IBM Backup, Archive and Retrieval System/VM User's Guide Release 1.3

Document Number SB11-8972-03

October 6, 1997 - 4:59 p.m.

Steve Garraway
Gerhard Widmayer

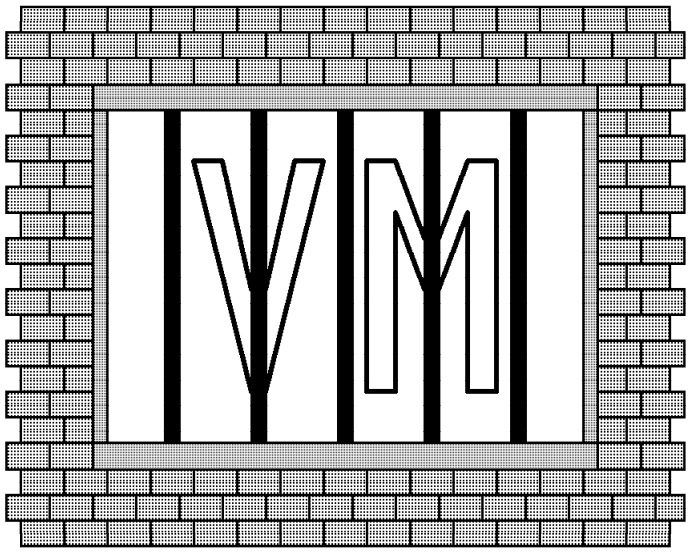
GARRAWAY at WINVMB
WIDMAYER at STUTVM1

SB11-8972-03

IBM Backup, Archive and Retrieval System/VM

User's Guide

Release 1.3



IBM Backup, Archive and Retrieval System/VM

SB11-8972-03

User's Guide

Release 1.3

Note!

Before using this information and the product it supports, be sure to read the general information under “Notices” on page v.

Fourth Edition (July 1995)

This edition, SB11-8972-03, applies to Release 1.3 of IBM Backup, Archive and Retrieval System/VM (BARS/VM), Program Number 5785-DJD, and to all subsequent releases and modifications until otherwise indicated in new editions.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

A form for readers’ comments appears at the back of this publication. If the form has been removed, address your comments to:

GARRAWAY at WINVMB
WIDMAYER at STUTVM1

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Notices v
Trademarks and Service Marks v

Preface vii

Chapter 1. Introduction 1

Chapter 2. Using the BACKUP Function 3
2.1 Incremental Backup and Retrieval 3
2.2 BPUT and BDELETE 4
2.3 RESTORE 4

Chapter 3. Using the ARCHIVE Function 5
3.1 Deferred Processing and APUT/AGET Completion 5
3.2 User IDs and Defining 7
3.3 Special ARCHIVE Commands 7
3.4 Archive Levels, Classes, Tags and Retention Dates 8
3.5 DQUERY and DRECOVER 9

Chapter 4. User Commands Description 11
4.1 Issuing BARS/VM Commands 11
4.1.1 Generic Specification 11
4.1.2 SFS Directory Specification 11
4.1.3 How to Read a Syntax Diagram 12
4.1.4 User Commands Summary 13
4.1.5 How to Authorize Users 14
4.1.6 Return and Reason Codes 14
4.1.7 Dates 14
4.2 Command Syntax Description 15
4.2.1 AALTER 15
4.2.2 ADEFINE 16
4.2.3 ADELETE 17
4.2.4 AGET 18
4.2.5 APUT 19
4.2.6 AQUERY 21
4.2.7 ARENAME 22
4.2.8 BACKUP 23
4.2.9 BDELETE 24
4.2.10 BLAH 25
4.2.11 BPUT 26
4.2.12 CQUEUE 29
4.2.13 DBR 29
4.2.14 DISPLAY 30
4.2.15 DQUERY 31
4.2.16 DRECOVER 32
4.2.17 FILELIST 33
4.2.18 HELP 34
4.2.19 LPUT 34
4.2.20 QUERY 35
4.2.21 RESTORE 37
4.2.22 RETRIEVE 38

4.3 Sample BARS/VM Commands 41
4.4 Authorizing other users 45
4.4.1 OWNED authorization 46
4.4.2 ESM authorization 46

Chapter 5. A Sample Communication with BARS/VM 49

Index 55

Notices

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter 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 the IBM Director of Commercial Relations, IBM Corporation, Purchase, NY 10577.

Trademarks and Service Marks

The following terms, denoted by an asterisk (*) in this publication, are trademarks or service marks of IBM Corporation in the United States and/or other countries:

IBM
RACF
VM/ESA

Preface

The objectives of IBM Backup, Archive and Retrieval System/VM are to improve the backup and retrieval of CMS and non-CMS user data for both the installation and users, provide an archive and retrieval facility for CMS files, an archive facility for minidisks, and a disaster backup and recovery facility for minidisks.

The BARS/VM product functions are implemented using certain specific resources (userID, EXEC, MODULE) named 'VMBARS'. In this document, the name VMBARS will be used to refer to these product functions.

BARS/VM is designed to run on VM/ESA* Release 1.1 and subsequent releases. Support for Access Verification Routines (ACI: Access Control Interface) is required unless RACF* (5740-XXH) is installed which provides an Access Control Interface by itself.

Chapter 1. Introduction

The term “backup” is defined as a process of making copies of data by the installation. IBM Backup, Archive and Retrieval System/VM (BARS/VM) provides two types of backup which are complementary - incremental backup and disaster backup.

BARS/VM is designed to automatically back up new and changed CMS files. These files may reside in minidisks and the Shared File System (SFS). BARS/VM also backs up non-CMS minidisks when the user or installation requests it. These backup copies may reside on DASD or tape. The installation may choose one of three methods of backup - to DASD only, to DASD with spill to tape, or directly to tape with no DASD.

BARS/VM provides the user with a QUERY facility which permits the displaying of the current backup data. The user is also furnished with a retrieval capability which obtains the backup copies of data. BARS/VM also provides authorized users with the capability of restoring minidisks, SFS directories, SFS file spaces, SFS storage groups, and complete real volumes from the last backup.

The term “disaster backup” is defined as a process of making copies of data by the installation to be stored outside the computing center, so that data can be recovered after a disaster where the normal backup data is not useable. BARS/VM is designed to dump, on operator request, data to a disaster backup tape pool. This data may be from the original minidisks, or may be the compacted and inactive BARS/VM backup data. It can back up either the complete installation's minidisks (CMS and non-CMS data) or only just a single minidisk or a logical set of items (defined via the LSET command). There is no BARS/VM function to recover disaster backup data. Recovery is achieved by running a RECOVERY EXEC which does not rely on BARS/VM, and which is automatically dumped along with the data.

The archive facilities of BARS/VM provide a means for users to save seldom used CMS files and free the space they occupy. Commands are provided to allow users to send files to the archive, retrieve from the archive and query archived files. Archived files reside on DASD or tape. The installation may choose one of three methods of storing archived files. They may be placed on DASD only, on DASD with spill to tape, or they may go directly to tape with no DASD.

HELP is provided for all commands and messages.

The incremental backup, disaster backup, retrieval and archive processes operate under the control of parameters specified by the installation.

Chapter 2. Using the BACKUP Function

2.1 Incremental Backup and Retrieval

This chapter is intended to provide information about the backup facilities of BARS/VM for the general user.

Backup is defined as the making of duplicate copies of CMS files, on CMS minidisks and Shared File System (SFS) directories, and non-CMS minidisks by the installation. The purpose of BARS/VM backup is to protect users against loss of data because of hardware failure or accidental erasure. The backup is managed automatically by BARS/VM. Retrieval of backed up data is normally the responsibility of the user who owns that data and commands and a full screen menu facility are provided to allow this.

The BARS/VM incremental backup function operates automatically each night. The frequency of backup of a particular minidisk or SFS directory may be set by the owner through the BACKUP command. For CMS data, any frequency up to 99 days may be set. However, all CMS data is backed up by default and normally the user need take no action. The default frequency of backup for CMS data is set by the BARS/VM administrator.

For non-CMS minidisks, the system overhead in carrying out backup is much higher. For this reason the minimum frequency of backup allowed is normally set to a larger number of days than for CMS data. Also non-CMS data is not backed up by default, so a BACKUP command must be issued if backup is required. If an important change has been made to a non-CMS minidisk, the user may specify that backup is to be performed that night by issuing the BACKUP command with the TONIGHT option. Users should ensure that the non-CMS minidisks for which they are responsible are backed up at an interval which satisfies their requirements but minimizes the system resources used. The BACKUP command may be used to change the backup parameters for a minidisk.

BARS/VM maintains a number of backup versions which is set by the administrator. Available backup data may be displayed by using the QUERY command and backup data may be recovered using the RETRIEVE command. If the FILE option of the QUERY command is used, the resulting information is sent as a file called REPORT LIST, otherwise it is displayed as messages on the screen. Retrieval of backup data may be requested either overnight or immediately. A RETRIEVE command for a non-CMS minidisk restores the data directly to the minidisk and an acknowledgement file is sent to the command issuer to confirm the status of the retrieval. A RETRIEVE for CMS files results in the files being sent to the virtual card reader. If one or more files fail to retrieve, a file, REPORT LIST, is sent giving a complete list of the files that were to be retrieved and the status of each one. An example is given below.

```
RETRIEVE OF TOM      DIRECTORY
WINSFSS:TOM.
BARQUERY EXEC      RETRIEVE SUCCESSFUL
BARSTART EXEC      RETRIEVE SUCCESSFUL
BARSTAT  EXEC      RETRIEVE SUCCESSFUL
BARTRANS EXEC      RETRIEVE SUCCESSFUL
BGET     EXEC      RETRIEVE FAILED  RC = 0028
BRAND    EXEC      RETRIEVE SUCCESSFUL
```

This file is not sent if all files are successfully retrieved. The return codes given in REPORT LIST result from errors either loading or copying files, for example,

```
Error loading file from tape                    RC = 3
The file was not found                          RC = 28
The file is too large to process                RC = 255
```

Any other return codes suggest problems with BARS/VM or CMS and the administrator should be notified.

When a file is erased from a minidisk, or when a minidisk is deleted, BARS/VM retains the backup data for an installation-defined period. After this time the backup is erased and retrieval is no longer possible.

2.2 BPUT and BDELETE

CMS files and non-CMS minidisks may be backed up at any time using the BPUT command. Since data is backed up regularly through the incremental backup process there should not be a need to use BPUT very often. However it may be useful after critical changes have been made. Care must be take however since BPUT will create a new level zero backup which will cause one of the old levels to be lost. So there is a danger of losing backup versions created by incremental backup which might be required for recovery.

The BDELETE command is used to delete backup data. Again this is only needed in special circumstances since backup will be automatically erased when it is not required. However there are occasions when immediate deletion is required such as when a minidisk is being transferred to another user and some files on the minidisk and the backup of those files are not to be given to the new user.

2.3 RESTORE

The user can restore a specified Shared File System (SFS) directory, file or alias with the last backup copy through the RESTORE command. When an SFS directory is restored, the directory is created and all files and aliases are restored that existed at the time of the last backup cycle. The sub-directories may also be restored. Directory and file authorizations are regranted unless specified otherwise.

Files and aliases may be restored directly into an existing SFS directory by giving a file specification.

The RESTORE command may also be used to restore a complete minidisk to the state it was in at the last backup.

Chapter 3. Using the ARCHIVE Function

This chapter is intended to provide information about the archive facilities of BARS/VM for the general user.

The purpose of BARS/VM archive is to minimize the use of online storage for CMS files which are not required for the work currently being performed. It provides a way for users to make copies of files on volumes managed by BARS/VM and to be able to retrieve those copies when required. While files are held on archive storage the original versions may be erased, thus making space available for new files.

Files are archived by the APUT command. In this command is specified the cuu of the minidisk or the SFS dirid where the file or files reside (e.g. 191) and the name of the file or files to be archived. More than one file may be archived with a single command by using the * and % notation as described in Chapter 4, “User Commands Description” on page 11. BARS/VM copies each file to a suitable archive minidisk giving it a unique name. This consists of a file name which is the current date and a file type of a sequence number unique to that date. To copy data BARS/VM links and accesses the user's minidisk or accesses the user's SFS directory in read-only mode and copies the requested files in packed format.

Files are retrieved from archive with the AGET command. BARS/VM locates the required file, unpacks it, and sends it to the requester's card reader in NETDATA format. The requester can then load the files using RECEIVE. As for APUT, more than one file may be retrieved with a single command by using the * and % notation as described in Chapter 4, “User Commands Description” on page 11. If the installation uses tape to hold archived files there may be delays during AGET processing while tapes are mounted.

An alternative way of archiving files is available for the situation where BARS/VM would be unable to copy the file, for example when it resides on temporary disk or if the archive is to be performed on a different system using the TONODE parameter on the BARS/VM command. For such cases a single file mode letter may be specified instead of a minidisk cuu. The file is then sent to BARS/VM for archive through DISK DUMP to its card reader.

A full screen interface to all of the archive commands is provided as an alternative to the line commands. An initial panel guides the user to the appropriate command where the possible operands are displayed.

3.1 Deferred Processing and APUT/AGET Completion

The process of archiving files or retrieving them from archive may be carried out immediately on demand or may be deferred to the next normal BACKUP cycle, usually overnight. It is preferable, whenever possible, to defer the process and thus minimize the impact on the VM system during prime shift and so limits are imposed on how many files may be sent to and retrieved from archive immediately. The precise values of these limits are decided by each installation and may be displayed using the DISPLAY OPTIONS command. Deferred archive and retrieval takes place at the beginning of BARS/VM BACKUP cycle.

Deferred commands which have been entered but not yet processed may be displayed by using the DISPLAY DEFERRED command. Should it be required to cancel a deferred AGET or APUT then the CANCEL option of the appropriate command is used.

On completion of an APUT, whether deferred or immediate, the requester is sent a file which describes the result of the APUT. This file has a file type of EXEC and a file name of the user ID under which the files were archived. The file is in CMS EXEC format and may be used to erase the files which were successfully archived. If some files fail to archive they are included in the EXEC but have an asterisk in column 1 and are marked ARCHIVE FAILED. An example of such a file which might be received after the successful completion of an APUT for all EXEC files is given below. This is followed by an example where one of the files failed to archive.

```
&1 FREE     EXEC     &2
&1 PROFILE  EXEC     &2
&1 SCRIP    EXEC     &2
&1 VRDRDEF  EXEC     &2
&1 VUTP     EXEC     &2


&1 PROFILE  EXEC     &2
&1 PROFILE  XEDIT    &2
*  LASTING  GLOBALV    ARCHIVE FAILED  RC =     0013
```

When the EXEC is called, the &1 and &2 are replaced by the first and second arguments. For example, if I receive a file called TOM EXEC, I can erase the files that have been archived by typing

```
EXEC TOM ERASE
```

or if they were on, say, my B disk

```
EXEC TOM ERASE B
```

It is very useful if many files have been archived and is similar to the function provided by the CMS LISTFILE command with the EXEC option. Lines that start with * are ignored and so files that failed to archive would not be erased in the above example. The return codes given in the EXEC file result from errors either receiving or copying the file.

```
   The file has an unknown record format (not F or V)    RC = 1
   Error receiving the file                              RC = 2
   Error copying file to tape                            RC = 3
   No space is available                                 RC = 13
   The file was not found                                RC = 28
   The file has an incorrect format                      RC = 255
   The file is too large to process                      RC = 255
```

Any other return codes suggest problems with BARS/VM or CMS and the administrator should be notified.

If an error occurs within BARS/VM during the execution of a deferred APUT or AGET command the requester is sent a file containing the command which was received and an indication of how to determine the cause of error that has occurred. These explanatory files are called ARCHIVE ERROR and AGET ERROR respectively. Below is an example of an ARCHIVE ERROR file. In this case the file NEW FILE did not exist in the specified Shared File System directory.

```
You issued the following BARS/VM command for deferred execution -
APUT     .    NEW      FILE
( LEVEL    CLASS  A RETAIN 001 TAG            FROM           FOR
This command failed. To obtain further information issue following command -
HELP MESSAGE HDF0338I
```

3.2 User IDs and Defining

Archived CMS files are associated with a particular **user ID** which is a one to 8 character name which identifies the owner of the archived files. This will normally be the same as the VM logon user ID of the person who archived the data. However this need not be so and, in fact, the user ID need not be an actual VM user ID. It is recognized that frequently files from different minidisks of different VM users may be required to be archived and associated with each other and under a single name thus creating what is sometimes called a **pool** of data. So the name under which the data is held is sometimes called a pool user ID or name.

Users may always archive data under their own VM user IDs but to archive data to such a pool they must be authorized to do so by someone who has special BARS/VM authority, normally the person who administers the users minidisks.

Because user IDs or pool names may be retained when persons move to different groups or departments, BARS/VM requires extra definition of the owners of archived data. Before data can be archived under a user ID or pool name, the ADEFINE command must be issued to give BARS/VM the name and also a department and account identifier to associate with that name. The account identifier is included to allow an installation to charge for archive space. BARS/VM saves the information from the ADEFINE command but does not verify it in any way. It is left to the installation to decide what to do with data associated with invalid departments or account identifiers.

The is a special consideration in a shared system (SSI) environment. In such a BARS/VM environment there may be the same user ID on several systems with the same owner or with different owners or different work. The ADEFINE command, through the SHARED and NONSHARED options allows the user to define whether there is to be a single archive or a separate archive on each system. If there is a single archive, all archive commands are routed to the node where the first ADEFINE was processed. All other ADEFINEs affect only that single archive in a BARS/VM coupled system. That should be the normal way of working for most users who don't know where they sign on and who have access to their minidisks on all systems. However ADEFINE also allows different sets of archive in such a complex. In this case only the local user ID gains access to the archive data.

3.3 Special ARCHIVE Commands

The ADEFINE, AGET and APUT commands have already been introduced in the discussions above. Once a user ID has been defined to BARS/VM, files may be archived under that user ID name and retrieved when required. Note that when an archive copy is made of a file it is not erased from the source minidisk, nor is the archive copy erased when an AGET is performed. To erase archived files the ADELETE command may be used. All file levels or only specified levels may be deleted. If an ADELETE command is used to delete all archived files for a user

ID, then an ADEFINE must be issued before files can be archived again for that user ID.

The files archived for a particular user ID or pool name may be listed by the AQUERY command. Since there may be many files it is normal to limit this display in some way. A single file may be displayed or the * and % notation may be used. Additional keywords on the AQUERY command also limit the display. For example, only files with a particular tag may be listed. See 3.4, “Archive Levels, Classes, Tags and Retention Dates” for a discussion of tags.

The AALTER command may be used to alter information associated with already archived files. The retention period may be changed as may the class and tag. See 3.4, “Archive Levels, Classes, Tags and Retention Dates” for a discussion of classes and tags. The ARENAME command can be used to rename archived files.

If a command has been sent to BARS/VM, the status of the BARS/VM system may be displayed to determine whether the command is being processed or is currently queued awaiting processing. The command DISPLAY STATUS QUEUES will display the queue of commands waiting for processing and DISPLAY STATUS USERS will display the status of each of the BARS/VM virtual machines and the work they are performing.

3.4 Archive Levels, Classes, Tags and Retention Dates

A facility which is important for the user of the archive system to understand is that of **levels** of archived files.

A user may create several levels of archive for the same file name, up to a maximum defined by the installation. These levels are defined as 0, 1, etc.. The user is allowed initially a number of levels corresponding to the default for the installation. The permitted number of levels may then be changed for a specific user by an authorized command issued by the BARS/VM administrator. If a user does an APUT without specifying LEVEL, the archived file is assigned a level of 0. Another APUT without LEVEL would make the previously archived file level 1 and the new one level 0. If LEVEL is specified on the APUT command, the newly archived file is assigned the specified level, unless doing so would create a gap in the levels. Here the lowest available level number smaller than the one specified is assigned.

As an example, APUT X EXEC would create an archived file with a level number of 0 (call it X0). Another APUT would make this archived file level 1, i.e. X0 is now X1, and create a new X0. If the command is now repeated but specifying LEVEL 1 then the level 1 file, X1, is deleted and a new X1 created. The level 0 file, X0, is untouched. If LEVEL 2 is specified, a new level 2 file, X2, is created. If, however, LEVEL 3 were specified instead of LEVEL 2 and since there is no level 2 the newly archived file would be assigned level 2 to avoid gaps in the levels.

An archived file is always assigned a security **class**. This is a single character whose precise meaning is defined by the installation, one particular character meaning UNCLASSIFIED, and so on. Users may specify a class on the APUT or AALTER commands and there is an installation default if it is omitted. The DISPLAY OPTIONS command will display this default value.

Tags may be assigned to archive files to provide additional information or to define logical group of files within the ARCHIVE. The AQUERY command may be instructed to list only those archived files which have a particular tag. The tag is a character string with a length of up to 8 characters. Since files may be archived for long periods of time it may be useful to use the tag mechanism as a aid to recalling the reason for archiving the file. Alternatively some checkpoint may be reached and a set of files archived with the same tag so they are thus related to each other within the archive system.

A **retention date** is always assigned to an archived file. This may be done explicitly when the file is archived. If none is explicitly given an installation default is given. When a file expires there are two alternative possibilities. If the DELEXP option was specified on ADEFINE, a file is erased when it expires. If not, the file is not erased and it is left to the administrators of the archive system to erase archived files. Various tools are available to them to identify and process expired files. If a user chooses not to use DELEXP he should be aware of the installation's policy in this matter.

3.5 DQUERY and DRECOVER

Archived files may be deleted by the user either explicitly by means of ADELETE or implicitly through an APUT which replaces a file. To protect users against accidental erasure of archived files BARS/VM saves information about any deleted file for an installation-defined number of days. During this period an erased file may be recovered by using the DRECOVER command. The DQUERY command may be used to list files which have been deleted from archive but for which information has been retained by BARS/VM to allow recovery. Note that the information is saved at the time the ADELETE or APUT command is processed causing the erasure of the file and so the file may be recovered immediately. The setting of the various installation options may be displayed by using the command DISPLAY OPTIONS.

Chapter 4. User Commands Description

4.1 Issuing BARS/VM Commands

Commands are sent to BARS/VM by the VMBARS EXEC. To issue a command to the local BARS/VM the EXEC name VMBARS is used followed by the command, e.g. VMBARS QUERY 191. A command may also be sent to a remote node by specifying VMBARS TONODE followed by the node name and command, e.g. VMBARS TONODE VMSYS1 QUERY 191. In this case the command is executed at the remote node (VMSYS1) and resulting messages and files are sent back to the issuing node.

You may issue all of the BARS/VM functions described in this chapter by using the BARS/VM full screen interface. This interface provides a selection panel and some input menus to enter the parameters required for a specific function, so there is no need to know the several command formats for the end-user. An ONLINE HELP display assists you in determining the appropriate input parameters.

The interface is called by issuing the VMBARS EXEC without operands.

BARS/VM will only allocate one subuser at a time to process commands for a general user. This prevents one person from using up all the subusers, and thus locking out all other users.

4.1.1 Generic Specification

In some commands a generic name may be given for such things as file names or tags. "*", or "%" may be specified in the name to indicate a set of items, usually files, rather than a single one. The special character "*" indicates that any number of any characters will be accepted as a correct match. The special character "%" indicates that only a single character in the position represented by the "%" will be accepted as a correct match. The two special characters may be combined in any order. Two simple examples may serve to clarify the use of the special characters. B* represents a set of character strings consisting of any number of characters starting with the character B. B% represents a set of character strings consisting of two characters, the first of which is B and the second of which may be any character.

4.1.2 SFS Directory Specification

Some commands allow the specification of a Shared File System directory name or "dirid". This can have the general form:

```
<filepoolid:><user ID>.<n1<.n2<.n3<.n4<.n5<.n6<.n7<.n8>>>>>>>>
```

where filepoolid gives the ID of the file pool in which the directory resides and n1 to n8 are directory names. At least a . must be specified. If the file pool ID is not specified, the default that you set with the CMS SET FILEPOOL command is used. The user ID is defaulted to the virtual machine issuing the command. The directory can also be specified in the form:

```
+fm.n1<.ni+1...ni+7>
or
-fm<.ni.ni+1...ni+7>
```

where these have the same meanings as described in the CMS command reference.

4.1.3 How to Read a Syntax Diagram

Follow the path of the line from left to right, from top to bottom.

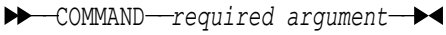
The definition of a command begins with two arrowheads pointing to the right.

The definition of a command ends with two arrowheads pointing to each other.

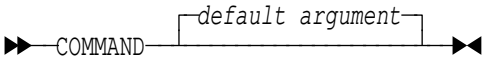
An arrowhead pointing to the right at the end of a line means that the definition is continued below.

An arrowhead pointing to the right at the beginning of a line means that the definition is continued from above.

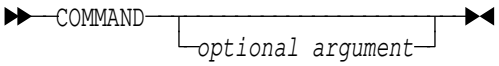
A required item is on the main path along the horizontal line.



A default item is above the main path.

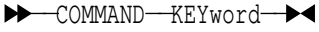


An optional item is below the main path.

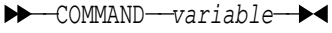


An item is a keyword, a variable, or a reference to a fragment in a syntax definition.

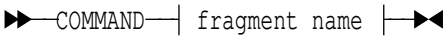
Keywords are shown in a Gothic font with the minimum abbreviation in uppercase.



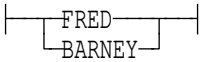
Variables are shown in lowercase italics.



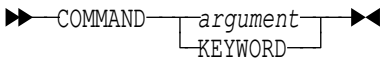
A reference to a fragment of a syntax definition breaks the main path with vertical bars. The fragment is defined later in the diagram.



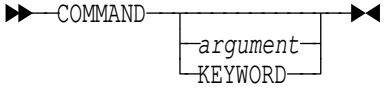
fragment name:



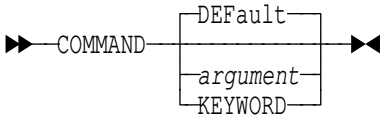
When you must choose between two or more items, they are stacked with the first one on the main path.



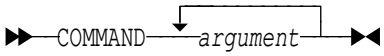
When you can select an item or take none, the choices are stacked below the main path.



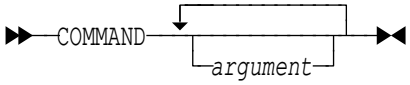
A default is shown above the main path.



An item may be repeated when an arrow returns to the left in front of it. The item is on the main path when you must write it at least once.



The item is below the main path when you may omit it altogether.



4.1.4 User Commands Summary

Following is a complete list of BARS/VM functions available to the general user:

Communication with Backup/Retrieve Processor

BACKUP	Controls backup parameters for minidisks and directories
BDELETE	Deletes files from backup
BLAH	Marks files backed up at a remote host
BPUT	Backs up files and minidisks immediately
QUERY	Requests information about backed up files and minidisks
RESTORE	Restores minidisks, SFS files and directories
RETRIEVE	Retrieves files and non-CMS minidisks from backup

Communication with Archive Processor

AALTER	Changes information for archived files
ADEFINE	Initializes archive processing for a user
ADELETE	Deletes files from archive
AGET	Retrieves files from archive
APUT	Submits files into archive

AQUERY	Requests information about archived files
ARENAME	Changes the name of an archived file
DQUERY	Requests information about deleted files in archive
DRECOVER	Makes deleted files available from archive

Communication with DBR Processor

DBR QUERY	Requests information about files in disaster backup
LPUT	Sends an LSET file to BARS/VM

Miscellaneous Functions

CQUEUE	Cancels, holds or releases commands which are queued but have not yet started executing.
DISPLAY	Displays general BARS/VM information
FILELIST	An alternative way of accessing the FILELIST screens for backup and archive.
HELP	Online HELP facility of BARS/VM

4.1.5 How to Authorize Users

Users may be authorized to issue BARS/VM commands on behalf of other users via the `OWNED` command. This can only be done by your BARS/VM administrator. Alternatively, if enabled by the BARS/VM administrator, authorization for some commands may be controlled via an External Security Manager. See 4.4, “Authorizing other users” on page 45 for more details.

4.1.6 Return and Reason Codes

Commands may give back return codes and reason codes. These may appear in acknowledgement files that are sent by some commands when something goes wrong. These files are called BACKUP ACK. The possible return code values are listed below.

```
 0 - successful completion
 4 - command not executed but not an error situation
 8 - syntax error
12 - bad value specified
16 - authorization error
20 - error obtaining or releasing storage
24 - backup or restore is in progress
28 - general error - see reason code and messages
32 - Callable Services Library error
```

Reason codes correspond to message numbers, and the explanation for a particular reason code may be found in the help for the corresponding message. For example, reason code 345 from APUT means the user is over his space allocation and the help for message HDF0345I explains this.

4.1.7 Dates

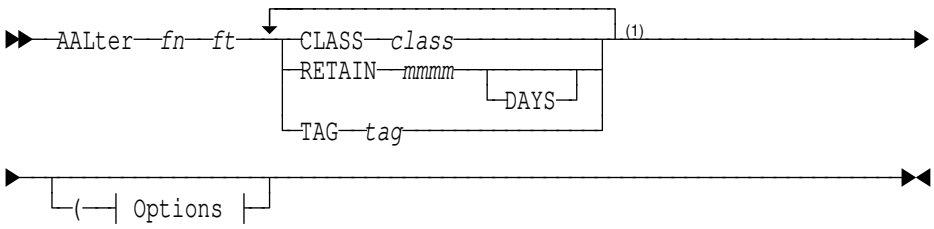
When dates are specified on commands, they are usually required to be of the form ddmmmyy, where dd is the day number, mmm is the mnemonic month and yy the year number, e.g. 10JAN96. Dates from 1JAN60 to 31DEC99 are taken to have years of 1960 to 1999 and dates from 1JAN00 to 31DEC59 are taken to run from years 2000 to 2059.

4.2 Command Syntax Description

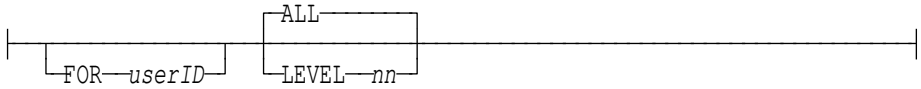
4.2.1 AALTER

The AALTER command allows the user to alter information associated with archived files.

General User Command



Options:



Note:
1 Each choice may only be specified once.

Operands

fn Specifies the file name of a specific file or set of files whose archive information is to be altered.

ft Specifies the file type of a specific file or set of files whose archive information is to be altered.

CLASS Specifies that the class of archived files is to be changed to that specified by *class*, which must be a single character.

RETAIN Specifies that the retention period is to be changed to *mmmm* months from now. If a file to be altered has been previously deactivated it is reactivated. *mmmm* may be any value from 1 to 999, but the actual maximum value is determined by the installation.

The optional **DAYS** keyword specifies that the RETAIN value be treated as days, rather than months. Again, it is limited by the installation maximum, but may be up to four digits.

TAG Specifies that the tags of archived files are to be changed to that specified by *tag*. The tag may not be more than 8 characters. The characters * and % are not allowed in the tag specification.

Options

FOR Specifies that the user ID whose data is to be changed belongs to the specified user ID rather than the user ID of the command issuer. The command issuer must have been authorized to do this.

LEVEL Specifies that only a particular archived file level (*nn*) is to be altered.

ALL Specifies that all levels of the specified files are to be altered.

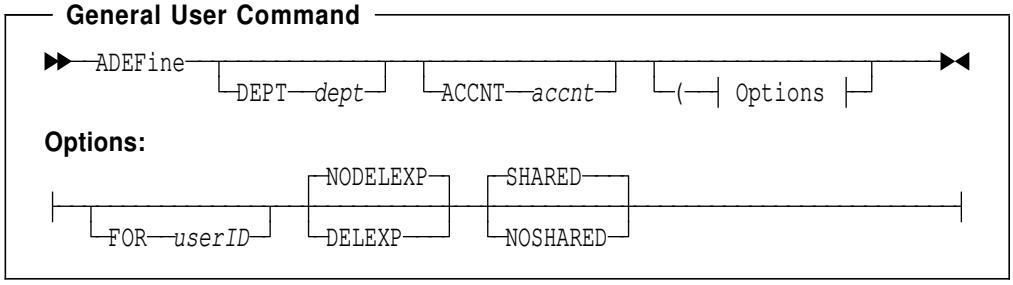
Usage Notes:

1. For *fn* and *ft*, the ***** and **%** notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12 is **not allowed** when using the command with the **RETAIN** operand.

See 4.3, “Sample BARS/VM Commands” on page 41. for some examples of the command.

4.2.2 ADEFINE

The ADEFINE command defines the potential owner of archived data via the department number or description and the account number. Each user ID under which data is to be archived must be first defined to BARS/VM so that data may be associated with a department or account number. When defining a new user ID, both DEPT and ACCNT must be specified. Data associated with an already existing user ID may be changed by specifying either or both of these operands. If only the options are to be changed, both DEPT and ACCNT may be omitted. The SHARED and NOSHARED operands apply to a shared system (SSI) environment only.



Operands

DEPT Specifies a one to 8 character department name or number.

ACCNT Specifies a one to 8 character account identification.

Options

FOR Specifies that the user ID to be defined is the specified user ID rather than the user ID of the command issuer. The command issuer must have been authorized to do this.

DELEXP Specifies that archived files may be erased by the installation after they have reached their expiry date.

NODELEXP Specifies that expired archived files may only be erased after they have been deactivated. This is the default.

SHARED NOSHARED

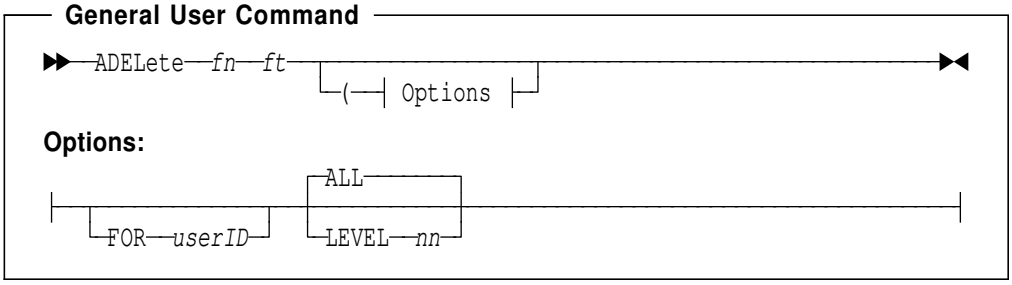
Apply to a shared data environment or SSI only. In such a BARS/VM environment there may be the same user ID on several systems with the same owner or with different owners or different work. The SHARED and NOSHARED options allow the user to define whether there is to be a single archive or a separate archive on each system. The default option in ADEFINE is SHARED, which means that the user ID has a single archive and all archive commands are routed to the node where the first ADEFINE was processed. All other ADEFINEs affect only that single archive in a BARS/VM coupled system. That is the way it should work for most

users who don't know where they sign on and who have access to their minidisks on all systems. NOSHARED allows different sets of archive in such a complex. Only the local user ID gains access to the archive data. When NOSHARED is specified on one system, all following ADEFINEs on other BARS/VM systems are reset to NOSHARED.

See 4.3, “Sample BARS/VM Commands” on page 41. for some examples of the command.

4.2.3 ADELETE

The ADELETE command is used to delete files from the archive system. Note that if the command is used to delete all files for a user ID, then an ADEFINE must be issued before files can be archived again for that user ID.



Operands

fn Specifies the file name of a specific file or set of files which are to be deleted from the archive subsystem.

ft Specifies the file type of a specific file or set of files which are to be deleted from the archive subsystem.

Options

FOR Specifies that the user ID whose data is to be deleted belongs to the specified user ID rather than the user ID of the command issuer. The command issuer must have been authorized to do this.

LEVEL Specifies that only a particular level of an archived file or files is to be deleted.

ALL Specifies that all levels of the specified files are to be deleted.

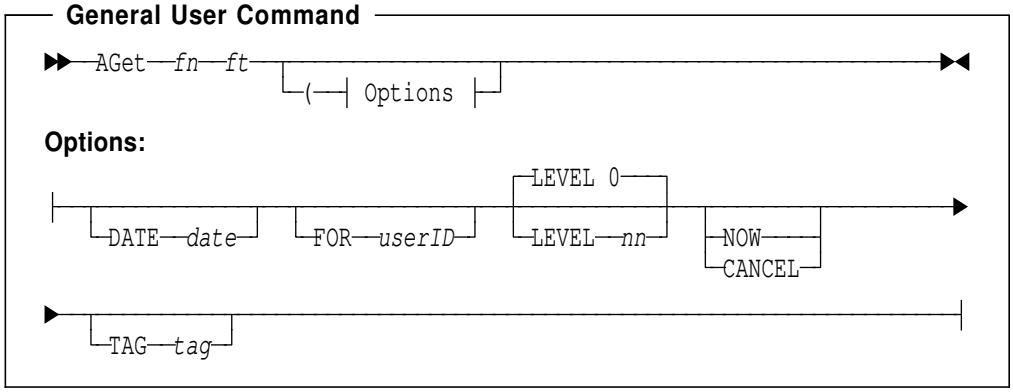
Usage Notes:

1. File name and file type may be specified using the * and % notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

See 4.3, “Sample BARS/VM Commands” on page 41. for some examples of the command.

4.2.4 AGET

The AGET command retrieves specified files from BARS/VM archive volumes. Individual archived files may be retrieved and these are placed in the command issuer's virtual card reader. The retrieves may be requested immediately or the operation deferred until a time determined by the installation. There are installation limits on how many files a user is allowed to get from archive during the day. Note that retrieved files are **NOT** erased from archive. Archived files are only erased explicitly with the ADELETE command. See Chapter 3, “Using the ARCHIVE Function” on page 5 for general information on the use of the AGET command.



Operands

fn Specifies the file name of a specific file or set of files to be retrieved from archive.

ft Specifies the file type of a specific file or set of files to be retrieved from archive.

Options

CANCEL Cancels a deferred request.

Only the first matching request is cancelled. The *fn*, *ft* and **FOR** *userID* fields are compared. Other fields, including **LEVEL**, are ignored.

DATE Specifies that only archived files which were archived on the specified date are to be retrieved. The date must be of the form ddmmmyy, where dd is the day number, mmm is the mnemonic month and yy the year number, e.g. 10JAN96.

FOR Specifies that the archived data is associated with the specified user ID rather than the user ID of the command issuer. The command issuer must have been authorized to do this.

LEVEL Specifies the level number for the archived file to be retrieved. *nn* specifies the particular level. The default is level 0.

NOW Specifies that the command is to be processed immediately and not deferred. Deferred commands may be displayed by the DISPLAY command and cancelled by using the **CANCEL** option of AGET.

TAG Specifies that only archived files which have the specified tag are to be retrieved. *tag* may be up to 8 characters in length. A generic tag may be specified using the * and % characters.

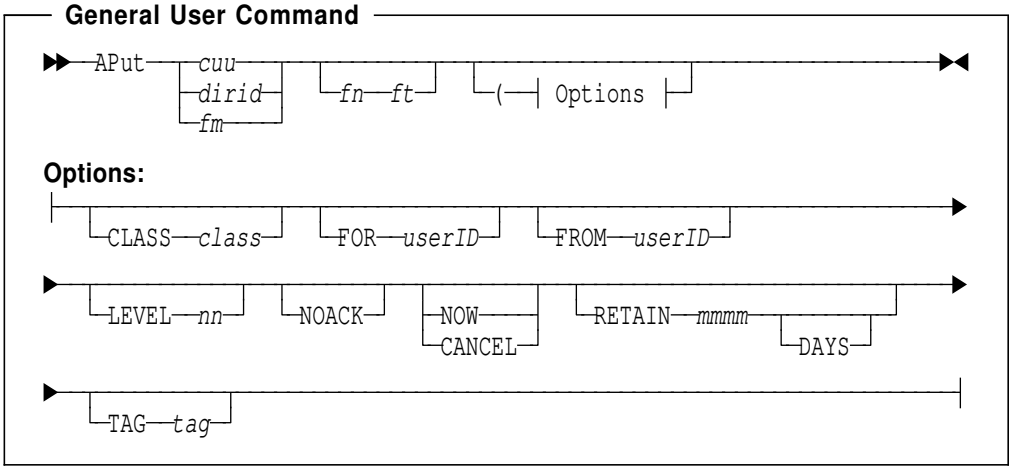
Usage Notes:

1. File name and file type may be specified using the * and % notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

See 4.3, “Sample BARS/VM Commands” on page 41. for some examples of the command.

4.2.5 APUT

The APUT command makes archive copies of specified CMS files. The copies may be made immediately or the operation deferred to an installation defined time of day. There are installation limits on how many files a user is allowed to archive during the day. Note that when a file is archived the original file on the user's minidisk or SFS directory is **NOT** erased. See Chapter 3, “Using the ARCHIVE Function” on page 5 for general information on the use of the APUT command.



Operands

cuu Specifies a minidisk on which the files to be archived reside. It must be specified as 2-4 characters.

dirid Specifies an SFS directory in which the files to be archived reside.

fm Specifies a single file mode letter defining the minidisk or SFS directory on which the file or files to be archived reside. If a file mode letter is specified for an accessed minidisk, the requested file or files are sent to BARS/VM using DISK DUMP instead of being copied from the minidisk on which they reside. This allows files to be sent to another node for archive. If archive is being done to the local BARS/VM system it is more efficient to use the cuu operand. If a file mode letter is specified for an accessed directory, the files are copied in the same way as if a dirid was specified. If the files reside on a temporary minidisk, then specifying a file mode letter is the only way of archiving them.

fn Specifies the file name of a specific file or set of files for which an archive copy is to be made.

ft Specifies the file type of a specific file or set of files for which an archive copy is to be made.

Options

CANCEL Cancels a deferred request.

Only the first matching request is cancelled. The *fn*, *ft*, *cuu* or *dirid*, **FROM** *userID* and **FOR** *userID* fields are compared. Other fields, including **LEVEL**, are ignored.

CLASS Specifies a one character class for the archived files. The meanings of these characters are defined by the installation. For example, `CLASS C` could have the meaning **"CONFIDENTIAL"**. The class definitions may be displayed by using the DISPLAY CLASS command.

FOR Specifies that the archived files are to be associated with the specified user ID rather than the user ID of the command issuer. The command issuer must have been authorized to do this.

FROM Specifies that the files to be archived reside on a minidisk belonging to the specified user ID rather than that of the command issuer. The command issuer must have been authorized to do this. This option is not allowed if "dirid" is specified since the user ID to whom the files belong is taken from the dirid.

LEVEL Specifies a level number for the archived file. nn is the Level number - 0 is the newest, then 1, 2 and so on. If LEVEL nn is not specified, the files being archived become level 0 and the older versions increase by 1 - 1 becomes 2, 2 becomes 3, 3 becomes 4, and so on. If LEVEL nn is specified, the existing nnth level is erased and a new nnth Level created, and other levels are unaffected. If no level exists, the lowest available level number smaller than the specified one is used. *nn* may be a number from 0 to 99 but the number of levels which may be created are limited by the installation.

NOACK Specifies that BARS/VM should not return a spool file acknowledgment when the command has completed.
Warning: This option should be used with care, since the archive operation may fail, and this file is normally used to provide confirmation of success.

NOW Specifies that the command is to be processed immediately and not deferred. Deferred commands may be displayed with the DISPLAY command and cancelled by using the **CANCEL** option of APUT.

RETAIN Specifies a retention period for the archived files. mmmm is the number of months the files should be retained by BARS/VM. After this time the files will be deactivated if an AAUDIT command with the DEACT operand is issued by the administrator. Some time later the deactivated data may be erased by the administrator via the AAUDIT command specifying DELDEACT. Expired data will also be deleted if the user specified DELEXP when issuing the ADEFINE command.

mmmm may be any value from 1 to 999 but the actual maximum value is determined by the installation. If RETAIN is not specified than the default value as defined by the installation is used. The optional **DAYS** keyword specifies that the RETAIN value be treated as days, rather than months. In this case, the installation maximum is converted from months to days (assuming 30 days per month)

before the limit is checked. Four digits may be specified with the DAYS option.

TAG Specifies that a "tag" value is to be associated with the archived files to aid later identification. If specified, the tag can be up to 8 characters in length. (e.g. TAG docrel32 could be used to group together all documentation from REL.32). The characters * and % are not allowed in the tag.

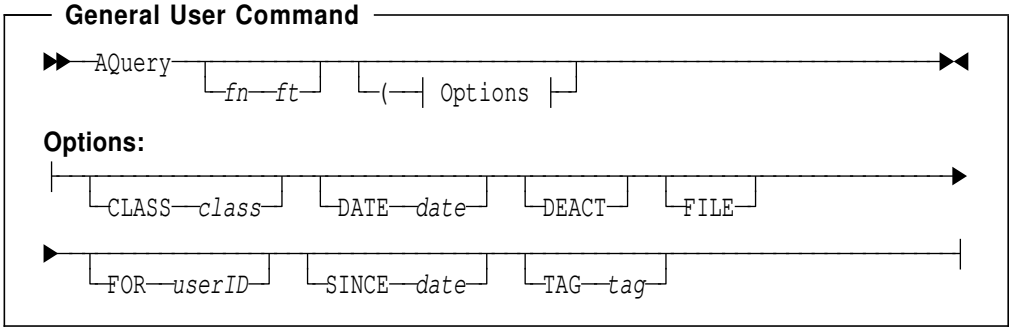
Usage Notes:

1. If the file name and type are omitted, then all files on the specified minidisk will be archived (i.e. equivalent to * *). However, only a one line acknowledgement file will be sent.
2. See 3.1, “Deferred Processing and APUT/AGET Completion” on page 5 for details of the acknowledgement file formats and return codes.
3. File name and file type may be specified using the * and % notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

See 4.3, “Sample BARS/VM Commands” on page 41. for some examples of the command.

4.2.6 AQUERY

The AQUERY command displays information describing archived files. The display may be of all archived files or may be limited in various ways.



Operands

fn ft Specify particular files.

If these are specified then BARS/VM displays additional information for the particular file or files specified.

Options

CLASS Specifies that only archived files which have the specified class are to be displayed. *class* must be a single character.

DATE Restricts the display of files to those files which were archived on the specified date. The date must be of the form ddmmmyy, where dd is the day number, mmm is the mnemonic month and yy the year number, e.g. 10JAN96. The DATE operand must not be used if SINCE is specified.

DEACT Specifies that only archived files which have been deactivated are to be displayed.

FILE Specifies that the result of the command is to be sent to the command issuer as a file to his card reader rather than being displayed at his terminal. The file is sent in NETDATA format.

FOR Specifies that the user ID to be queried is the specified user ID rather than the user ID of the command issuer. The command issuer must have been authorized to do this.

SINCE Restricts the display of files to those files which were archived on or after the specified date. The date must be of the form ddmmmyy, where dd is the day number, mmm is the mnemonic month and yy the year number, e.g. 10JAN96. The SINCE operand must not be used if DATE is specified.

TAG Specifies that only archived files which have the specified tag are to be displayed. *tag* may be up to 8 characters in length.

Usage Notes:

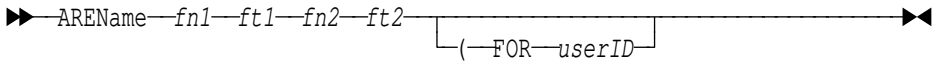
1. File name, file type and tag may be specified using the * and % notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.
2. SFS aliases are not displayed if the SINCE or DATE option is specified.

See 4.3, “Sample BARS/VM Commands” on page 41. for some examples of the command.

4.2.7 ARENAME

The ARENAME command allows a user to rename archived files.

General User Command



Operands

fn1 Specifies the file name of the archived file to be renamed. A single asterisk may be used to specify all file names but in this case fn2 must be specified as = (see below).

ft1 Specifies the file type of the archived file to be renamed. A single asterisk may be used to specify all file types but in this case fn2 must be specified as = (see below).

fn2 Specifies a new file name. It may be specified as = if the file name is to remain unchanged.

ft2 Specifies a new file type. It may be specified as = if the file type is to remain unchanged.

Options

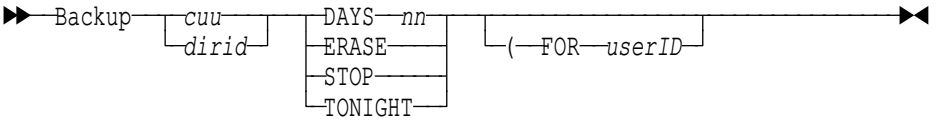
FOR Specifies that the user ID whose data is to be renamed is the specified user ID or pool name rather than the user ID of the command issuer. The command issuer must have been authorized to do this.

See 4.3, “Sample BARS/VM Commands” on page 41. for some examples of the command.

4.2.8 BACKUP

The BACKUP command changes the backup parameters for a specified minidisk or SFS directory.

General User Command



Operands

cuu Defines a minidisk whose backup parameters are to be changed.

dirid Specifies an SFS directory whose backup parameters are to be changed.

DAYS nn Requests an automatic backup every nn days. The number specified may be between an installation defined minimum and 99.

ERASE Ends automatic backup and requests that all backup data be erased after an installation defined period. A DISPLAY OPTIONS command displays the value of the period.

STOP Ends automatic backup of the minidisk or directory.

TONIGHT Requests that a backup should occur at the next backup cycle. The scheduling of cycles is the responsibility of the BARS/VM administrator.

Options

FOR Specifies that a minidisk whose backup parameters are be changed belongs to the specified user ID rather than the user ID of the command issuer.

This is only used in the case where one user ID owns another, i.e. is authorized to issue BARS/VM commands on behalf of that user ID. This option is not allowed if an SFS "dirid" was specified since the user ID is taken from the dirid.

Usage Notes:

1. Files on CMS minidisks and SFS files in file pools selected for backup by the administrator are backed up unless the user or installation requests otherwise.

   The last **'v'** versions of CMS files are backed up. **'v'** is an installation option and may be shown with the DISPLAY OPTIONS command.

2. A non-CMS minidisk (i.e. a minidisk not CMS formatted or a BLOCKIO minidisk) is **only** backed up if the user or administrator issues a BACKUP command for it.

   Note: Minidisks reserved for ***BLOCKIO** (e.g. SQL minidisks) are treated as **NON-CMS minidisks**. Therefore you have to issue a BACKUP command to get backup for those minidisks.

   There are two ways to request backup:

   a. `VMBARS BACKUP cuu DAYS nn`

b. `VMBARS BACKUP cuu TONIGHT`

DAYS nn. - request that backup occurs every *nn* days.

Advantage: Automatic backup every *nn* days.

Disadvantage: Your 'critical' changes will not be backed up until *nn* days.

TONIGHT - request a 'deferred BACKUP', which means that backup occurs in the **next backup cycle**.

Advantage: You can enter BACKUP after 'critical' changes.

Disadvantage: You must remember to enter the command.

TONIGHT - may be used for **one time only** backup.

Note: Users should use the BACKUP command only if they require backup. If they will never RETRIEVE data, they should issue a BACKUP STOP or ERASE command.

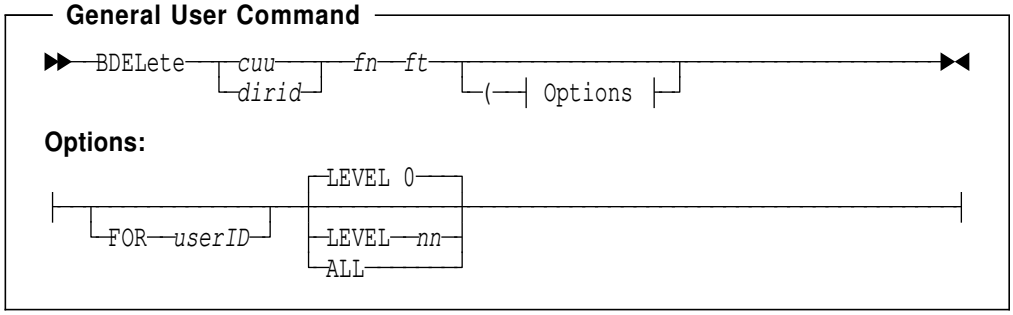
3. An erasure date set by the ERASE option will be removed by a BPUT command specifying that all files are to be backed up. See the usage notes for the BPUT command.

4.2.9 BDELETE

The BDELETE command deletes specified files from the backup catalogs.

It is not normally necessary to erase any backup data since BARS/VM automatically deletes backup copies of CMS files following their erasure from the user's minidisk or SFS directory. The backup copies of any erased files are retained by BARS/VM for a number of days as defined by the installation.

This command is useful in circumstances where sensitive data may have been backed up without proper protection. (e.g. highly confidential files which have been accidentally backed up unciphered).



Operands

cuu Defines a minidisk from which the files were backed up.

dirid Specifies an SFS directory from which the files were backed up.

fn Specifies the file name of a specific file or set of files to be deleted. "*" on its own is not allowed.

ft Specifies the file type of a specific file or set of files to be deleted. "*" on its own is not allowed.

Options

LEVEL nn Requests deletion of a particular backup level (0 is the latest level, 1 the next to latest level and so on).

The QUERY command may be used to determine which backup levels exist.

ALL Specifies that all existing backup levels are to be deleted.

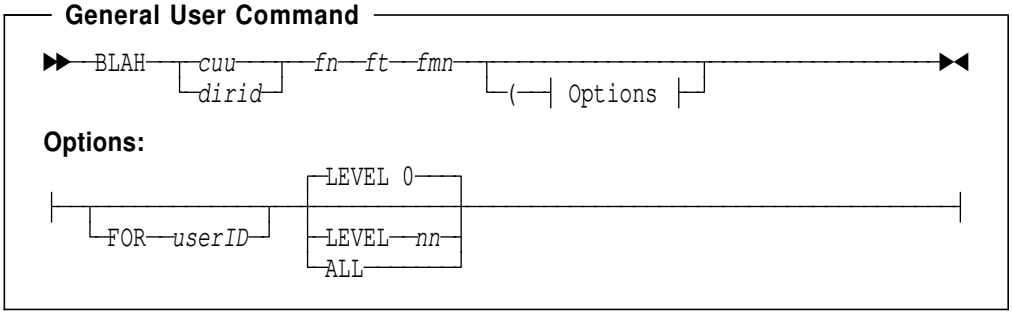
FOR Specifies that the files to be deleted are owned to the specified user ID rather than the user ID of the command issuer. This is only used in the case where one user ID owns another, i.e. is authorized to issue commands on behalf of that user ID. This option is not allowed if an SFS "dirid" was specified since the user ID is taken from the dirid.

Usage Notes:

1. Data erased with the BDELETE command is irretrievable.
2. File name and file type may be specified using the * and % notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

4.2.10 BLAH

The BLAH command marks the specified files as backed up at a remote host system. This informs the local BARS/VM system that it may erase the backup on this system and reuse the space that it occupied.



Operands

cuu Defines a minidisk from which the files were backed up.

dirid Specifies an SFS directory from which the files were backed up.

fn Specifies the file name of a specific file or set of files to be processed.

ft Specifies the file type of a specific file or set of files to be processed.

fmn Specifies the filemode number of the file to be processed.

The filemode number may be specified either with a number only or with a letter combined with a number.

```
e.g. 1 and C1 means both filemode number 1.
```

Options

LEVEL nn Specifies a particular backup level (0-latest level, 1-next to latest level).

The QUERY command may be used to determine which backup levels exist.

ALL Specifies that all existing backup levels are processed.

FOR Specifies that the files belongs to the specified user ID rather than the user ID of the command issuer.

This is only used in the case where one user ID owns another, i.e. is authorized to issue commands on behalf of that user ID. This option is not allowed if an SFS "dirid" is specified since the user ID is taken from the dirid.

Usage Notes:

1. This command is normally sent by a remote host BARS/VM system to another BARS/VM system; it is **not** provided to be issued by a general user.
2. Data processed by the BLAH command cannot be retrieved from the local BARS/VM system.
3. File name and file type may be specified using the * and % notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

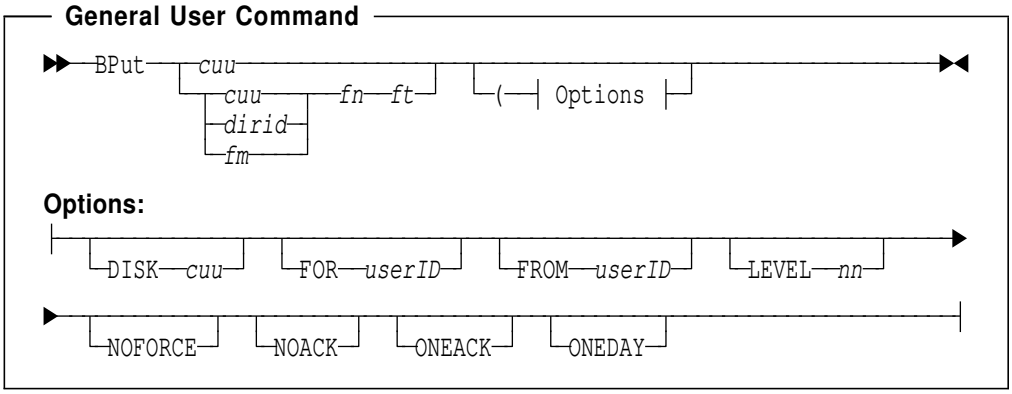
4.2.11 BPUT

The BPUT command makes backup copies of specified CMS files, or it may be used to backup a complete non-CMS minidisk (see usage note 2 on page 28). Users should be aware that any file backed up by this command has a new level 0 backup created and the oldest level is then lost, so care must be taken to avoid destroying a good backup of a file which might be needed in the event of its loss.

CMS files are backed up even if they are unchanged since the last backup cycle unless the NOFORCE option is specified.

SFS aliases are not backed up by the BPUT command. However, entries are maintained in the backup catalogs.

When the file name and file type are specified (i.e. not a non-CMS BPUT), a multi-line acknowledgement file is returned on command completion which contains one record for each file processed. The record indicates whether the file was processed successfully, ignored (if NOFORCE was specified, and flagged by an * in column 1), or failed to be processed (a return code is shown). Production of this file can be overridden by use of the ONEACK or NOACK options.



Operands

cuu Specifies a minidisk to be backed up. It must be specified as 2-4 characters.

dirid Specifies an SFS directory on which files to be backed up reside.

fm Specifies a single filemode letter defining a minidisk on which a file or files to be backed up reside. If a filemode letter is specified for an accessed minidisk, the requested file or files are sent to BARS/VM using DISK DUMP instead of being copied from the minidisk on which they reside. This allows files to be sent to another node for backup. If fm is specified then DISK (see below) should also be given to define the minidisk cuu.

If a filemode letter is specified for an accessed directory, the files are copied in the same way as if a dirid was specified.

fn Specifies the file name of a specific file or set of files for which backup copies are to be made.

ft Specifies the file type of a specific file or set of files for which backup copies are to be made.

Options

LEVEL nn Specifies a level number for the backed up file. nn is the Level number - 0 is the newest, then 1,2.. If LEVEL nn is not specified, a new one is built as level 0 and the older one increases by 1, i.e. 1 becomes 2, 2 becomes 3, 3 becomes 4, and so on. If LEVEL nn is specified, the existing nnth level is erased and a new nnth Level created. Levels before are not changed. If no level exists, the lowest available level number smaller than the specified one is used. *nn* may be a number from 0 to 99 but the number of levels which may be created are limited by the installation.

DISK Specifies the disk on which the files to be backed up reside. This option is used when a filemode is specified as the first operand of the command to give the cuu of the minidisk with which the backed up file is to be associated. This option is not allowed if an SFS "dirid" is specified since the user ID is taken from the dirid.

NOFORCE Specifies that each CMS file to be backed up is only to be backed up if it has changed since the last backup time. This command option is useful for automating database backup at a time when the database is consistent. Setting such a minidisk to NOBACKUP will enable a user to have complete control over his backup times.

NOACK Specifies that the command issuer does not wish to receive the acknowledgement file normally created by the BPUT command. NOACK will override the ONEACK option.

ONEACK Specifies that the command issuer wishes to receive a one-line acknowledgement file rather than the normal multi-line acknowledgement file. This option is overridden by the NOACK option.

ONEDAY Specifies that the backup is to proceed *only* if no backup has already taken place today for the specified minidisk, i.e. the response to a QUERY command shows todays date. This option is useful if you have a requirement of ensuring that a BPUT is only executed once a day for a minidisk. If no backup is done because the last backup date is today, then a normal one-line acknowledgement file is sent indicating success.

FROM Specifies that the files to be backed up reside on a minidisk belonging to the specified user ID rather than that of the command issuer. The command issuer must have been authorized to do this. This option is not allowed if an SFS "dirid" is specified since the user ID to whom the files belong is taken from the dirid.

FOR Specifies that the backed up files are to be associated with the specified user ID rather than the user ID of the command issuer. The command issuer must have been authorized to do this. This option is not allowed if an SFS "dirid" is specified since the user ID with whom the backed up files will be associated is taken from the dirid.

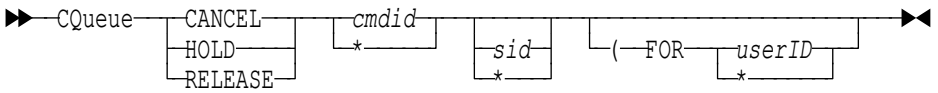
Usage Notes:

1. File name and file type may be specified using the ***** and **%** notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.
2. File name and file type must be specified with a dirid or fm. If a cuu is specified and file name and file type are omitted, this requests that the minidisk be backed up as **non-CMS**.
3. The date last processed (which is displayed by the QUERY command) is only set to the current date by a BPUT of CMS files when * is specified for the file name and file type (i.e. when all files are to be processed). For a non-CMS minidisk the date is always set since the entire minidisk is backed up.
4. When * is specified for the file name and file type (i.e. when all files are to be processed), then any existing date which is set for the deletion of all backup data for the minidisk or directory is removed. Such a date may be set, for example, by the BACKUP command with the ERASE option.
5. If the NOFORCE option is specified, then any erased files are detected (and marked as such) in the catalog file.
6. Automatic volume separation is not enforced when the BPUT command is used to backup a file or minidisk. Thus, there can be no guarantee that the level 0 and 1 copies will reside on different backup volumes.
7. If the installation has specified that backup is directly to tape, then BPUT of CMS files and non-CMS minidisks will be done to tape.

4.2.12 CQUEUE

The CQUEUE command alters the status of commands which are queued for execution but not yet started. These commands are shown by DISPLAY STATUS QUEUES.

General User Command



Operands

CANCEL Cancels the specified queued commands.

HOLD Specifies that commands are to be held in the command queue and not executed.

RELEASE Specifies that commands are to be released from HOLD status.

cmdid Is the command identification as shown by the DISPLAY STATUS QUEUES command. An * specifies all command identifications.

sid Is the session identification as shown by the DISPLAY STATUS QUEUES command. An * specifies all session identifications. If the operand is not specified it defaults to the current session.

Options

FOR This may only be used by authorized users. It specifies that commands initiated by the specified user id are to be processed and not those of the command issuer. If * is specified, commands from all user ids are processed.

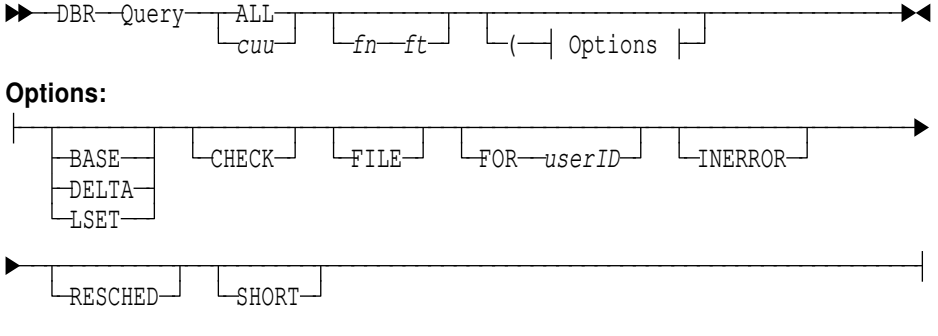
Usage Notes:

1. This command does not process deferred commands. Such commands may only be cancelled through the CANCEL option on the relevant command.

4.2.13 DBR

The DBR QUERY command displays disaster backup information for the minidisk defined by the parameter *cuu* or for all minidisks where disaster backup data exists if **ALL** is specified.

General User Command



Operands

ALL Requests disaster backup information for all minidisks owned by the specified user ID.

cuu Defines the user minidisk address for which disaster backup information should be displayed.

fn ft Specify particular CMS files. If these are specified then BARS/VM displays additional information for the particular file or files specified.

Options

BASE Restricts the display of files to those files which have been backed up for the last BASE DBR run.

DELTA Restricts the display of files to those files which have been backed up for the last DELTA DBR run.

LSET Restricts the display of files to those files which have been backed up for the last LSET or User ID DBR run.

CHECK Specifies that the result of the QUERY command is stripped down to those which have **not completed the last DBR run**.

FILE Specifies that the result of the DBR QUERY command is to be sent as a file to the command issuer's card reader in NETDATA format.

FOR Specifies that the user ID to be queried is the specified user ID rather than the user ID of the command issuer. The command issuer must have been authorized to do this.

INERROR Restricts the display of files to those files which could not be dumped during the specified DBR run.

RESCHED Restricts the display of files to those files which could not be dumped from a backup volume during the specified DBR run, but got rescheduled from the users minidisk. The RESCHED option is useful to verify the consistency of the BARS/VM backup data.

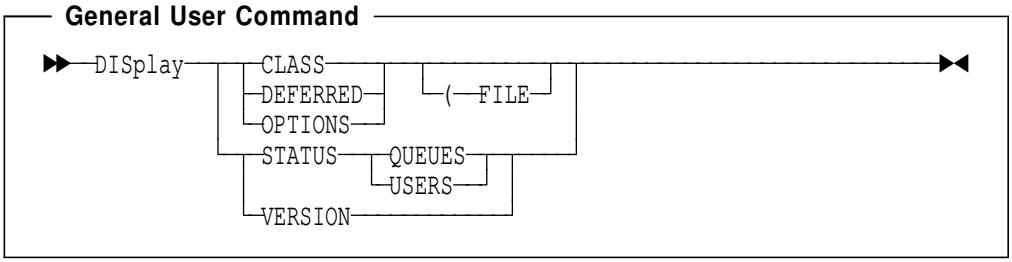
SHORT Specifies that the result of the DBR QUERY is to be a single line message for each minidisk requested giving the format of the minidisk and when disaster backup occurred. The SHORT option is useful when specifying **ALL** to reduce the number of messages received.

Usage Notes:

1. The cuu, file name and file type may be specified using the ***** and **%** notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

4.2.14 DISPLAY

The DISPLAY command displays general BARS/VM information.



Operands

CLASS Displays the archive classes and their meanings. These classes are used on the APUT command.

DEFERRED Requests a display of outstanding deferred commands for the user issuing the command, i.e. RETRIEVE, APUT or AGET commands which are to be processed at the time determined by the installation for deferred command processing.

OPTIONS Displays the current settings of the BARS/VM installation options.

STATUS Displays the current status of BARS/VM.

QUEUES Lists all pending commands which are queued but not yet started execution. These commands may be cancelled, held, or released using the CQUEUE command.

USERS Lists the current status of the various subuser virtual machines.

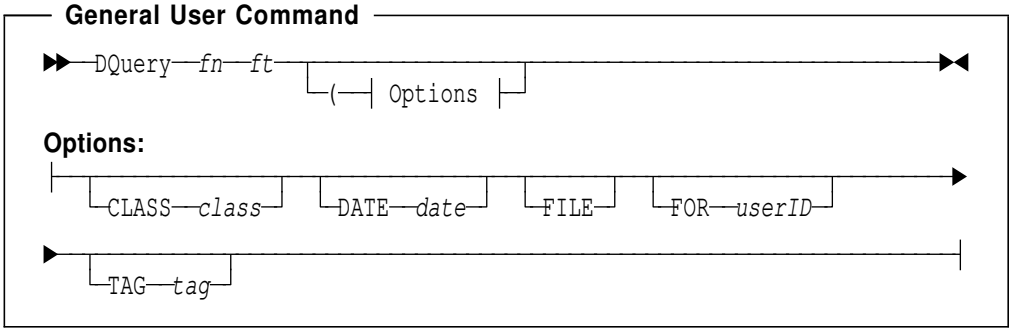
VERSION Displays the version, release and modification level of the BARS/VM system.

Options

FILE Specifies that the result of the command is to be sent to the command issuer as a file to his card reader rather than being displayed at his terminal. The file is sent in NETDATA format.

4.2.15 DQUERY

The DQUERY command displays information describing archived files which have been deleted but are still available for recovery via the DRECOVER command. See 3.5, “DQUERY and DRECOVER” on page 9 for a discussion on recovering deleted archive files.



Operands

fn Specifies the file name of a specific file or set of files whose information is to be displayed.

ft Specifies the file type of a specific file or set of files whose information is to be displayed.

Options

CLASS Specifies that only deleted files which have the specified class are to be displayed. *class* must be a single character.

DATE Restricts the display of files to those files which were archived on the specified date. The date must be of the form ddmmmyy, where dd is the day number, mmm is the mnemonic month and yy the year number, e.g. 10JAN96.

FILE Specifies that the result of the command is to be sent to the command issuer as a file to his card reader rather than being displayed at his terminal. The file is sent in NETDATA format.

FOR Specifies that the user ID to be queried is the specified user ID rather than the user ID of the command issuer. The command issuer must have been authorized to do this.

TAG Specifies that only deleted files which have the specified tag are to be displayed. *tag* may be up to 8 characters in length.

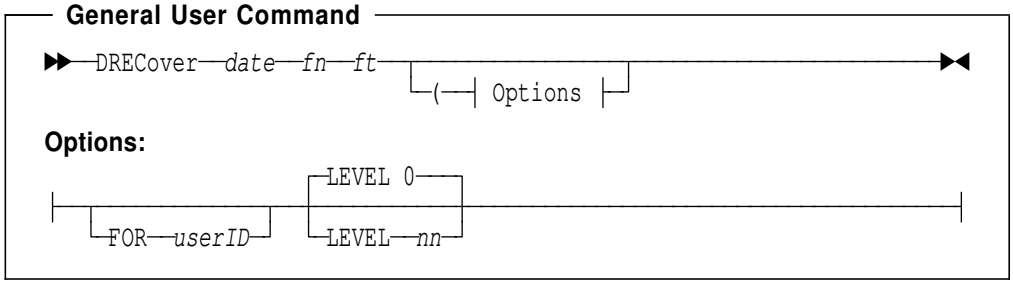
Usage Notes:

1. File name, file type and tag may be specified using the ***** and **%** notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

See 4.3, “Sample BARS/VM Commands” on page 41. for some examples of the command.

4.2.16 DRECOVER

The DRECOVER command recovers specified files which have been deleted from archive by the user but whose archive information is still available, thus allowing access to the archived file. The recovered files are placed in the command issuers virtual card reader. See 3.5, “DQUERY and DRECOVER” on page 9 for a discussion on recovering deleted archive files.



Operands

date Defines the date on which the file was deleted from archive. This is the date shown in the “Deletion Date” column of the DQUERY response. It must be of the form ddmmmyy, where dd is the day number, mmm is the mnemonic month and yy the year number, e.g. 10JAN96.

fn Specifies the file name of a specific file or set of files to be recovered.

ft Specifies the file type of a specific file or set of files to be recovered.

Options

FOR Specifies that the user ID whose data is to be recovered is the specified user ID rather than the user ID of the command issuer. This is only used in the case where one user ID owns another, i.e. is authorized to issue commands on behalf of that user ID.

LEVEL Requests a particular archive level (0-latest level, 1-next to latest level). The DQUERY command may be used to determine which archive levels exist. If **LEVEL** is unspecified, then level 0 is assumed.

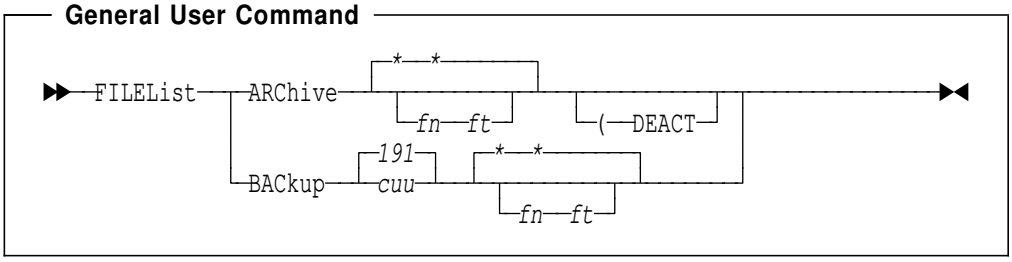
Usage Notes:

1. File name and file type may be specified using the * and % notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

See 4.3, “Sample BARS/VM Commands” on page 41. for some examples of the command.

4.2.17 FILELIST

The FILELIST command is an alternative method of accessing the FILELIST panel results screens for archive and backup, without having to go via the intermediate panels. On exit from the display, you are returned to where you were when you invoked the command, and hence do not need to step back out through the panels.



Operands

cuu Defines a user minidisk address for which the backup files should be displayed. ALL may be specified to show all minidisks. The default for cuu is 0191.

fn Specifies the file name of a specific file or set of files to be displayed.

ft Specifies the file type of a specific file or set of files to be displayed.

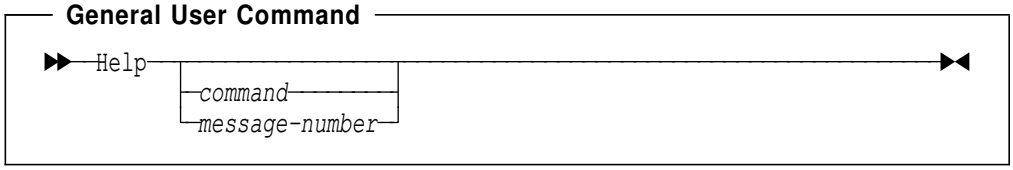
Options

DEACT Specifies that only archived files which have been deactivated are to be displayed. If this is not specified, then only active files are shown.

Usage Notes:

1. File name and file type may be specified using the * and % notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

4.2.18 HELP

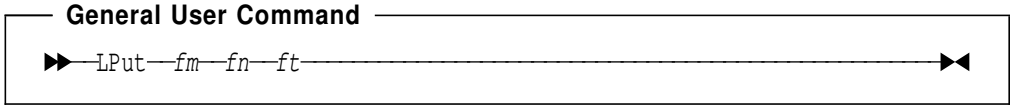


The HELP command explains a specified command or message or gives a list of available commands.

- If `VMBARS HELP command-name` is specified then a description of the specified command/function will be displayed.
- If `VMBARS HELP message-number` is specified then a description of the specified message will be displayed. The format of message-number is **HDFnnnns** where **nnnn** is the numeric part of the message and **s** is the severity. For example, for message `HDFRET0303E` the corresponding message-number for HELP would be `HDF0303E`.
- If no command or message number is specified, a list of commands is provided with the option of obtaining further help for any of those.

4.2.19 LPUT

The LPUT command sends an LSET file to the backup subsystem to be applied to the disaster backup definitions. The LSET file must exist. Only the owning or defined authorized users are allowed to manipulate data.
LSET is an abbreviation of "Logical SET of minidisks", which is a description of related minidisk objects for an LSET based disaster dump.



Operands

fm Specifies a single file mode letter defining the minidisk or SFS directory on which the LSET file to be updated resides. The specified file is sent together with the command.

fn Specifies the file name of a specific LSET file for which disaster backup information is to be updated.

ft Specifies the file type of a specific LSET file. The only file type allowed is LSET.

Usage Notes:

1. The use of the LPUT command requires CMS PIPELINE.
2. The following is a possible layout of an LSET file. Note that only *_OWNER: and *_AUTHORIZED: are used for authorization verification.

```
*_OWNER: MANAGER NODEID * owner name, department, location
*_Authorized: TOM HARRY
*_LSET description: This LSET is a critical application.
*_LSET description: It is required to ...........
*_Critical LSET: N
*_Corequisite LSET: CMSR10 PL1
*_Corequisite LSET: PASCAL
*_Disaster Backup Recovery Plan: RECOVER PLAN USERREC 191
*_Cycle: 12 days
APPLID1 *
APPLID  0191
HARRY 0192
```

Sample possible LSET file layout

The interpretation is as follows:

***_OWNER:** Owner userid is MANAGER at NODEID. This userid is seen as the person who owns the LSET definition.

***_AUTHORIZED:** A maximum of two parameters can be defined. These local userids are seen as authorized by the owner to change and adjust the LSET definitions.

***_Corequisite LSETs:** Any number of statements can appear. Each word is checked at dump time for an existing LSET file with such a filename and, if found, the minidisk definitions are appended to the list of minidisks to be dumped.

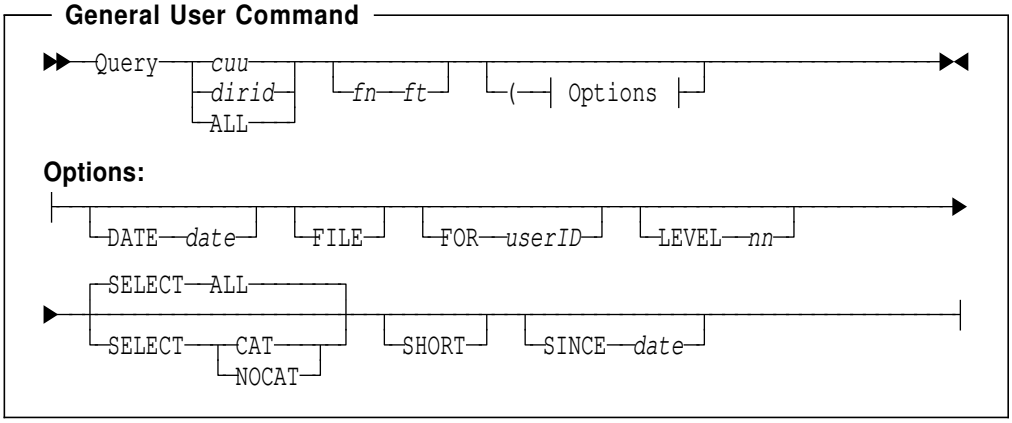
***_Disaster Backup Recovery plan:** The third and fourth word define the userid and minidisk address where the recovery information resides. This minidisk is automatically added to the minidisk definitions to be dumped.

***_xxxxx:** All other comment cards are treated as "comments".

Userid CCUU defines a Userid/Minidisk pair, one on each line. Userid and minidisk may be specified using the ***** and **%** notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

4.2.20 QUERY

The QUERY command displays backup information for minidisks and SFS directories.



Operands

cuu Defines a user minidisk address or set of addresses for which backup information should be displayed.

ALL Requests backup information for all minidisks and SFS directories for the specified user ID.

dirid Specifies an SFS directory for which backup information is to be displayed.

fn ft Specify particular CMS files.

If these are specified then BARS/VM displays additional information for the particular file or files specified.

Options

DATE Restricts the display of files to those files which were backed on the specified date. The date must be of the form ddmmmyy, where dd is the day number, mmm is the mnemonic month and yy the year number, e.g. 10JAN96. The DATE operand must not be used if SINCE is specified.

FILE Specifies that the result of the QUERY command is to be sent as a file to the command issuers card reader in NETDATA format.

FOR Specifies that the user ID whose backup data is to be queried is the specified user ID rather than the user ID of the command issuer.

This is only used in the case where one user ID owns another, i.e. is authorized to issue commands on behalf of that user ID. This option is not allowed if an SFS "dirid" is specified since the user ID is taken from the dirid.

LEVEL Specifies that only a particular backup level (*nn*) is to be displayed.

SELECT Determines which information is selected for display. SELECT CAT limits the display to objects for which there exists a catalog entry. SELECT NOCAT limits it to objects with no catalog. The default, SELECT ALL, displays all objects.

SHORT Specifies that the result of the QUERY is to be a single line message for each minidisk requested giving the format of the minidisk and whether backup will take place and at what interval. The SHORT option is useful when specifying **ALL** to reduce the number of messages received.

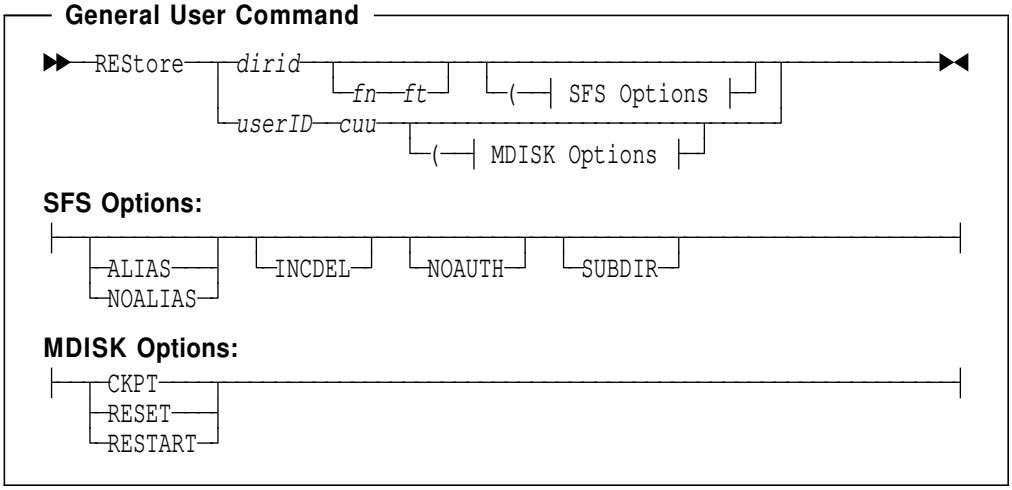
SINCE Restricts the display of files to those files which were backed up on or after the specified date. The date must be of the form ddmmmyy, where dd is the day number, mmm is the mnemonic month and yy the year number, e.g. 10JAN96. The SINCE operand must not be used if DATE is specified.

Usage notes:

1. The cuu, file name and file type may be specified using the * and **%** notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.

4.2.21 RESTORE

The RESTORE command restores a specified SFS directory, file or alias with the latest backup copy or it restores a specified minidisk.



A restore of a minidisk restores it to the state it had at the last backup.

A restore of an SFS directory includes creation of the directory, the restoration of all files within the directory and the creation of any aliases which existed at the time of the last backup cycle. If the SUBDIR option is specified, then any sub-directories are also restored. Directory and file authorizations are regranted unless the NOAUTH option is specified. The specified directory must not already exist or the command will be rejected.

If fn and ft are specified then this specifies a file or alias restore. The specified objects are restored to the directory, which must already exist. Subdirectories are not processed in this case.

SFS Operands

dirid Specifies an SFS directory to be restored, or, if fn and ft are specified, the SFS directory into which the specified files are to be restored.

fn Identifies the file name of a file or alias to be restored.

ft Identifies the file type of a file or alias to be restored.

SFS Options

ALIAS Specifies that only aliases are to be restored. This option is useful when backup exists for a file and an alias of the same name. It allows only the alias to be restored. It is also useful for restoring only aliases after all files have been restored. This ensures that the aliases can be recreated successfully.

NOALIAS Specifies that aliases are not to be restored. This option is most useful when restoring directories which may have had aliases to files in other directories which have not yet been restored. After the files have been restored, the aliases can be separately restored using the ALIAS option.

INCDEL Allows files and aliases to be restored which are marked as deleted, i.e. did not exist in the directory when it was last backed up. If this operand is not specified, such files are not restored.

NOAUTH Specifies that SFS directory and file authorizations are not to be regranted as part of the restore process.

SUBDIR Requests that all SFS sub-directories of the specified "dirid" also be restored. This option may not be used if *fn* and *ft* are specified.

Minidisk Operands

userID Specifies the user ID that owns the minidisk to be restored. The command issuer must be authorized to act for the user ID if it is not his own.

cuu Specifies the minidisk to be restored.

Minidisk Options

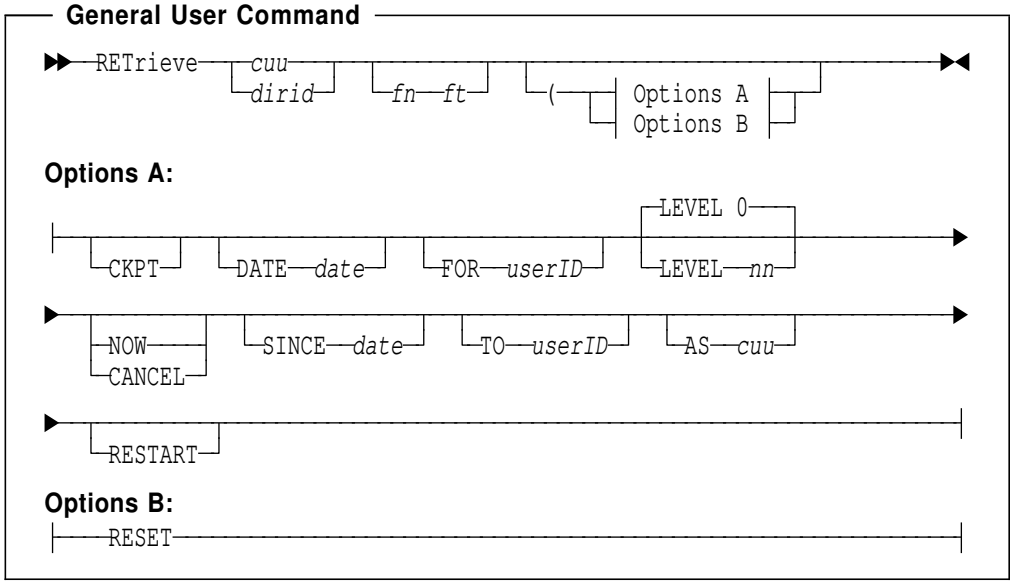
CKPT Specifies that the minidisk restore is to be checkpointed.

RESET Specifies that checkpoint information associated with this minidisk is to be erased.

RESTART Specifies that the minidisk restore is to restart from a checkpoint created by a previous restore of the same minidisk.

4.2.22 RETRIEVE

The RETRIEVE command retrieves specified data from BARS/VM backup minidisks and tapes. For CMS minidisks and SFS directories, individual files may be retrieved and these are placed in the virtual card reader. For non-CMS minidisks the data is retrieved by a DDR process directly to the specified minidisk.



Operands

cuu Defines a minidisk whose backup data is to be retrieved.

dirid Specifies an SFS directory whose backup data is to be retrieved.

fn Specifies the file name of a specific file or set of files to be retrieved from the backup of CMS data.

ft Specifies the file type of a specific file or set of files to be retrieved from the backup of CMS data.

Options A

CANCEL Eliminates a previous deferred retrieval request.

Only the first matching request is cancelled. The *fn*, *ft*, *cuu* or *dirid* and **FOR** *userID* fields are compared. Other fields, including **LEVEL**, are ignored.

CKPT Specifies that the retrieve is to be checkpointed.

DATE date Causes the command to retrieve only files which were backed up on the specified date. The date must be of the form ddmmmyy, where dd is the day number, mmm is the mnemonic month and yy the year number, e.g. 10JAN96. The DATE operand must not be used if SINCE is specified.

FOR Specifies that the user ID whose backup data is to be retrieved is the specified user ID rather than the user ID of the command issuer.

This is only used in the case where one user ID owns another, i.e. is authorized to issue commands on behalf of that user ID. This option is not allowed if an SFS "dirid" is specified since the user ID is taken from the dirid.

LEVEL n Requests a particular backup version (0 - latest level, 1 - next to latest level).

The QUERY command may be used to determine which backup levels exist. If **LEVEL** is left unspecified, the latest level (level 0) is retrieved.

NOW Requests immediate retrieval. If NOW is not specified the retrieval is deferred to the next BACKUP cycle.

SINCE date Causes the command to retrieve only files which were backed up on or after the specified date. The date must be of the form ddmmmyy, where dd is the day number, mmm is the mnemonic month and yy the year number, e.g. 10JAN96. The date is specified as it appears in the QUERY command output. The SINCE operand must not be used if DATE is specified.

TO Specifies, for a CMS RETRIEVE, that retrieved files are to be sent to the specified user ID instead of the command issuer. For a non-CMS RETRIEVE the target minidisk is that of the specified userid instead of the command issuer.

This is only allowed for an authorized user ID, the operator user ID and when the command issuer owns the target user ID, i.e. is authorized to issue commands on behalf of that user ID.

AS May only be used for a non-CMS RETRIEVE and it specifies the cuu of the target minidisk when it is to be different from the one whose backup is being retrieved.

RESTART Specifies that the RETRIEVE is to restart from a checkpoint created by a previous RETRIEVE of the same minidisk.

Options B

RESET Specifies that checkpoint information associated with this minidisk is to be erased.

Usage Notes:

1. File name and file type may be specified using the ***** and **%** notation as described in 4.1.3, “How to Read a Syntax Diagram” on page 12.
2. File name and file type must be specified with a dirid. If a cuu is specified and file name and file type are omitted, this requests that a complete non-CMS minidisk be retrieved.
3. SFS aliases cannot be retrieved. They may, however, be restored through the RESTORE command.

4.3 Sample BARS/VM Commands

This chapter shows and explains examples of frequently needed BARS/VM commands.

ADEFINE

```
VMBARS ADEFINE DEPT EDUCATN ACCNT H919722
```

This command defines the user ID of the command issuer as belonging to department EDUCATN and having an account identification of H919722.

```
VMBARS ADEFINE ACCNT H919822
```

This command changes the account identification of the command issuer to H919822. The command issuer's user ID must be already defined to BARS/VM.

```
VMBARS ADEFINE DEPT USUPPORT ACCNT D231400 (FOR USUPP1
```

This command defines the user ID USUPP1 as belonging to department USUPPORT and having an account identification of D231400. The user ID of the command issuer must have been authorized to issue this command on behalf of USUPP1.

APUT

```
VMBARS APUT 191 TEST DATA (NOW CLASS U
```

This command requests the creation of a level 0 archived copy of the file TEST DATA and that it be assigned a class of U. The file resides on the 191 minidisk of the command issuer. The operation is to be carried out immediately.

```
VMBARS APUT .CODE62 R62* ASSEMBLE (TAG REL6.2
```

This command requests archive for all files starting with the characters R62 and with a file type of ASSEMBLE which reside in the SFS directory CODE62 of the command issuer and that the archived files have the tag REL6.2 associated with them. The copying of the files is to be deferred until the time determined by the installation for deferred command processing.

```
VMBARS APUT 291 DOCREL DATA (LEVEL 1 RETAIN 6
```

This command requests archive for the file DOCREL DATA on the command issuer's 291 minidisk. The archived file is to be retained for 6 months and assigned to level 1. Any existing level 1 copy of the file will be erased. The archiving of the file is to be deferred until the time determined by the installation for deferred command processing.

```
VMBARS APUT 191 * ASSEMBLE (FROM PUSER1 FOR APOOL
```

This command requests archive for all files with a file type of ASSEMBLE which reside on the 191 disk of user PUSER1. The archived files are to be associated with user ID APOOL. The archiving of the file is to be deferred until the time determined by the installation for deferred command processing.

```
VMBARS APUT 191 * ASSEMBLE (FROM PUSER1 FOR APOOL CANCEL
```

This command cancels the deferred APUT request which was illustrated in the previous example.

AGET

```
VMBARS AGET TEST DATA (NOW
```

This command requests immediate retrieval from archive of level 0 of the file TEST DATA.

```
VMBARS AGET R62* ASSEMBLE (LEVEL 1
```

This command requests retrieval from archive of the level 1 of all archived files with file name starting with the characters R62 and with a file type of ASSEMBLE. The retrieval is to be deferred until the time determined by the installation for deferred command processing.

```
VMBARS AGET R62* ASSEMBLE (CANCEL
```

This command cancels the deferred AGET request which was illustrated in the previous example.

```
VMBARS AGET * * (TAG REL6.2
```

This command requests deferred retrieval from archive of all level 0 files which have a tag of REL6.2.

AQUERY

```
VMBARS AQUERY TEST DATA
```

This command requests information about archived levels of the file TEST DATA.

```
VMBARS AQUERY R62* ASSEMBLE (CLASS U
```

This command requests data for all files with file name starting with the characters R62 and with a file type of ASSEMBLE and which were archived with a class of U.

```
VMBARS AQUERY * * (TAG REL6.2
```

This command requests information for all files which are archived with a tag of REL6.2

```
VMBARS AQUERY * * (DEACT
```

This command requests information for all deactivated files.

AALTER

```
VMBARS AALTER TEST DATA CLASS U
```

This command requests that all archived levels of the file TEST DATA should be altered to have class U.

```
VMBARS AALTER R62CNV ASSEMBLE RETAIN 12 (LEVEL 1
```

This command requests that the file R62CNV ASSEMBLE have the retention period for its level 1 archive changed to 12 months from now. If the file level is deactivated it will be reactivated.

ADELETE

```
VMBARS ADELETE TEST DATA
```

This command requests that all archived levels of the file TEST DATA be deleted.

```
VMBARS ADELETE R62* ASSEMBLE (LEVEL 1
```

This command requests that all files with file name starting with the characters R62 and with a file type of ASSEMBLE have their level 1 archived files deleted.

ARENAME

```
VMBARS ARENAME TEST DATA TEST16 DATAOLD
```

This command renames the archived file TEST DATA to the new name TEST16 DATAOLD.

```
VMBARS ARENAME * LEVEL1 =  LEVEL2
```

This command requests a rename of all archived files with a file type of LEVEL1 to a file type of LEVEL2. The file names are to remain unchanged.

```
VMBARS ARENAME TEST DATA =  DAT2 (FOR APOOL
```

This command renames the file TEST DATA to TEST DAT2 in the archive belonging to user APOOL. You must be authorized for the user ID APOOL by BARS/VM.

DQUERY

```
VMBARS DQUERY TEST DATA
```

This command requests information about previously deleted versions of the file TEST DATA.

```
VMBARS DQUERY R62* ASSEMBLE (CLASS U
```

This command requests data for all deleted files with file name starting with the characters R62 and with a file type of ASSEMBLE and which were archived with a class of U.

```
VMBARS DQUERY * * (TAG REL6.2
```

This command requests information for all deleted files which were archived with a tag of REL6.2

```
VMBARS DQUERY * * (FILE FOR APOOL
```

This command requests information for all deleted files for user ID APOOL and requests the result of the command to be send as a file.

DRECOVER

```
VMBARS DRECOVER 10OCT94 TEST DATA
```

This command requests recovery of the level zero archived file TEST DATA which was deleted on the 10th October 1994.

```
VMBARS DRECOVER 31JUL94 R62* ASSEMBLE (LEVEL 1
```

This command requests recovery for all deleted files with file name starting with the characters R62 and with a file type of ASSEMBLE and which were deleted on 31st July 1994. The level 1 versions of these files are to be recovered.

```
VMBARS DRECOVER 1JAN95 * * (FOR APOOL
```

This command requests recovery for all deleted files which were archived under the pool name of APOOL and deleted on 1st January 1995.

BDELETE

```
VMBARS BDELETE 191 TEST DATA
```

This command deletes backup data for the file TEST DATA on the command issuers minidisk 191 with backup level 0.

```
VMBARS BDELETE 191 T* ASSEMBLE (LEVEL 1
```

This command requests deletion for all files with file name starting with the character T and with a file type of ASSEMBLE. The level 1 versions of these files are to be deleted.

```
VMBARS BDELETE 192 TEST DATA (FOR APOOL
```

This command deletes backup information from the backup of the 192 minidisk of user ID APOOL for the file TEST DATA with backup level 0. You must be authorized for the user ID APOOL.

QUERY

```
VMBARS QUERY 191 * *
```

This command queries the backup of the command issuer's 191 minidisk and lists information for all files.

```
VMBARS QUERY .NEW.FILES * EXEC
```

This command queries the backup of the command issuer's directory NEW.FILES and lists information for files with a file type of EXEC.

RESTORE

```
VMBARS RESTORE PUSER1 191
```

This command requests a restore of the 191 minidisk of user PUSER1. If the command issuer is not PUSER1, previous authorization must have been given.

```
VMBARS RESTORE .DATA (SUBDIR
```

This command requests a restore of the directory DATA and all its subdirectories. The directory must not exist. The directory tree will be created and all files, aliases and authorizations restored.

```
VMBARS RESTORE MYPOOL:ARNOLD.MY.LIFE * EXEC
```

This command restores all files with a file type of EXEC to the directory ARNOLD.MY.LIFE in filepool MYPOOL, which must already exist.

RETRIEVE

```
VMBARS RETRIEVE 191 * EXEC
```

This command requests a retrieve of all files with a file type of EXEC that were backed up from the 191 minidisk of the command issuer. The retrieve is to be deferred.

```
VMBARS RETRIEVE .REL001 * ASSEMBLE (NOW LEVEL 1
```

This command requests a retrieve of all files with a file type of ASSEMBLE that were backed up from the directory REL001 of the command issuer. The retrieve is to be done now and to return the level 1 backed up data.

4.4 Authorizing other users

4.4.1 OWNED authorization

A BARS/VM administrator may use the OWNED command to grant one user authority to issue any BARS/VM general user command on behalf of another user. For example, if user BOB is granted OWNED authority for user TIM, then BOB may use the FOR operand of any BARS/VM general user command to perform actions on TIM's backup or archive data.

Note: This mechanism gives BOB authority for all commands —i.e. it does not allow authority to be restricted to only archive or backup, or just one particular minidisk.

4.4.2 ESM authorization

An installation may choose to also allow control of access to BARS/VM data via the use of an External Security Manager (ESM), such as RACF/VM*. This option is controlled via an installation option set by the administrator. If the ESM is used, then general users can control who has authority to access their BARS/VM data themselves, without involving the administrator. Any such access is ***in addition*** to access allowed via the OWNED command.

Use of an ESM allows TIM access to BOB's BARS/VM data in a way equivalent to his access to BOB's minidisks. For example, if TIM has READ access to BOB's 191 minidisk, then TIM can QUERY or RETRIEVE files backed up from BOB's 191 minidisk. However, he cannot use BDELETE or RESTORE, which require a higher level of access authority.

Note: The ESM only controls access to the DBR QUERY command and the minidisk (not SFS) functions of the backup commands.

The following tables define the ESM authority required to issue the relevant BARS/VM commands. When no options are used, the command operates on the command issuer's own data, and hence explicit authority is not required.

frame=box rules=both
split=yes width=column.

Figure 1. Backup Command ESM authority

Operands/Options	Target authority
cuu	—
cuu (FOR uid	Control uid.cuu

frame=box rules=both
split=yes width=column.

Figure 2. BDelete Command ESM authority

Operands/Options	Target authority
cuu	—
cuu (FOR uid	Update uid.cuu

frame=box rules=both
split=yes width=column.

Figure 3. Blah Command ESM authority

Operands/Options	Target authority
cuu	—
cuu (FOR uid	Update uid.cuu

frame=box rules=both
split=yes width=column.

Figure 4. BPut Command ESM authority

Operands/Options	Source authority	Target authority
cuu	—	—
cuu (FOR uid1	—	Update uid1.cuu
cuu (FROM uid2	Read uid2.cuu	—
cuu (FOR uid1 FROM uid2	Read uid2.cuu	Update uid1.cuu

frame=box rules=both
split=yes width=column.

Figure 5. DBR Query Command ESM authority

Operands/Options	Target authority
cuu	—
cuu (FOR uid	Read uid.cuu

frame=box rules=both
split=yes width=column.

Figure 6. Query Command ESM authority

Operands/Options	Target authority
cuu	—
cuu (FOR uid	Read uid.cuu

frame=box rules=both
split=yes width=column.

Figure 7. Restore Command ESM authority

Operands/Options	Target authority
uid cuu	Update uid.cuu

scale='0.8'
frame=box rules=both
split=yes width=column.

Figure 8. Retrieve Command ESM authority

Operands/Options	Source authority	Target authority
CMS:		
cuu1	—	—
cuu1 (FOR uid1	Read uid1.cuu1	—
cuu1 (TO uid2	—	—
cuu1 (FOR uid1 TO uid2	Read uid1.cuu1	—
Non-CMS:		
cuu1	—	—
cuu1 (FOR uid1	Read uid1.cuu1	Update uid1.cuu1
cuu1 (TO uid2	—	Update uid2.cuu1
cuu1 (AS	—	—
cuu1 (FOR uid1 TO uid2	Read uid1.cuu1	Update uid2.cuu1
cuu1 (FOR uid1 AS cuu2	Read uid1.cuu1	Update uid1.cuu2
cuu1 (FOR uid1 TO uid2 AS cuu2	Read uid1.cuu1	Update uid2.cuu2
cuu1 (TO uid2 AS cuu2	—	Update uid2.cuu2

Chapter 5. A Sample Communication with BARS/VM

The following is a console log of a sample communication with BARS/VM showing the results of some of the functions:

```
vmbars query 191
R;

HDFQUE004I QUERY command started
VMBARS   0191
                CMS Format, Backup Interval 1 Day   , Latest    18Nov94
             Disk label HJF191, Number of cylinders        2, Block size 2048
HDFQUE001I QUERY command complete



vmbars query 191 * script
R;

HDFQUE004I QUERY command started
VMBARS   0191
                CMS Format, Backup Interval 1 Day   , Latest    18Nov94
             Disk label HJF191, Number of cylinders        2, Block size 2048
-CMS File Backed Up-   Version  Creation   Backup  Backup  Erasure
Filename Filetype Fm#   Number    Date    Sequence Volume    Date
ACCTIN   EXEC      1       0       21Mar94  9W974   AAAAWJ
BARTEXT  EXEC      1       0       13Jun94  AGR3H   AAAATT
BARTEXT  EXEC      1       1        8Nov92  7KRUM   AAAAS2
CHAY     EXEC      1       0       26Sep94  B28IY   AAAARJ
CHECKAUX EXEC      1       0       13Jan94  9ECAG   AAAAX8
CHECKAUX EXEC      1       1       22Dec93  9AYPQ   AAAAXB
COLOFF   EXEC      1       0       17Sep91  5HGHY   AAAAQ9    23Nov94
ECCY     EXEC      1       0       15Nov94  BCL6M   AAAAXQ
FRAGI    EXEC      1       0       17Sep91  5HGNQ   AAAAQ9
NN       EXEC      1       0       26Apr94  A54QQ   AAAAXF    22Nov94
PROFILE  EXEC      0       0       10Dec93  97U39   AAAAQ1
PROFILE  EXEC      0       1        8Oct93  8UU0S   AAAARL
TOOLDSM  EXEC      1       0       12Aug93  8LAHP   AAAAQJ
TOOLDSM  EXEC      1       1       31Jan93  7WCXJ   AAAAVM
TOOLR8   EXEC      1       0        6Aug94  ASVE3   AAAAT6
UPDTDATH EXEC      5       0       17Sep91  5HH1V   AAAAQ9
HDFQUE001I QUERY command complete



vmbars retrieve 191 profile exec (level 0
R;

HDFDCA309I Request has been deferred



vmbars display deferred
R;

VMBARS  : RETRIEVE 191 PROFILE  EXEC     For:          From:
HDFDIS004I DISPLAY command complete
```

```
vmbars retrieve 191 profile exec (level 0  cancel
R;

HDFDCA311I RETRIEVE request cancelled


vmbars retrieve 191 profile exec (level 0  now
R;

HDFRET004I RETRIEVE command started


vmbars display status users

Userid   Status           Origin    Command being processed
-------------------------------------------------------------------------
BARSD    working          VMBARS   RETRIEVE191     PROFILE EXEC    A1
BARSE    waiting for work
BARSF    logged off       BARSA
BARSG    logged off       BARSA
BARSH    logged off       BARSA
R;


vmbars display options

Backup Installation Options:
----------------------------
Maximum number of files allowed to be retrieved per day:             20
Maximum number of non-CMS cylinders allowed to be retrieved per day:404
Number of days between file erasure and backup erasure:              21
Number of days between BACKUP ERASE command and backup erasure:      14
Number of days between erasure from directory and backup erasure:    14
Number of non-CMS backup copies requested:                            2
Number of CMS file backup copies requested:                           2
Minimum number of days allowed in the BACKUP command (NON-CMS):       7
Minimum number of days allowed in the BACKUP command (CMS):           1
Default backup frequency for CMS minidisks:                           1
Archive Installation Options:
-----------------------------
Default retention period:         012
Default archive class:              A
Maximum retention period:         024

Maximum AGETs allowed per day:     40
Maximum APUTs allowed per day:     40
R;
```

```
vmbars display status queues

No commands queued
R;

PUN FILE 5567 FROM BARSD    COPY 001 NOHOLD
RDR FILE 0700 SENT FROM BARSD    PUN WAS 0107 RECS 0005 CPY  001 A NOHOLD NOKEEP
HDFRET313I RETRIEVE of VMBARS   191 complete

q r all
ORIGINID FILE CLASS RECORDS  CPY HOLD DATE  TIME     NAME      TYPE
BARSD    0700 A PUN 00000005 001 NONE 10/27 12:13:10 PROFILE   EXEC
R;
```

Index

A

AALTER command 15
  discussion 8
  examples 43
accidental erasure 9
account identification 7
acknowledgement file 3, 14
Acknowledgment
  APUT 6
ADEFINE command 16
  discussion 7
  examples 41
ADELETE command 17
  discussion 7
  examples 43
AGET command 18
  discussion 5
  examples 42
AGET ERROR file 6
alias 22, 40
  restoring 37
APUT command 19
  acknowledgment file 6
  completion of 6
  deferred processing 5
  discussion 5
  examples 41
  from temporary disk 5
  to another node 5
AQUERY command 21
  discussion 8
  examples 42
ARCHIVE
  command examples 41
  deactivation 15, 20
  general user command 15, 16, 17, 18, 19, 21, 22
  Using the 5
ARCHIVE ERROR file 6
ARENAME command 22
  discussion 8
  examples 43
Authorization
  commands 14, 45

B

backup
  general user command 23
  Using the 3
BACKUP ACK 3, 14
BACKUP command 23
  discussion 3
Backup of data
  CMS 23
Backup of data *(continued)*
  Non-CMS 23
BDELETE
  general user command 24
BDELETE command 24
  discussion 4
  examples 44
BLAH
  general user command 25
BLAH command 25
BPUT
  general user command 26
BPUT command 26
  discussion 4

C

Cancelling commands 29
class
  altering 8
  definition 8
CMS backup 23
Command queue 29, 31
Commands
  authorization 14, 45
  cancelling 29
  concurrent processing 11
  Examples 41—51
  general user 11
  general user limits 11
  holding 29
  releasing 29
  to remote nodes 19, 27
Concurrent command processing 11
CQUEUE command 29, 31

D

date
  in commands 14
DAYS 15, 20
DBR command
  general user command 29, 34
Deactivation 15, 20
deferred processing
  cancelling 6
  displaying 6
  specifying 5
department name 7
DISPLAY command
  discussion 6
  general user command 30
DISPLAY STATUS command
  discussion 8

DQUERY command
  discussion 9
  examples 43
  general user command 31
DRECOVER command 32
  discussion 9
  examples 44

E

examples
  for ARCHIVE commands 41

F

FILE option 3
FILELIST command 33

G

General user commands 9—40
  AALTER 15
  ADEFINE 16
  ADELETE 17
  AGET 18
  APUT 19
  AQUERY 21
  ARENAME 22
  authorization 14, 45
  BACKUP 23
  BDELETE 24
  BLAH 25
  BPUT 26
  CQUEUE 29
  DBR 29
  DISPLAY 30
  DQUERY 31
  DRECOVER 32
  external security manager 14, 45
  FILELIST 33
  HELP 34
  LPUT 34
  QUERY 35
  RESTORE 37
  RETRIEVE 38

H

HELP
  general user command 34
HELP command 34
Holding commands 29

I

issuing commands 11

L

LPUT command 34
LSET
  LPUT command 34

N

NETDATA 5
NOACK 20, 28
Non-CMS backup 23

O

ONEACK 28
ONEDAY 28
ownership of data 7

P

pool 7
  defining 7
  listing archived files for 8

Q

QUERY
  general user command 35
QUERY command 3, 35
  examples 45
Queue
  command 29, 31

R

reason codes 14
RECEIVE 5
recovery of erased files 9
Releasing commands 29
remote nodes 11
renaming files 8
REPORT LIST 3
Restore
  alias 4
  directory 4
  file 4
  minidisk 4
RESTORE command 37
  discussion 4
  examples 45
Retention periods 19
RETRIEVE command 38
  deferred processing 3
  examples 45
return codes 4, 14

S

sample session 49

security classification 8
SFS authorization
  restoring 37
SFS directory
  restoring 37
SSI 7, 16
Summary
  general user 13

T

tag 8, 9
  altering 8
TONODE 5, 11

U

user ID 7
  defining 7
  listing archived files for 8
userID EXEC file 6

Program Number: 5785-DJD

Printed in Hursley Park

SB11-8972-03