenced directory, without subordinate structures, is archived.

TAG *text-string*

Specify the TAG option to specify tag information for the resulting archive parcel. If specified, TAG must be the last option listed. All of the text after the TAG keyword is treated as tag information.

Usage notes

1. When you are archiving files from an SFS directory, the SFS directory that contains the files that you want to archive must be accessed by your user ID. (The default is to archive only the directory that is accessed.)

For example, consider the following SFS file space:

```
filepool:filesystem.
filepool:filesystem.dir1
filepool:filesystem.dir1.subdir1
filepool:filesystem.dir2
```

If you issue the ACCESS command: ACCESS filepool:filesystem.dir1 B

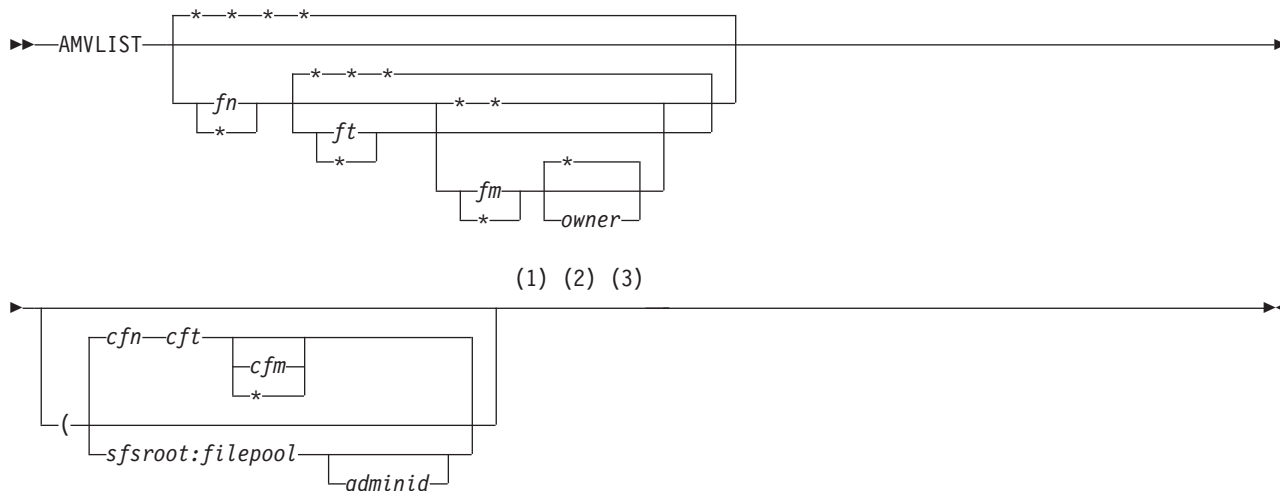
and then issue this ARCHIVE command	The resulting archive will contain
ARCHIVE * * B	only the files in filepool:filesystem.dir1
ARCHIVE * * B (SUBDIRS	all of the files in filepool:filesystem.dir1 and all of the files in filepool:filesystem.dir1.subdir1
ARCHIVE * * B (FILESPEC	all of the files in filepool:filesystem. and all files and all subdirectories

In all cases, selection of files and directories is subject to *filename* or *filetype* filtering as specified in the *fn ft* specification of the ARCHIVE command.

2. A successful archive operation using the ARCHIVE command generates a parcel that is identified by an eight character parcel identifier (*parcelid*).

AMVLIST

Issue the AMVLIST command to view information about archive files, parcels, owners, devices, and so on, within the archive catalog.



Notes:

- 1 If you invoke AMVLIST without parameters, it searches for the default configuration file, AMVSYSTEM CONFIG, on any accessed file mode.
- 2 If you invoke AMVLIST with an alternate configuration file parameter (*filename filetype*), the file mode defaults to an asterisk (*) if not specified.
- 3 If you invoke AMVLIST with *sfsroot:filepool*, no commands (such as RECALL) can be issued unless the appropriate administrator user ID (*adminid*) is also provided.

Figure 2. AMVLIST command syntax

Authorization

The AMVLIST command shows the portion of the archive catalog you are authorized to view.

Operands

- fn** The file names to display. Wildcard characters are permitted.
- ft** The file types to display. Wildcard characters are permitted.
- fm** The file modes to display. The value must be an asterisk (*) or a number (0-6).
- owner** The file owners to display. Wildcard characters are permitted.
- cfn** The file name of an alternate configuration file to use to determine the catalog and admin user ID.
- cft** The file type of an alternate configuration file to use to determine the catalog and admin user ID.
- cfm** The file mode of an alternate configuration file to use to determine the catalog and admin user ID.
- sfsroot**
An SFS catalog root to browse for archive jobs (as opposed to obtaining the root from an archive configuration file).

Note: *sfsroot* requires a *filepool* parameter.

filepool

Specifies a root file pool to be browsed for archive jobs (rather than obtaining the file pool from an archive configuration file).

Note: *filepool* is required with the *sfsroot* parameter.

adminid

The archive administrative user ID to which commands are to be issued when an SFS root and file pool are provided (rather than obtaining the appropriate information from an archive configuration file).

Note: Specifying *adminid* is optional, but no RECALL commands can be issued when you invoke the AMVLIST command with *sfsroot:filepool* unless *adminid* is also specified.

Usage notes

1. AMVLIST displays the portion of the archive catalog you are authorized to view, subject to the filtering of any *fn*, *ft*, *fm*, or owner operands on the command line.

If your authority allows you to see all, or a large part of the archive catalog, the default of (* * * *) can produce a very large file and require significant real time, CPU time, and virtual storage. To limit the initial selection, use the *fn*, *ft*, *fm*, and *owner* fields.

2. Lines are displayed as they are returned from the archive catalog inquiry. Use the sort options (PF5 and PF6) or filters to change both the lines that are displayed and the order in which they are displayed.
3. Filter the initial selection by specifying *fn*, *ft*, *fm*, *owner* on the command line. Further filter the results by using the name, type, mode, and owner filters on the initial AMVLIST panel (all filters default to *, so the initial display shows all lines that meet the command line filters). Filter on *filename*, *filetype*, *filemode*, or owner. In all cases, the filter is a regular expression in which various arbitrary characters can be specified as described in Table 1.

Table 1. Filter characters

Character	Description
* (asterisk)	Zero or more of any character
% (percent sign)	Any one character. (A percent sign indicates that a character is required, but it can be any character)
# (pound sign)	One numeric (0-9) character
& (ampersand)	One alphabetic (a-z, A-Z) character
@ (at symbol)	One hexadecimal (0-9, A-F, a-f) character
" (double quotation mark)	An escape character which causes the next character to be used as itself (even if it would normally have)

If you specify a set of filters that excludes all lines, the message `No lines passed filters.` is displayed and all lines in the file are displayed.

Because all archive catalog data is uppercase, all filters are translated to uppercase before they are applied.

PF key definitions

Table 2. AMVLIST PF key definitions

Key	Setting	Action
Enter	Run	Run the command (or commands) type on file lines.
PF1	Help	Display the AMVLIST command description.
PF3	Quit	Exit from AMVLIST.
PF4	Return	Return to the previous level, or exit if already at the highest level.
PF5	Sort up	Sort the displayed lines in ascending order by the column (job, instance, owner, type, device, or time or date) on which the cursor is located.
PF6	Sort down	Sort the displayed lines in descending order by the column (job, instance, owner, type, device, or time or date) on which the cursor is located.
PF7	Backward	Scroll back one screen.
PF8	Forward	Scroll forward one screen.
PF10	RECALL	If the cursor is on a line that displays an instance, the RECALL panel is invoked for that instance.
PF11	Display	If the cursor is on a line that displays a granule, detailed information from that granule is extracted and displayed.

Messages

Table 3 shows the messages for AMVLIST.

Table 3. AMVLIST messages

Message number	Message
AMV8807E	No catalog name supplied.
AMV8808E	No entries in the catalog (or the catalog is not accessible).
AMV8809E	Unexpected catalog error <i>rc</i> . The list might be incomplete.
AMV8810E	Error <i>rc</i> closing catalog.
AMV8811E	No entries returned for volume search.
AMV8812E	No records passed filters.
AMV8815E	Cursor is not within a valid sort field.
AMV8816E	Cursor is not on a file line.
AMV8820E	No granule name passed to expand.
AMV8821E	Error expanding granule file.
AMV8827E	File too large. Out of storage.

Return codes

Table 4 shows the return codes for AMVLIST.

Table 4. AMVLIST return codes

Return code	Description
0	Finished correctly.

Table 4. AMVLIST return codes (continued)

Return code	Description
4	No entries in the catalog (or the catalog is not accessible).
12	No catalog name supplied.

HELP

Issue the HELP command to access Archive Manager help topics.



Figure 3. HELP command syntax

Authorization

All users can issue the HELP command to access help.

Operands

cmd (Optional.) The command for which you want to display help.

Examples

Figure 4 shows how to use the HELP command to access the Archive Manager help menu:

```
HELP AAMV
```

Figure 4. Accessing the help menu

Figure 5 shows how to access help for the Archive Manager ARCHIVE command:

```
HELP AAMV ARCHIVE
```

Figure 5. Accessing help for a specific Archive Manager command

QUERY CLASS

The QUERY CLASS command displays information about a specific storage class. The command response includes class name, media type, simplex or duplex status, retention period information, storage media, the filename of the ROSTER file that identifies the users that are permitted to use the storage class, and the class attribute flags.



Figure 6. QUERY CLASS command syntax

Authorization

Administrator and general users.

Operands

storageclass

A valid 1-8 character storage class name.

Command response format

The command response includes the following information about the specified storage class.

Classname

The storage class name.

Media type

The storage class type (EDF, SFS, or TAPE).

Plex state

SIMPLEX (single-copy) or DUPLEX (dual-copy).

Expire min

The minimum retention period (in days).

Expire def

The default retention period (in days).

Expire max

The maximum retention period (in days).

Primary

The virtual device (minidisk), directory path (SFS), or tape pool (tape) allocated for primary archive parcel copy.

Secondary

The virtual device (minidisk), directory path (SFS), or tape pool (tape) allocated for secondary archive parcel copy (only displayed for DUPLEX classes).

Occupancy info

Occupancy information for minidisk or SFS file space. (Not applicable for tape.)

Roster The associated ROSTER file that is used to control which users can access storage class resources.

Flags The storage class flags as defined in the AMVSRVR CONFIG file.

Example (Class name: DEFAULT)

Figure 7 on page 15 shows an example of the QUERY CLASS command with a class name of DEFAULT specified.


```

msg amvarkiv query class default
Ready;
Class name: DEFAULT
Media type: EDF
Plex state: SIMPLEX
Expire min: 1
Expire def: 365
Expire max: 365
Primary: E00
Occupancy: 1097 4K blocks in use; 34903 free; 03 % in use.
Roster: ALLUSERS
Flags: Y Y Y Y N N N
Return code: 0

```

Figure 7. QUERY CLASS example (Class name: DEFAULT)

Example (Class name: SFSDUPL)

Figure 8 shows an example of the QUERY CLASS command with a class name of SFSDUPL specified.

```

msg amvarkiv query class sfsdupl
Ready;
Class name: SFSDUPL
Media type: SFS
Plex state: DUPLEX
Expire min: 1
Expire def: 365
Expire max: 99999999
Primary: ROCKSFS1:AMVARKIV.DUPLEX.PRIMARY
Occupancy: 711 4K blocks in use; 203385 free; 0.00348365475 % in use.
Secondary: ROCKSFS2:AMVARKIV.DUPLEX.SECONDARY
Occupancy: 218 4K blocks in use; 203878 free; 0.0010681248 % in use.
Roster: ALLUSERS
Flags: Y Y Y Y Y N N
Return code: 0

```

Figure 8. QUERY CLASS example (Class name: sfsdupl)

QUERY CLASSES

The QUERY CLASSES command displays a list of all storage classes that are defined in the archive. The command response lists the class name, media type, simplex or duplex status, and retention period minimum, default, and maximum number of days.

►►—QUERY CLASSES—◄◄

Figure 9. QUERY CLASSES command syntax

Authorization

Administrator or general user.

Operands

The QUERY CLASSES command has no operands.

Command response format

Each line of text in the command response is displayed in the following format:

```
classname classtype plextype min_ret/default_ret/max_ret
```

Where:

Classname

The storage class name.

Classtype

The storage class type (EDF, SFS, or TAPE).

Plextype

SIMPLEX (single-copy) or DUPLEX (dual-copy).

Min_ret

The minimum retention period (in days).

Default_ret

The default retention period (in days).

Max_ret

The maximum retention period (in days).

Example

Figure 10 shows an example of the QUERY CLASSES command.

```
msg amvarkiv query classes
Ready;
DEFAULT EDF SIMPLEX 1/365/365
CUSTOM EDF DUPLEX 1/365/99999999
TAPE1 TAPE SIMPLEX 1/365/365
TAPE2 TAPE DUPLEX 1/365/365
SFSSIMPL SFS SIMPLEX 1/365/365
SFSDUPL SFS DUPLEX 1/365/99999999
Return code: 0
```

Figure 10. QUERY CLASSES example

QUERY EXPIRE

The QUERY EXPIRE command displays the expiration date that is in effect for a specific archive data parcel.

```
►►—QUERY EXPIRE—parcel_id—◄◄
```

Figure 11. QUERY EXPIRE command syntax

Authorization

- Administrators can issue the QUERY EXPIRE command to display the expiration information for any data parcel in the archive.
- General users can issue the QUERY EXPIRE command to display expiration information only for the parcels they own.

Operands

parcel_id

A valid parcel identifier (1-8 hexadecimal digits).

Example

Figure 12 shows an example of the QUERY EXPIRE command.

```
smsg amvarkiv query parcel 132
Ready;
*** Parcel 00000132 is owned by user "USER1"
*** 4K Blocks: 24
*** Storage group: ALLUSERS
*** Storage class: TAPE2
*** Image type is: SFS
*** Source media: VMDEVU
*** Expires after: 26 Aug 2011
Return code: 0
```

Figure 12. QUERY EXPIRE example

QUERY PARCEL

The QUERY PARCEL command displays information about a specific archive data parcel.

►►—QUERY PARCEL—*parcel_id*—*query_arg*—◄◄

Figure 13. QUERY PARCEL command syntax

Authorization

- Administrators can query any parcel in the archive.
- General users can query any parcel that they own.

Operands

parcel_id

The 1–8 character parcel identifier.

query_arg

(Optional.) Specify one of the following options:

DETAILS

List detailed information about archive parcel contents. In addition to basic parcel attributes, display information about the original minidisk or SFS directory from which data were archived, the date and time of archive (and for CMS files, information about the individual files).

ELEMENTS

List detailed information about parcel metadata files that are stored in the archive catalog.

MD5 Display the MD5 checksum information for the archive parcel.

MESSAGES

(Short form: MSGS.) Display the messages that are generated by the data packaging tools during archive creation.

Note: If you do not specify any of the *query_arg* options (the default), basic parcel information are displayed.

Example (QUERY PARCEL)

Figure 14 shows an example of the QUERY PARCEL command with no *query_arg* options specified.

```
smsg amvarkiv query parcel 132
Ready;
*** Parcel 00000132 is owned by user "ABCUSER"
***   4K Blocks: 24
*** Storage group: ALLUSERS
*** Storage class: TAPE2
*** Image type is: SFS
*** Source media: VMDEVU
*** Expires after: 26 Aug 2011
Return code: 0
```

Figure 14. QUERY PARCEL command (default; no *query_arg* options are specified)

Example (QUERY PARCEL with MD5 option)

Figure 15 shows an example of the QUERY PARCEL command with the MD5 option specified.

```
smsg amvarkiv query parcel 132 md5
Ready;
*** Parcel 00000132 is owned by user "ABCUSER"
***   4K Blocks: 24
*** Storage group: ALLUSERS
*** Storage class: TAPE2
*** Image type is: SFS
*** Source media: VMDEVU
*** Expires after: 26 Aug 2011

*** MD5 signature:

* 00000132 PARCEL   T2 10/08/26 15:51:45 b816 5487 934b 8992 08c0 e251 a022 8490
Return code: 0
```

Figure 15. QUERY PARCEL command (MD5 option specified)

Example (QUERY PARCEL with ELEMENTS option)

Figure 16 on page 19 shows an example of the QUERY PARCEL command with the ELEMENTS option specified.

```

msg amvarkiv query parcel 132 elements
Ready;
*** Parcel 00000132 is owned by user "ABCUSER"
***      4K Blocks: 24
*** Storage group: ALLUSERS
*** Storage class: TAPE2
*** Image type is: SFS
*** Source media: VMDEVU
*** Expires after: 26 Aug 2011

*** There are 3 elements cataloged:

Element 1 "00000132 CATALOG Z2 V      528      14      2  8/26/10 15:54:13"
Element 2 "00000132 MD5 Z1 V          80        1      1  8/26/10 15:51:45"
Element 3 "00000132 MESSAGES Z2 V    152       16      1  8/26/10 15:51:45"
Return code: 0

```

Figure 16. QUERY PARCEL command (ELEMENTS option specified)

Example (QUERY PARCEL with MESSAGES option)

Figure 17 shows an example of the QUERY PARCEL command with the MESSAGES option specified.

```

msg amvarkiv query parcel 132 messages
Ready;
*** Parcel 00000132 is owned by user "ABCUSER"
***      4K Blocks: 24
*** Storage group: ALLUSERS
*** Storage class: TAPE2
*** Image type is: SFS
*** Source media: VMDEVU
*** Expires after: 26 Aug 2011

*** Image creation messages:

BKRED08104I Output handler CMSFILE initializing...
BKRED08076I Job name is: 00000132
BKRED08105I Output is directed to file 00000132 PARCEL T 2.
BKRCDG8200I New file 00000132 CATALOG T 2 created to contain catalog granule contents.
BKRSFD8013I Base file: VMDEVU:ABCUSER.UTILITIES/SPELL EXEC
BKRSFD8013I Base file: VMDEVU:ABCUSER.UTILITIES/SPELLDEFEXEC
BKRSFD8013I Base file: VMDEVU:ABCUSER.UTILITIES/SPELLFIXEXEC
BKRSFD8013I Base file: VMDEVU:ABCUSER.UTILITIES/SPELLLEXEXEC
VMDEVU ABCUSER SFS 00002 00600000 00312237 090% 00000004 00000000 00000000 00000000 0000 0000 00000154 15:51:44 08/26/10 15:51:45

Return code: 0

```

Figure 17. QUERY PARCEL command (MESSAGES option specified)

Example (QUERY PARCEL with DETAILS option)

Figure 18 on page 20 shows an example of the QUERY PARCEL command with the DETAILS option specified.

```

msg amvarkiv query parcel 132 details
Ready;
*** Parcel 00000132 is owned by user "ABCUSER"
*** 4K Blocks: 24
*** Storage group: ALLUSERS
*** Storage class: TAPE2
*** Image type is: SFS
*** Source media: VMDEVU
*** Expires after: 26 Aug 2011
*** Parcel summary:
***
*** Archive Time: 20:51:44.839991 on Thursday, August 26, 2010 (UTC)
*** Present Loc: Tape file 17 on volume P20095

*** Image type is CMS SFS Filespace:
***
*** Pool:Owner = VMDEVU:ABCUSER
*** Storage group: 2; 312237 blocks of 600000 in use.
*** Warning threshold: 90%

Filename Filetype yy/mm/dd hh:mn:ss Format Records Blocks Directory
SPELL EXEC 95/11/28 16:46:40 F/1024 7 2 VMDEVU:ABCUSER.UTILITIES
SPELLDEF EXEC 87/05/11 16:56:00 V/62 49 1 VMDEVU:ABCUSER.UTILITIES
SPELLFIX EXEC 87/05/14 11:52:20 V/55 56 1 VMDEVU:ABCUSER.UTILITIES
SPELLLEX EXEC 87/05/11 16:55:56 V/57 70 1 VMDEVU:ABCUSER.UTILITIES
Return code: 0

```

Figure 18. QUERY PARCEL command (DETAILS option specified)

QUERY PARCELS

Issue the QUERY PARCELS command to obtain information about archived data. The command response lists parcel IDs, owning user, storage class, number of 4K data blocks, and the parcel expiration date.

►► QUERY PARCELS ◀◀

Figure 19. QUERY PARCELS command syntax

Authorization

- Administrators can display information for all parcels in the archive.
- General users can display information for parcels that they own.

Note: For those users with many parcels in the archive, the command response could potentially be verbose.

Operands

The QUERY PARCELS command has no operands.

Example

Figure 20 on page 21 shows an example of the QUERY PARCELS command.

```

smsg amvarkiv query parcels
Ready;
*** Parcel 000000EC: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 6 Mar 2011.
*** Parcel 000000E0: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 24 Feb 2011.
*** Parcel 000000E1: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 24 Feb 2011.
*** Parcel 000000E2: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 24 Feb 2011.
*** Parcel 000000E3: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 2 Mar 2011.
*** Parcel 000000E4: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 2 Mar 2011.
*** Parcel 000000E5: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 2 Mar 2011.
*** Parcel 000000E6: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 2 Mar 2011.
*** Parcel 000000E7: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 2 Mar 2011.
*** Parcel 000000E8: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 3 Mar 2011.
*** Parcel 000000E9: Owner SAMPUSER, Storage class DEFAULT, 10 4K Blocks, Expires 3 Mar 2011.
*** Parcel 000000FF: Owner SAMPUSER, Storage class DEFAULT, 8 4K Blocks, Expires 7 Apr 2011.
*** Parcel 000000F6: Owner SAMPUSER, Storage class TAPE1, 261 4K Blocks, Expires 6 Apr 2011.
*** Parcel 000000F8: Owner SAMPUSER, Storage class TAPE1, 261 4K Blocks, Expires 6 Apr 2011.
*** Parcel 000000F9: Owner SAMPUSER, Storage class TAPE1, 263 4K Blocks, Expires 6 Apr 2011.
*** Parcel 00000100: Owner SAMPUSER, Storage class DEFAULT, 217 4K Blocks, Expires 7 Apr 2011.
*** Parcel 00000101: Owner SAMPUSER, Storage class DEFAULT, 217 4K Blocks, Expires 7 Apr 2011.
*** Parcel 00000102: Owner SAMPUSER, Storage class DEFAULT, 217 4K Blocks, Expires 7 Apr 2011.
*** Parcel 00000104: Owner SAMPUSER, Storage class SFSSIMPL, 217 4K Blocks, Expires 7 Apr 2011.
*** Parcel 0000011E: Owner SAMPUSER, Storage class TAPE1, 1805 4K Blocks, Expires 12 Aug 2011.
*** Parcel 0000011F: Owner SAMPUSER, Storage class TAPE1, 217 4K Blocks, Expires 12 Aug 2011.
*** Parcel 00000110: Owner SAMPUSER, Storage class DEFAULT, 8 4K Blocks, Expires 11 Aug 2011.
*** Parcel 00000111: Owner SAMPUSER, Storage class DEFAULT, 8 4K Blocks, Expires 11 Aug 2011.
*** Parcel 00000112: Owner SAMPUSER, Storage class DEFAULT, 8 4K Blocks, Expires 11 Aug 2011.
*** Parcel 00000113: Owner SAMPUSER, Storage class DEFAULT, 8 4K Blocks, Expires 11 Aug 2011.
*** Parcel 00000120: Owner SAMPUSER, Storage class SFSSIMPL, 26 4K Blocks, Expires 12 Aug 2011.
*** Parcel 00000121: Owner SAMPUSER, Storage class DEFAULT, 8 4K Blocks, Expires 13 Aug 2011.
*** Parcel 00000122: Owner SAMPUSER, Storage class CUSTOM, 8 4K Blocks, Expires 13 Aug 2011.
*** Parcel 00000123: Owner SAMPUSER, Storage class SFSSIMPL, 217 4K Blocks, Expires 13 Aug 2011.
*** Parcel 00000124: Owner SAMPUSER, Storage class SFSDUPL, 217 4K Blocks, Expires 13 Aug 2011.
*** Parcel 00000125: Owner SAMPUSER, Storage class TAPE1, 3198 4K Blocks, Expires 13 Aug 2011.
*** Parcel 00000126: Owner SAMPUSER, Storage class DEFAULT, 8 4K Blocks, Expires 20 Aug 2011.
*** Parcel 00000127: Owner SAMPUSER, Storage class DEFAULT, 2 4K Blocks, Expires 20 Aug 2011.
*** Parcel 00000132: Owner SAMPUSER, Storage class TAPE2, 24 4K Blocks, Expires 26 Aug 2011.
Return code: 0

```

Figure 20. QUERY PARCELS example

QUERY QUOTA

The QUERY QUOTA command displays archive quota information.



Figure 21. QUERY QUOTA command syntax

Authorization

System administrator or general user as noted.

Options

CLASS *storageclass*

(Administrator or general user.) Display class-specific storage quotas, if defined. Administrators can display quotas for any storage class. General users can display quotas for storage classes that are affiliated with their user ID only.

DEFAULT

(Administrator or general user.) Displays the default system-wide quota limit (if defined).

USER *user_id*

Display the number of 4K blocks that are occupied by content that is owned by *user_id* and one of the following quotas:

- The user-specific quota
- The system default quota (as it applies to the user in question)

If there is a user-specific quota, it is displayed. If there is no user-specific quota, then the storage class or system-wide default quota is displayed.

Administrators can display information for any user of the archive. General users are permitted to display quota information for their own user ID only.

Example QUERY QUOTA USER (no user-specific quota defined)

Figure 22 shows an example of the QUERY QUOTA USER command. In the example, the user (USER1) is subject to the system-wide DEFAULT quota only. No user-specific quota was defined for the user.

```
cp smsg amvarkiv query quota
Ready;
User USER1 has used 15054 4K blocks in the archive.
Return code: 4 obtaining quota limit for user USER1;
reason "- No user-specific or system-wide default quota for user USER1.
```

Figure 22. QUERY QUOTA USER example

Example QUERY QUOTA USER (user-specific quota defined)

Figure 23 shows an example of the QUERY QUOTA USER command. In the example, user (USER2) is subject to an archive-wide limit of 98,765,432 4K data blocks. This limit can be reduced if storage class-specific quotas impose a lower limit for a particular storage class.

```
smsg amvarkiv query quota user user2
Ready;
User USER2 has used 7515 4K blocks in the archive.
User USER2 has a user-specific quota of 98765432 4K blocks.
Return code: 0
```

Figure 23. QUERY QUOTA USER example

Example QUERY QUOTA DEFAULT (system-wide archive quota)

Figure 24 shows an example of the QUERY QUOTA DEFAULT command. In the example, the default system-wide archive quota is displayed.

```
smsg amvarkiv query quota default
Ready;
Return code: 0 System default quota is 199999998 4K blocks.
```

Figure 24. QUERY QUOTA DEFAULT example (system-wide archive quota)

Example QUERY QUOTA CLASS (class-specific quota)

Figure 25 on page 23 shows an example of the QUERY QUOTA DEFAULT command. In the example, a class-specific quota for the DEFAULT storage class is

set to 99,999,999 4K blocks.

```
msg amvarkiv query quota class default
Ready;
Return code: 0 Class-level quota for class DEFAULT is 99999999 4K blocks
```

Figure 25. QUERY QUOTA DEFAULT example

Example QUERY QUOTA CLASS (no class-specific quota defined)

Figure 26 shows an example of the QUERY QUOTA CLASS command. In this example, the class CUSTOM has no class-specific quota defined. User-specific quotas are in effect if defined for a particular user (otherwise, the system-wide default quota is enforced).

```
msg amvarkiv query quota class custom
Ready;
Return code: 4 No class-specific quota for storage class CUSTOM.
```

Figure 26. QUERY QUOTA CLASS example

QUERY TAG

The QUERY TAG command displays tag information that is associated with a specific archive data parcel.

►►—QUERY TAG—*parcel_id*—◄◄

Figure 27. QUERY TAG command syntax

Authorization

- Administrators can issue the QUERY TAG command to display tag information for any archive data parcel.
- General users can issue the QUERY TAG command to display tag information for parcels they own.

Note: The QUERY TAG command is not enabled for use through the service machine console.

Operands

parcel_id

The 1- 8 byte parcel identifier.

Example (parcel with no tag data)

Figure 28 on page 24 shows an example of the QUERY TAG command. The parcel does not have tag data.

```

smsg amvarkiv query tag 132
Ready;
Return code: 4 No tag data for parcel 00000132.

```

Figure 28. QUERY TAG example (parcel with no tag data)

Example (parcel with tag data)

Figure 29 shows an example of the QUERY TAG command. The parcel has tag data that was set using the SET TAG command or the TAG option of the ARCHIVE command.

```

smsg amvarkiv query tag 133
Ready;
Return code: 0 Parcel 00000133 tag: "THIS IS TEST TAGTEXT"

```

Figure 29. QUERY TAG example (parcel with tag data)

RECALL

The RECALL command recalls archived client data (files or disk images) stored in the archive server.

Retrieve previously archived data using the RECALL command, as follows:

- Image data that is originally archived from CKD can be recalled to CKD.
- Files that are originally archived from EDF minidisk can be recalled to EDF, RDR, or SFS.
- Files that are originally archived from SFS can be recalled to SFS, EDF, or RDR.

Note: Only base files (SFS objects that represent regular CMS files and not directories, aliases, or permissions) can be recalled to non-SFS destinations.

```

▶▶—RECALL—parcel_id—TO—| Destinationspec |—————▶▶

```

Destinationspec:

```

|—CKD—username vdev—————|
|—EDF—username vdev regexp——|
|—SFS—poolname username regexp—|
|—RDR—username nodename regexp—|

```

Figure 30. RECALL command syntax

Authorization

- Administrators can recall content from any parcel.
- General users can recall content from any parcel that they own.

Note: Archive Manager uses the RECALL command internally to process recall requests that are submitted through the panel interface (AMVLIST). It can also be issued directly through CP SMSG.

Operands

parcel_id

The parcel to recall (specify the 1 - 8 character parcel identifier).

Destinationspec

Specify one of the following syntax options to indicate the destination of the parcel to recall from the archive.

Recall to RDR (RDR)

Username

The destination user ID.

Nodename

The destination RSCS or NJE node or '-'.

Regexp

A regular expression; the form varies according to the source data type.

Notes:

1. When the source data type is CMS MINIDISK (EDF), *regexp* requires three tokens. For example:
 - Specifying * * * recalls all files.
 - Specifying * EXEC 2 recalls all files with a file type of EXEC and file mode number of 2.
2. When the source data type is CMS SHARED FILE SYSTEM (SFS), *regexp* requires four tokens. For example:
 - Specifying * * * * recalls all files from all directories in the archive image.
 - Specifying **pathname** * ASSEMBLE 2 recalls all files where the SFS directory includes the string *pathname*, with a file type of ASSEMBLE and a file mode number of 2.
3. When the source data type is CKD DASD IMAGE (CKD), omit *regexp* because CKD images can be recalled to a CKD destination only.

Recall to CMS format minidisk (EDF)

Username

The user ID that owns the destination minidisk.

Vdev The minidisk address of the target minidisk.

Regexp

A regular expression. The form varies according to the source data type.

Recall to SFS file space (SFS)

Username

The destination SFS file space or user name.

Poolname

The SFS file pool name for the destination file space.

Regexp

A regular expression. The form varies according to the source data type.

Recall CKD image to CKD minidisk extent (CKD)

Username

The user ID that owns the destination minidisk.

Vdev The minidisk address of the target minidisk.

Regexp

A regular expression. The form varies according to the source data type.

Example (RECALL command issued through CP SMSG)

Although the RECALL command is intended to be issued from the panel interface, you can also issue it through CP SMSG. Figure 31 shows an example of the RECALL command that is issued through CP SMSG. In the example, the RECALL command recalls data from parcel 127, sending the results as spool files (RDR) to user SAMPUSER on the local system, selecting ("* * * *") any file name, file type, file mode, and directory path from the parcel.

```
smsg amvarkiv recall 127 to rdr sampuser - * * * *
Ready;
RDR FILE 7756 SENT FROM AMVARKIV PUN WAS 0074 RECS 0009 CPY 001 R NOHOLD NOKEEP
Return code: 0, Recall successfully initiated.

query rdr * 7756 all
ORIGINID FILE CLASS RECORDS CPY HOLD DATE TIME NAME TYPE DIST
AMVARKIV 7756 R PUN 00000009 001 NONE 08/31 16:44:24 SFSDIR INDEX *ARCHIV*
Ready;
```

Figure 31. RECALL example

REMOVE

The REMOVE command removes a data parcel from the archive.

```
►►—REMOVE—parcel_id—◄◄
```

Figure 32. REMOVE command syntax

Authorization

- Administrators can delete any parcel from the archive.
- General users can delete parcels that they own from archive, if the storage class configuration permits use of the REMOVE command.

Note: The REMOVE command is not enabled for use through the service machine console.

Operands

parcel_id

The 1–8 byte parcel identifier.

Example

Figure 33 on page 27 shows how to use the REMOVE command to remove parcel ID 107:

```
msg amvarkiv remove 107
Ready;
Return code: 0
```

Figure 33. REMOVE example

SET EXPIRE

The SET EXPIRE command sets or modifies, the expiration date that is associated with an archive data parcel.

►►—SET EXPIRE—*parcel_id*—*days*—◄◄

Figure 34. SET EXPIRE command syntax

Authorization

- Administrators can issue the SET EXPIRE command to set, or modify, the retention period for any data parcel.
- General users can issue the SET EXPIRE command to set, or modify, the retention period for any data parcel that they own (if the storage class definition allows renewal of expiration dates).

Operands

parcel_id

The 1–8 byte parcel identifier.

days The number of days where *days* can be specified as:

- An integer that specifies the number of days until a parcel expires.
- A relative value that modifies the existing expiration date. This value is specified as an integer that represents the number of days that are prefixed by "+" or "-". Specify *+days* to extend the expiration date of a parcel. Specify *-days* to shorten the retention period.

In all cases, the value of the expiration date is constrained according to the minimum and maximum retention period in the associated storage class definition.

Example

Figure 35 on page 28 shows an example of the SET EXPIRE command where the expiration date is extended by seven days.

```

msg amvarkiv query parcel 2
Ready;
**** Parcel 00000002 is owned by user "AMVADMIN"
***      4K Blocks: 2588
*** Storage group: ALLUSERS
*** Storage class: TAPE1
*** Image type is: EDF
*** Source media: 191
*** Expires after: 16 Aug 2006
Return code: 0
*

msg amvarkiv set expire 2 +7
Ready;
*Expiration for parcel 00000002 changed from 16 Aug 2006 to 23 Aug 2006.
Return code: 0
*

```

Figure 35. SET EXPIRE example

SET TAG

The SET TAG command specifies tag data to associate with the parcel specified by *parcel_id*. (Use the QUERY TAG command to view data that is specified by the SET TAG command.)

►►—SET TAG—*parcel_id*—*tag_text*—◄◄

Figure 36. SET TAG command syntax

Authorization

- System administrator. Specify tag data (*tag_text*) for any parcel in the archive that is specified by *parcel_id*.
- General user. Specify tag data for the parcel that is specified by *parcel_id*. (The parcel must be owned by the user that issues the SET TAG command.)

Note: The SET TAG command is not enabled for use through the service machine console.

Operands

parcel_id

The 1–8 byte parcel identifier.

tag_text

A text string (maximum length: 120 bytes). Archive Manager converts all text to uppercase.

Example

Figure 37 on page 29 shows an example of the SET TAG command.

```
msg amvarkiv set tag 2 Snapshot of my a-disk from 23 august 2005
Ready;
*Return code: 0 Tag updated*.

msg amvarkiv query tag 2
Ready;

*Return code: 0 Parcel 00000002 tag: "SNAPSHOT OF MY A-DISK FROM 23
AUGUST 2005"
*
```

Figure 37. SET TAG example

Notices

This information was developed for products and services offered in the U.S.A. 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 give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

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 states 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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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.

All 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 the names and addresses used by an actual business enterprise 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. 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 IBM's application programming interfaces.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)[®] are trademarks or registered trademarks of International Business Machines Corp., 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>.

Adobe, the Adobe logo, PostScript and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

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

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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

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

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

Bibliography

You might need to refer to other sources of information when you are using Archive Manager.

This section lists the documentation that supports Archive Manager. Use the appropriate library for the version of z/VM that you are using.

Tip: To quickly locate a specific book, use the IBM Publications Center, which is located at www.elink.ibm.link.ibm.com/public/applications/publications/cgibin/pbi.cgi. After you enter your country information, click **Search for publications**, and enter the publication number (xxxx-xxxx) of the book that you want.

z/VM titles cited in this book

- *z/VM REXX/VM Reference*, SC24-6221
- *z/VM REXX/VM User's Guide*, SC24-6222
- *z/VM CMS User's Guide*, SC24-6173
- *z/VM CMS Commands and Utilities Reference*, SC24-6166
- *z/VM CP Commands and Utilities Reference*, SC24-6175
- *CMS File Pool Planning, Administration, and Operation*, SC24-6167
- *Archive Manager for z/VM Administration Guide*, SC18-9345

Index

A

- ADMIN authority
 - SFS 5
- AMVARKIV 2
- AMVLIST command 9
- archive
 - CKD 5
 - EDF 5
 - SFS directory 5
- ARCHIVE command 7
- archive data
 - to staging disk 6
- Archive Manager
 - commands 7
 - features 2
 - overview 1, 2
 - terminology 2
- archive quota
 - displaying information 21
- archived client data
 - recalling 24
- archived data
 - recalling 5
- archiving
 - data 5

B

- benefits of archiving 2

C

- changes for this edition 1
- CKD 2
- CKD archive 5
- CMS
 - files 2
- CMS EDF minidisk 2
- CMS SFS 2
- commands
 - AMVLIST 9
 - ARCHIVE 7
 - Archive Manager 7
 - HELP 13
 - QUERY CLASS 13
 - QUERY CLASSES 15
 - QUERY EXPIRE 16
 - QUERY PARCEL 17
 - QUERY PARCELS 20
 - QUERY QUOTA 21
 - QUERY TAG 23
 - RECALL 24
 - REMOVE 26
 - SET EXPIRE 27
 - SET TAG 28
- comments
 - methods for providing vi
- Count Key Data 2

D

- data
 - archiving 5
 - supported types 2
- data parcel
 - displaying information 17, 20
 - removing 26

E

- EDF archive 5
- expiration date
 - displaying 16
 - setting 27

F

- features
 - Archive Manager 2
- feedback
 - methods for providing vi
- files
 - CMS 2

H

- help
 - accessing 13
- HELP command 13

L

- legal notices
 - notices 31
 - trademarks 33
- Linux 2

N

- notices 31

Q

- QUERY CLASS command 13
- QUERY CLASSES command 15
- QUERY EXPIRE command 16
- QUERY PARCEL command 17
- QUERY PARCELS command 20
- QUERY QUOTA command 21
- QUERY TAG command 23

R

- RECALL command 24
- recalling
 - archived client data 24
 - archived data 5

REMOVE command 26
ROSTER file 13

S

SET EXPIRE command 27
SET TAG command 28
SFS 6

- ADMIN authority 5

SFS directory archive 5
Shared File Systems 2
staging disk

- archiving data to 6

storage classes

- displaying 15
- displaying information about 13

storage media

- supported 2

syntax diagrams

- how to read v

System z 2

T

tag

- displaying information 23
- setting 28

terminology

- Archive Manager 2

trademarks 33

Z

z/VSE volumes 2



Product Number: 5697-J05

Printed in USA

SC18-9345-02

