English

FUJITSU Software

BS2000 OSD/BC V10.0

Commands
Volume 3: DELETE-ALTERNATE-INDEX – LOGOFF

Valid for

SDF V4.7D
SDF-P-BASYS V2.5E
ASE V1.0B
BLSSERV V2.8A
CONV2PDF V1.0B
DSSM V4.3B
IMON-GPN V3.3A
JV V15.1A
POSIX-BC V10.0A
RFA V19.0A
RSO V3.6A
SECOS V5.4A
SPACEPRO V1.0A
SPOOL V4.9A

Edition June 2017

Comments… Suggestions… Corrections…

The User Documentation Department would like to know your opinion on this manual. Your feedback helps us to optimize our documentation to suit your individual needs.

Feel free to send us your comments by e-mail to:
manuals@ts.fujitsu.com

Certified documentation according to DIN EN ISO 9001:2008

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2008.

cognitas. Gesellschaft für Technik-Dokumentation mbH
www.cognitas.de

Copyright and Trademarks

DELETE-ALTERNATE-INDEX

Delete secondary indices of NK-ISAM file

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING

Function

Using the DELETE-ALTERNATE-INDEX command, the user can delete secondary indices in a file.
If the command is terminated, the user will be unable to see the specified secondary indices (output produced by SHOW-INDEX-ATTRIBUTES), but there may still be blocks in the file which have not been released. These blocks will be released when a logical copy is made of the file (e.g. using the REPAIR-DISK-FILES command or the PERCON utility program).

Format

DELETE-**ALTER**NATE-**INDEX**
FILE-**NAME** = <filename 1..54> ,**KEY-NAME** = ***ALL** / list-poss(30): <name 1..8>

Operands

FILE-NAME = <filename 1..54>
The name of the file from which the specified secondary indices are to be deleted.

KEY-NAME = *ALL / list-poss(30): <name 1..8>
Identifies the secondary index/indices which are to be deleted.
Specifying *ALL will delete all the secondary indices in the file.
One list may specify the names of up to 30 secondary indices which are to be deleted.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without errors
	32	DMS0A37	Internal system error
	32	DMS0A3A	Inconsistent control block found
	32	DMS0A42	Internal error on opening the file
	32	DMS0A43	Internal error on closing the file
	32	DMS0A4F	RDTFT error
	32	DMS0A30	Remote BS2000 system has a version < V10.0A
	32	DMS0A31	Specified catalog ID does not exist
	64	DMS0A35	Secondary index does not exist
	64	DMS0A46	Not an NK-ISAM file
	64	DMS0A4E	SHARUPD = YES was specified
	128	DMS0A49	Command interrupted
	130	DMS0A32	Specified catalog ID not available
	130	DMS0A38	Insufficient virtual memory
	130	DMS0A3C	Maximum number of secondary keys
	130	DMS0A3E	ISAM pool is overloaded

DELETE-FILE

Delete file

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION TSOS
Routing code:	$ (bei NBCONOPI=N) bzw. E (bei NBCONOPI=Y)

Function

The DELETE-FILE command can be used to delete temporary or permanent files which are cataloged under the user's own user ID, and selected according to criteria which must be specified in the command. A tape file may also be specified, but since an implicit EXPORT-FILE is executed for it, only its catalog entry will be deleted.

There are separate commands for deleting file generations and file generation groups (see DELETE-FILE-GENERATION and DELETE-FILE-GROUP).

The DELETE-FILE command has 4 main functions:

1. Selection

With the SELECT operand, the user specifies which files or catalog entries are to be processed. The attributes stored in the catalog entries are used as selection criteria. These selection criteria correspond with those of the SHOW-FILE-ATTRIBUTES command.

2. Action parameters

The OPTION operand permits the user to control the internal execution of the DELETE-FILE processing. The scope of the deletions can be specified, i.e. the handling of the catalog entry, the data contained in the file, and the reserved storage space.

3. Verifying command execution

The DIALOG-CONTROL, OUTPUT and SUPPRESS-ERRORS operands allow the user to define his own user interface, within certain limits. For example, he may use interactive prompts, request the names of deleted files on SYSOUT, or have DMS errors ignored (for error handling in procedures).

4. File protection

The IGNORE-PROTECTION and PASSWORDS-TO-IGNORE operands save the user from having to enter the MODIFY-FILE-ATTRIBUTES or ADD-PASSWORD commands when files which are protected against write access by passwords, retention periods, or active access control need to be deleted.

Privileged functions

The privileged functions for systems support staff (TSOS privilege) include specification of temporary or cataloged files of any user ID, the ability to ignore protection attributes of the file being deleted and, in the case of a user ID change, to branch to guided dialog. Systems support staff can use wildcards in the user ID. If no catalog ID is specified in such cases, the specified files are deleted from the associated default pubsets of the relevant users.
By default, systems support (TSOS privilege) is a co-owner of all the files (and can therefore delete files under any user ID). When SECOS is used, this co-ownership can be restricted for permanent files.
Systems support can generate files under any user ID (TSOS privilege).In conjunction with the SECOS software product a user can allow other user IDs to act as co-owners. Co-owners of a user ID are also allowed to delete files under that ID.

Overview of functions

	Function / Meaning	Level 1 operands	Level 2/3 operands
3-21	Name of file to be deleted	FILE-NAME	
3-22	Selection criteria for the files to be deleted	SELECT= *BY-ATTRIBUTES	
	Date specification		
3-22	– creation date – Time		CREATION-DATE TIME
3-26	– Expiration date (implicit retention period) – Time		EXPIRATION-DATE TIME
3-31	– date of last access – Time		LAST-ACCESS-DATE TIME
3-35	– Date of last write access – Time		LAST-CHANGE-DATE TIME
3-55	– Date when file is freed for deletion – Time		FREE-FOR-DELETION TIME
3-50	– Number of accesses to the file		ACCESS-COUNTER
3-38	Storage space		
3-38	– Type of volume		SUPPORT
	– Pubset and Net-Storage – Private disk – tape		=*PUBLIC-DISK =*PRIVATE-DISK =*TAPE
3-59	– Volume set		VOLUME-SET
3-39	– Volume identifier (VSN)		VOLUME
3-60	– Availability		AVAILABILITY
3-39	– Storage type		STORAGE-TYPE
3-39	– File type (on Net-Storage)		FILE-TYPE
3-59	– Storage class		STORAGE-CLASS
3-40	– Reserved storage space		SIZE
3-41	– Number of extents		NUMBER-OF-EXTENTS
3-41	– Size of reserved storage space not yet used		NUMBER-OF-FREE-PAGES
3-42	– Number of the last page used (last page pointer)		HIGHEST-USED-PAGE
3-43	– Number of logical blocks in a tape file		BLOCK-COUNTER

Table 44: Overview of DELETE-FILE command functions (Part 1 of 4)

	Function / Meaning	Level 1 operands	Level 2/3 operands
	File security/file protection		
3-43	– Access		ACCESS
3-44	– Shareability		USER-ACCESS
3-47	– Basic access control list		BASIC-ACL
3-49	– Password protection		PASSWORD
3-50	– Highest activated access control		PROTECTION-ACTIVE
3-43	– Password protection		GUARDS
3-53	– protection against release of storage space		SPACE-RELEASE-LOCK
3-45	– BACKUP level		BACKUP-CLASS
3-60	– Work file attribute		WORK-FILE
3-53	– Type of file		TYPE-OF-FILES
	– Files only		FILE
	– PLAM libraries only		PLAM-LIBRARY
3-60	– preferred file format		FILE-PREFORMAT
3-61	– Type of file encryption		ENCRYPTION
3-45	– File format		BLOCK-CONTROL-INFO
3-44	– Access method at creation		FILE-STRUCTURE
3-51	– Code table (CCS)		CODED-CHARACTER-SET
3-53	– Performance attributes		IO-ATTRIBUTES
	– Performance requirements		PERFORMANCE
	– Type of I/O operations		USAGE
3-54	– Time of ensuring data consistency after write operations		DISK-WRITE
3-51	– Status of the file		STATUS
	– Closed		CLOSED-OUTPUT
	– Processed in a cache		CACHED
	– Not closed properly and not yet reconstructed		REPAIR-NEEDED
	– Locked or opening allowed		OPEN-ALLOWED
	– Contains defective disk blocks		DEFECT-REPORTED
	– Modified data not written out to disk when closed		CACHE-NOT-SAVED

Table 44: Overview of DELETE-FILE command functions (Part 2 of 4)

	Function / Meaning	Level 1 operands	Level 2/3 operands
3-59	– user information		USER-INFORMATION
3-59	– systems support information		ADM-INFORMATION
	HSMS-related file attributes		
3-46	– Ability to migrate from processing level (S0) to background level (S1 or S2)		MIGRATE
3-60	– Ability to migrate within processing level (S0)		S0-MIGRATION
3-46	– Storage level		STORAGE-LEVEL
3-59	– Management class		MANAGEMENT-CLASS
	Deletion parameters	OPTION	
3-61	– Delete the catalog entry and release storage space	=*ALL	
3-61	– Release storage space only, retain catalog entry	=*SPACE	
3-61	– Logical deletion: delete the data-related attributes of the file, change the catalog entry accordingly, retain the storage space allocation	=*DATA	
3-62	– Logical deletion as with *DATA, but the data-related attributes of the file are retained	=*DATA-KEEP-ATTRIBUTES	
3-62	– Delete the catalog entry, release the storage space and overwrite its contents	=*DESTROY-ALL	
3-62	Mount request	MOUNT	
	Control parameters	DIALOG-CONTROL	
3-63	– User intervention not permitted (default setting for procedures and batch mode)	=*NO	
3-64	– Dialog when more than one file is referenced if the file name specified is not fully qualified (default setting for interactive mode)	=*MORE-THAN-ONE-FILE	

Table 44: Overview of DELETE-FILE command functions (Part 3 of 4)

	Function / Meaning	Level 1 operands	Level 2/3 operands
3-63	– Dialog when an error occurs	=*ERROR	
3-64	– Dialog when catalog ID changes	=*CATALOG-CHANGE	
3-63	– For each file selected, there will be an interactive dialog with the user to determine whether the current DELETE-FILE command should be processed or not	=*FILE-CHANGE	
3-64	– Dialog when user ID changes	=*USER-ID-CHANGE	
3-64	Message for successfully deleted files	OUTPUT	
	– Suppress	=*NO	
	– Output to SYSOUT	=*SYSOUT	
3-64	Ignore protection attributes	IGNORE-PROTECTION	
	– Ignore write protection provided by ACCESS=READ, BASIC-ACL or GUARDS	=*ACCESS	
	– Ignore retention periods	=*EXPIRATION-DATE	
	– Ignore password protection	=*WRITE-PASSWORD / *READ-PASSWORD / *EXEC-PASSWORD	
3-65	Ignore password protection providedby specified passwords	PASSWORDS-TO-IGNORE	
3-65	Suppress error situations	SUPPRESS-ERRORS	

Table 44: Overview of DELETE-FILE command functions (Part 4 of 4)

Format

```
DELETE-FILE                                                              Alias: DLF

FILE-NAME = *DUMMY / <filename 1..54 without-gen with-wild(80)>
,SELECT = *ALL / [*BY-ATTRIBUTES](...)
    [*BY-ATTRIBUTES](...)
    |   CREATION-DATE = *ANY / *NONE / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) /
    |                   <date>(...) / *INTERVAL(...)
    |       *TODAY(...)
    |       |   TIME = *ANY / [*INTERVAL](...)
    |       |       [*INTERVAL](...)
    |       |       |   FROM = 00:00:00 / <time>
    |       |       |  ,TO = 23:59:59 / <time>
    |       *YESTERDAY(...)
    |       |   TIME = *ANY / [*INTERVAL](...)
    |       |       [*INTERVAL](...)
    |       |       |   FROM = 00:00:00 / <time>
    |       |       |  ,TO = 23:59:59 / <time>
    |       <integer -99999..991231>(...)
    |       |   TIME = *ANY / [*INTERVAL](...)
    |       |       [*INTERVAL](...)
    |       |       |   FROM = 00:00:00 / <time>
    |       |       |  ,TO = 23:59:59 / <time>
    |       <date>(...)
    |       |   TIME = *ANY / [*INTERVAL](...)
    |       |       [*INTERVAL](...)
    |       |       |   FROM = 00:00:00 / <time>
    |       |       |  ,TO = 23:59:59 / <time>
```

(Part 1 of 11)

```
*INTERVAL(...)

   FROM = *EARLIEST / *TODAY(...) / *YESTERDAY(...) /
                <integer -99999..991231>(...) / <date>(...)

      *TODAY(...)
         TIME = 00:00:00 / <time>

      *YESTERDAY(...)
         TIME = 00:00:00 / <time>

      <integer -99999..991231>(...)
         TIME = 00:00:00 / <time>

      <date>(...)
         TIME = 00:00:00 / <time>

  ,TO = *TODAY (...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)

      *TODAY(...)
         TIME = 23:59:59 / <time>

      *YESTERDAY(...)
         TIME = 23:59:59 / <time>

      <integer -99999..991231>(...)
         TIME = 23:59:59 / <time>

      <date>(...)
         TIME = 23:59:59 / <time>

,EXPIRATION-DATE = *ANY / *NONE / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
                   <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)

   *TOMORROW(...)

      TIME = *ANY / [*INTERVAL](...)

         [*INTERVAL](...)

            FROM = 00:00:00 / <time>
           ,TO = 23:59:59 / <time>
   *TODAY(...)

      TIME = *ANY / [*INTERVAL](...)

         [*INTERVAL](...)

            FROM = 00:00:00 / <time>
           ,TO = 23:59:59 / <time>
```

(Part 2 of 11)

```
      *YESTERDAY(...)
         | TIME = *ANY / [*INTERVAL](...)
         |
         |   [*INTERVAL](...)
         |
         |      | FROM = 00:00:00 / <time>
         |      |,TO = 23:59:59 / <time>

      <integer -99999..991231>(...)
         | TIME = *ANY / [*INTERVAL](...)
         |
         |   [*INTERVAL](...)
         |
         |      | FROM = 00:00:00 / <time>
         |      |,TO = 23:59:59 / <time>

      <date>(...)
         | TIME = *ANY / [*INTERVAL](...)
         |
         |   [*INTERVAL](...)
         |
         |      | FROM = 00:00:00 / <time>
         |      |,TO = 23:59:59 / <time>

      *INTERVAL(...)
         | FROM = EARLIEST / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
         |             <integer -99999..991231>(...) / <date>(...)
         |
         |   *TOMORROW(...)
         |      | TIME = 00:00:00 / <time>
         |
         |   *TODAY(...)
         |      | TIME = 00:00:00 / <time>
         |
         |   *YESTERDAY(...)
         |      | TIME = 00:00:00 / <time>
         |
         |   <integer -99999..991231>(...)
         |      | TIME = 00:00:00 / <time>
         |   <date>(...)
         |      | TIME = 00:00:00 / <time>
```

(Part 3 of 11)

```
        |       | ,TO = LATEST / TODAY(...) / *LATEST / *TOMORROW(...) / *YESTERDAY(...) /
        |       |         <integer -99999..991231>(...) / <date>(...)
        |       |
        |       |     *TODAY(...)
        |       |       |   TIME = 23:59:59 / <time>
        |       |
        |       |     *TOMORROW(...)
        |       |       |   TIME = 23:59:59 / <time>
        |       |
        |       |     *YESTERDAY(...)
        |       |       |   TIME = 23:59:59 / <time>
        |       |
        |       |     <integer -99999..991231>(...)
        |       |       |   TIME = 23:59:59 / <time>
        |       |
        |       |     <date>(...)
        |       |       |   TIME = 23:59:59 / <time>
        |
        | ,LAST-ACCESS-DATE = *ANY / *NONE / *TODAY(...) / *YESTERDAY(...) /
        |                      <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)
        |
        |    *TODAY(...)
        |       |   TIME = *ANY / [*INTERVAL](...)
        |       |
        |       |     [*INTERVAL](...)
        |       |       |   FROM = 00:00:00 / <time>
        |       |       |  ,TO = 23:59:59 / <time>
        |
        |    *YESTERDAY(...)
        |       |   TIME = *ANY / [*INTERVAL](...)
        |       |
        |       |     [*INTERVAL](...)
        |       |       |   FROM = 00:00:00 / <time>
        |       |       |  ,TO = 23:59:59 / <time>
        |
        |    <integer -99999..991231>(...)
        |       |   TIME = *ANY / [*INTERVAL](...)
        |       |
        |       |     [*INTERVAL](...)
        |       |       |   FROM = 00:00:00 / <time>
        |       |       |  ,TO = 23:59:59 / <time>
        |    <date>(...)
        |       |   TIME = *ANY / [*INTERVAL](...)
        |       |
        |       |     [*INTERVAL](...)
        |       |       |   FROM = 00:00:00 / <time>
        |       |       |  ,TO = 23:59:59 / <time>
```

(Part 4 of 11)

```
    *INTERVAL(...)
        FROM = *EARLIEST / *TODAY(...) / *YESTERDAY(...) /
                    <integer -99999..991231>(...) / <date>(...)
            *TODAY(...)
                TIME = 00:00:00 / <time>
            *YESTERDAY(...)
                TIME = 00:00:00 / <time>
            <integer -99999..991231>(...)
                TIME = 00:00:00 / <time>
            <date>(...)
                TIME = 00:00:00 / <time>
       ,TO = *TODAY (...) / *YESTERDAY(...) / <integer -99999..991231>(...) /
                <date>(...)
            *TODAY(...)
                TIME = 23:59:59 / <time>
            *YESTERDAY(...)
                TIME = 23:59:59 / <time>
            <integer -99999..991231>(...)
                TIME = 23:59:59 / <time>
            <date>(...)
                TIME = 23:59:59 / <time>
   ,LAST-CHANGE-DATE = *ANY / *NONE / *TODAY(...) / *YESTERDAY(...) / <integer -99999..0>(...) /
                            <date>(...) / *INTERVAL(...)
    *TODAY(...)
        TIME = *ANY / [*INTERVAL](...)
            [*INTERVAL](...)
                FROM = 00:00:00 / <time>
               ,TO = 23:59:59 / <time>
    *YESTERDAY(...)
        TIME = *ANY / [*INTERVAL](...)
            [*INTERVAL](...)
                FROM = 00:00:00 / <time>
               ,TO = 23:59:59 / <time>
```

(Part 5 of 11)

```
      <integer -99999..0>(...)
           TIME = *ANY / [*INTERVAL](...)
              [*INTERVAL](...)
                   FROM = 00:00:00 / <time>
                  ,TO = 23:59:59 / <time>
      <date>(...)
           TIME = *ANY / [*INTERVAL](...)
              [*INTERVAL](...)
                   FROM = 00:00:00 / <time>
                  ,TO = 23:59:59 / <time>
      *INTERVAL(...)
           FROM = *EARLIEST / <integer -99999..0>(...) / <date>(...) / *TODAY(...) /
                       *YESTERDAY(...)
              <integer -99999..0>(...)
                  ,TIME = 00:00:00 / <time>
              <date>(...)
                   TIME = 00:00:00 / <time>
              *TODAY(...)
                   TIME = 00:00:00 / <time>
              *YESTERDAY(...)
                   TIME = 00:00:00 / <time>
          ,TO = *TODAY(...) / <integer -99999..0>(...) / <date>(...) / *YESTERDAY(...)
              *TODAY(...)
                   TIME = 23:59:59 / <time>
              <integer -99999..0>(...)
                   TIME = 23:59:59 / <time>
              <date>(...)
                   TIME = 23:59:59 / <time>
              *YESTERDAY(...)
                   TIME = 23:59:59 / <time>
  ,SUPPORT = *ANY / list-poss(3): *PUBLIC-DISK / *PRIVATE-DISK / *TAPE
   STORAGE-TYPE = *ANY / *PUBLIC-SPACE / *NET-STORAGE(...)
      *NET-STORAGE(...)
           FILE-TYPE = *ANY / *BS2000 / *NODE-FILE
```

(Part 6 of 11)

```
,VOLUME = *ANY / <vsn 1..6>
,SIZE = *ANY / *FREESIZE / <integer 0..2147483647> / [*INTERVAL](...)
   [*INTERVAL](...)
      FROM = 0 / <integer 0..2147483647>
     ,TO = 2147483647 / <integer 0..2147483647>
,NUMBER-OF-EXTENTS = *ANY / <integer 0..65535> / [*INTERVAL](...)
   [*INTERVAL](...)
      FROM = 0 / <integer 0..65535>
     ,TO = 65535 / <integer 0..65535>
,NUMBER-OF-FREE-PAGES = *ANY / *SIZE / <integer 0..2147483647> / *ALL-ALLOCATED /
                        [*INTERVAL](...)
   [*INTERVAL](...)
      FROM = 0 / <integer 0..2147483647>
     ,TO = 2147483647 / <integer 0..2147483647>
,HIGHEST-USED-PAGE = *ANY / <integer 0..2147483647> / [*INTERVAL](...)
   [*INTERVAL](...)
      FROM = 0 / <integer 0..2147483647>
     ,TO = 2147483647 / <integer 0..2147483647>
,BLOCK-COUNTER = *ANY / <integer 0..2147483647> / [*INTERVAL](...)
   *INTERVAL(...)
      FROM = 0 / <integer 0..2147483647>
     ,TO = 2147483647 / <integer 0..2147483647>
,ACCESS = *ANY / *READ / *WRITE
,PASSWORD = *ANY / list-poss(4): *NONE / *READ-PASSWORD / *WRITE-PASSWORD /
             *EXEC-PASSWORD
,USER-ACCESS = *ANY / list-poss(3): *OWNER-ONLY / *ALL-USERS / *SPECIAL
,FILE-STRUCTURE = *ANY / list-poss(5): *PAM / *SAM / *ISAM / *BTAM / *NONE
,BACKUP-CLASS = *ANY / list-poss(5): *A / *B / *C / *D / *E
,BLOCK-CONTROL-INFO = *ANY / list-poss(9): *NONE / *NO / *WITHIN-DATA-BLOCK /
                      *WITHIN-DATA-2K-BLOCK / *WITHIN-DATA-4K-BLOCK /
                      *PAMKEY / *NK / *NK2 / *NK4
```

(Part 7 of 11)

```
,MIGRATE = *ANY / list-poss(3): *ALLOWED / *INHIBITED / *FORBIDDEN

,STORAGE-LEVEL = *ANY / list-poss(3): *S0 / *S1 / *S2

,BASIC-ACL = *ANY / *NONE / *YES / [*PARAMETERS](...)

   [*PARAMETERS](...)

        OWNER = *ANY / *NO-ACCESS / [*PARAMETERS](...)

           [*PARAMETERS](...)

                READ = *ANY / *NO / *YES
               ,WRITE = *ANY / *NO / *YES
               ,EXEC = *ANY / *NO / *YES

       ,GROUP = *ANY / *NO-ACCESS / [*PARAMETERS](...)

           [*PARAMETERS](...)

                READ = *ANY / *NO / *YES
               ,WRITE = *ANY / *NO / *YES
               ,EXEC = *ANY / *NO / *YES

       ,OTHERS = *ANY / *NO-ACCESS / [*PARAMETERS](...)

           [*PARAMETERS](...)

                READ = *ANY / *NO / *YES
               ,WRITE = *ANY / *NO / *YES
               ,EXEC = *ANY / *NO / *YES

,GUARDS = *ANY / *YES / *NO / [*PARAMETERS](...)

   [*PARAMETERS](...)

        READ = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>

       ,WRITE = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>

       ,EXEC = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>

,PROTECTION-ACTIVE = *ANY / list-poss(3): *LEVEL-0 / *LEVEL-1 / *LEVEL-2

,ACCESS-COUNTER = *ANY / <integer 0..2147483647> / [*INTERVAL](...)

   [*INTERVAL](...)

        FROM = 0 / <integer 0..2147483647>
       ,TO = 2147483647 / <integer 0..2147483647>

,CODED-CHARACTER-SET = *ANY / *NONE / <name 1..8>
```

(Part 8 of 11)

```
,STATUS = *ANY / [*PARAMETERS](...)

   [*PARAMETERS](...)

        CLOSED-OUTPUT = *ANY / *YES / *NO
       ,CACHED = *ANY / *YES / *NO
       ,REPAIR-NEEDED = *ANY / *YES / *NO
       ,OPEN-ALLOWED = *ANY / *YES / *NO
       ,DEFECT-REPORTED = *ANY / *YES
       ,CACHE-NOT-SAVED = *ANY / *YES

,TYPE-OF-FILES = *ANY / *FILE / *PLAM-LIBRARY

,SPACE-RELEASE-LOCK = *ANY / *NO / *YES

,IO-ATTRIBUTES = *ANY / [*PARAMETERS](...)

   [*PARAMETERS](...)

        PERFORMANCE = *ANY / list-poss(3): *STD / *HIGH / *VERY-HIGH
       ,USAGE = *ANY / list-poss(3): *READ-WRITE / *WRITE / *READ

,DISK-WRITE = *ANY / *IMMEDIATE / *BY-CLOSE

,FREE-FOR-DELETION = *ANY / *NONE / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
                     <integer -99999..99999>(...) / <date>(...) / *INTERVAL(...)

   *TOMORROW(...)

        TIME = *ANY / *INTERVAL(...)

           *INTERVAL(...)

                FROM = 00:00:00 / <time>
               ,TO = 23:59:59 / <time>

   *TODAY(...)
        TIME = *ANY / *INTERVAL(...)

           *INTERVAL(...)

                FROM = 00:00:00 / <time>
               ,TO = 23:59:59 / <time>

   *YESTERDAY(...)
        TIME = *ANY / *INTERVAL(...)

           *INTERVAL(...)

                FROM = 00:00:00 / <time>
               ,TO = 23:59:59 / <time>
```

(Part 9 of 11)

```
        <integer -99999..99999>(...)
              TIME = *ANY / *INTERVAL(...)

                 *INTERVAL(...)

                        FROM = 00:00:00 / <time>
                       ,TO = 23:59:59 / <time>

        <date>(...)
              TIME = *ANY / *INTERVAL(...)

                 *INTERVAL(...)

                        FROM = 00:00:00 / <time>
                       ,TO = 23:59:59 / <time>

        *INTERVAL(...)
              FROM = *EARLIEST / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
                          <integer -99999..99999>(...) / <date>(...)
                 *TOMORROW(...)
                        TIME = 00:00:00 / <time>

                 *TODAY(...)
                        TIME = 00:00:00 / <time>

                 *YESTERDAY(...)
                        TIME = 00:00:00 / <time>

                 <integer -99999..99999>(...)
                        TIME = 00:00:00 / <time>

                 <date>(...)
                        TIME = 00:00:00 / <time>

             ,TO = *LATEST / *TODAY(...) / *TOMORROW(...) / *YESTERDAY(...) /
                      <integer -99999..99999>(...) / <date>(...)

                 *TODAY(...)
                        TIME = 23:59:59 / <time>

                 *TOMORROW(...)
                        TIME = 23:59:59 / <time>

                 *YESTERDAY(...)
                        TIME = 23:59:59 / <time>

                 <integer -99999..99999>(...)
                        TIME = 23:59:59 / <time>

                 <date>(...)
                        TIME = 23:59:59 / <time>
```

(Part 10 of 11)

,STORAGE-CLASS = *ANY / *NONE / <composed-name 1..8>

,MANAGEMENT-CLASS = *ANY / *NONE / <composed-name 1..8>

,ADM-INFORMATION = *ANY / *NONE / <c-string 1..8 with-low>

,USER-INFORMATION = *ANY / *NONE / <c-string 1..8 with-low>

,VOLUME-SET = *ANY / <cat-id 1..4>

,AVAILABILITY = *ANY / list-poss(2): *STD / *HIGH

,S0-MIGRATION = *ANY / list-poss(2): *ALLOWED / *FORBIDDEN

,WORK-FILE = *ANY / *NO / *YES

,FILE-PREFORMAT = *ANY / list-poss(4): *NONE / *K / *NK2 / *NK4

,ENCRYPTION = *ANY / list-poss(3): *NONE / *AES / *DES

,OPTION = *ALL / *SPACE / *DATA / *DATA-KEEP-ATTRIBUTES / *DESTROY-ALL

,MOUNT = *FIRST-DISK / *ALL-DISKS

,DIALOG-CONTROL = *STD / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE /
*CATALOG-CHANGE / *USER-ID-CHANGE

,OUTPUT = *STD / *NO / *SYSOUT

,IGNORE-PROTECTION = *NONE / list-poss(5): *ACCESS / *EXPIRATION-DATE / *WRITE-PASSWORD /
*READ-PASSWORD / *EXEC-PASSWORD

,PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> /
<integer -2147483648..2147483647>

,SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>

(Part 11 of 11)

Operands

FILE-NAME = *DUMMY / <filename 1..54 without-gen with-wild(80)>
The name of the file to be deleted. Only the user’s own user ID or a user ID for which the user is co-owner may be specified. If the wildcard “*” is the first character in a wildcard string, it must be entered twice.

FILE-NAME = *DUMMY
Designates the dummy file *DUMMY, which “always exists”, and which satisfies all selection criteria. Any other operands, except for the control parameters DIALOG-CONTROL, OUTPUT and SUPPRESS-ERRORS will be checked for their formal correctness, but otherwise ignored. If *DUMMY is specified, no catalog or data access is necessary. The main use of *DUMMY is in testing procedures.

SELECT = *ALL
All the files in the set specified by FILE-NAME are selected.

SELECT = *BY-ATTRIBUTES(...)
Restricts the files in the set specified in FILE-NAME to files which satisfy the criteria which follow. The default values, *ANY and ANY, each indicate that the file set is not to be restricted to any particular values for the attribute concerned.

CREATION-DATE = *ANY / *NONE / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)
The user can select the files to be deleted by their creation date. On a given date, the selection can be restricted to a time interval within which the file was created (see the relevant TIME operands in each structure).
On deleting file generation groups and file generations see the DELETE-FILE-GROUP and DELETE-FILE-GENERATION commands. See also the *CRE-DATE* and *CRE-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

CREATION-DATE = *ANY
The creation date is not to be used as a selection criterion.

CREATION-DATE = *NONE
Only those files which have the value NONE entered in the CREATION-DATE field in their catalog entry are deleted, i.e. files which have never been opened.

CREATION-DATE = *TODAY(...)
Only those files which have today's date entered in the CREATION-DATE field in their catalog entry are deleted.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files for deletion to a time interval related to the specified creation date.

TIME = *INTERVAL(...)
Deletes only those files which were created on the specified day within the specified time interval that follows.

FROM = 00:00:00 / <time>
Selects files for which the time of creation ≥ the specified time.

TO = 23:59:59 / <time>
Selects files for which the time of creation ≤ the specified time.

CREATION-DATE = *YESTERDAY(...)
Only those files which have yesterday’s date entered in the CREATION-DATE field in their catalog entry are deleted.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified LAST-CHANGE-DATE.

TIME = *INTERVAL(...)
Deletes only those files which were created on the specified day within the specified time interval that follows.

FROM = 00:00:00 / <time>
Selects files for which the time of creation ≥ the specified time.

TO = 23:59:59 / <time>
Selects files for which the time of creation ≤ the specified time.

CREATION-DATE = <integer -99999..991231>(...)
Only those files which have the specified date entered in the CREATION-DATE field in their catalog entry are deleted. Here, the user can specify the creation date in either of two ways:

1. as an *absolute date value*: (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value*: (6 digits, with preceding sign) the number of days from today’s date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ ±0)

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified creation date.

TIME = *INTERVAL(...)
Deletes only those files which were created on the specified day within the specified time interval that follows.

FROM = 00:00:00 / <time>
Selects files for which the time of creation ≥ the specified time.

TO = 23:59:59 / <time>
Selects files for which the time of creation ≤ the specified time.

CREATION-DATE = <date>(...)
Only those files which have the specified date entered in the CREATION-DATE field in their catalog entry are deleted. The user can specify the creation date in the form [yy]yy-mm-dd.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified creation date.

TIME = *INTERVAL(...)
Deletes only those files which were created on the specified day within the specified time interval that follows.

FROM = 00:00:00 / <time>
Selects files for which the time of creation ≥ the specified time.

TO = 23:59:59 / <time>
Selects files for which the time of creation ≤ the specified time.

CREATION-DATE = *INTERVAL(...)
Only those files which were created within the specified time period are deleted. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified using <integer> values). It is also possible to specify limits using the operands FROM (lower limit) and TO (upper limit). Whichever of the operands is not specified will be replaced by the default value for use as the limit of the range. The use of range limits for deletion can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = *EARLIEST / *TODAY(...) / *YESTERDAY(...) /
<integer -99999..991231>(...) / <date>(...)
Only those files which have been created since the specified date (i.e. CREATION-DATE ≥ specified date) are deleted. If EARLIEST is specified, the system will use the earliest possible date.

FROM = *TODAY(...)
Deletes only those files which were created after the specified date (i.e. files for which CREATION-DATE ≥ current date).

TIME = 00:00:00 / <time>
Time on the specified date. All files created at or after the specified time are selected.

FROM = *YESTERDAY(...)
Deletes only those files which were created after the specified date (i.e. files for which CREATION-DATE ≥ date of preceding day).

TIME = 00:00:00 / <time>
Time on the specified date. All files created at or after the specified time are selected.

FROM = <integer -99999..991231>(...)
Deletes only those files which were created after the specified date (i.e. files for which CREATION-DATE ≥ specified date).

TIME = 00:00:00 / <time>
Time on the specified date. All files created at or after the specified time are selected.

FROM = <date>(...)
Deletes only those files which were created after the specified date (i.e. files for which CREATION-DATE ≥ specified date).

TIME = 00:00:00 / <time>
Time on the specified date. All files created at or after the specified time are selected.

TO = *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)
Deletes only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ specified date).

TO = *TODAY(...)
Deletes only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ current date).

TIME = 23:59:59 / <time>
Time on the specified date. All files created at or before the specified time are selected.

TO = *YESTERDAY(...)
Deletes only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ date of preceding day).

TIME = 23:59:59 / <time>
Time on the specified date. All files created at or before the specified time are selected.

TO = <integer -99999..991231>(...)
Deletes only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ specified date).

TIME = 23:59:59 / <time>
Time on the specified date. All files created at or before the specified time are selected.

TO = <date>(...)
Deletes only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ specified date).

TIME = 23:59:59 / <time>
Time on the specified date. All files created at or before the specified time are selected.

EXPIRATION-DATE = *ANY / *NONE / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)
The user can select the files to be deleted by their release date (expiration date). The selection can be restricted for a specified date to a time interval in which the last file update occurred (see the TIME operand for each structure). Note that at present the time 00:00:00 is always entered in the file catalog as the time of the expiration date. On deleting file generation groups and file generations see the DELETE-FILE-GROUP and DELETE-FILE-GENERATION commands. See also the *EXPIR-DATE* and *EXPIR-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

EXPIRATION-DATE = *ANY
The expiration date is not to be used as a selection criterion.

EXPIRATION-DATE = *NONE
Only those files which have no expiration date (i.e. the value NONE) entered in the EXPIRATION-DATE field in their catalog entry are deleted.

EXPIRATION-DATE = *TOMORROW(...)
Deletes only those files which have tomorrows's date entered as the EXPIRATION-DATE in their catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes only those files whose expiration date lies within the specified time interval that follows.

FROM = <u>00:00:00</u> / <time>
Selects files for which the time of expiration ≥ the specified time.

TO = <u>23:59:59</u> / <time>
Selects files for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = *TODAY(...)
Only those files which have today’s date entered as the EXPIRATION-DATE in their catalog entry are deleted.

TIME = <u>*ANY</u> / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes only those files whose expiration date lies within the specified time interval that follows.

FROM = <u>00:00:00</u> / <time>
Selects files for which the time of expiration ≥ the specified time.

TO = <u>23:59:59</u> / <time>
Selects files for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = *YESTERDAY(...)
Only those files which have yesterday’s date entered as the EXPIRATION-DATE in their catalog entry are deleted.

TIME = <u>*ANY</u> / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes only those files whose expiration date lies within the specified time interval that follows.

FROM = <u>00:00:00</u> / <time>
Selects files for which the time of expiration ≥ the specified time.

TO = <u>23:59:59</u> / <time>
Selects files for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = <integer -99999..991231>(...)
Only those files which have the specified date entered as the EXPIRATION-DATE in their catalog entry are deleted. Here, the user can specify the expiration date in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value:* (6 digits with preceding sign), the number of days from today's date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ ±0 or TOMORROW ≙ +1)

TIME = _*ANY_ / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes only those files whose expiration date lies within the specified time interval that follows.

FROM = _00:00:00_ / <time>
Selects files for which the time of expiration ≥ the specified time.

TO = _23:59:59_ / <time>
Selects files for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = <date>(...)
Only those files which have the specified date entered as the EXPIRATION-DATE in their catalog entry are deleted. The user can specify the expiration date in the form [yy]yy-mm-dd.

TIME = _*ANY_ / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes only those files whose expiration date lies within the specified time interval that follows.

FROM = _00:00:00_ / <time>
Selects files for which the time of expiration ≥ the specified time.

TO = _23:59:59_ / <time>
Selects files for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = *INTERVAL(...)
Only those files for which the expiration date lies within the specified time period, i.e. files for which the retention period ends within the specified time period, will be deleted. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified for EXPIRATION-DATE = <integer...>. It is also possible to specify limits using the operands FROM (lower limit) and TO (upper limit). Whichever of the operands is not specified will be replaced by the default value for use as the limit of the range. The use of range limits for deletion can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = <u>EARLIEST</u> / TOMORROW(...) / TODAY(...) / YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)
If EARLIEST is specified, the system will use the earliest possible date. Only those files for which the retention period expires on or after the specified date (EXPIRATION-DATE ≥ specified date) are deleted.

FROM = *TOMORROW(...)
Deletes only those files for which the EXPIRATION-DATE ≥ the date of next day.

TIME = <u>00:00:00</u> / <time>
Time on the specified date. All files that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = TODAY(...)
Deletes only those files for which the EXPIRATION-DATE ≥ current date.

TIME = <u>00:00:00</u> / <time>
Time on the specified date. All files that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = *YESTERDAY(...)
Deletes only those files for which the EXPIRATION-DATE ≥ date of preceding day.

TIME = <u>00:00:00</u> / <time>
Time on the specified date. All files that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = <integer -99999..991231>(...)
Deletes only those files for which the EXPIRATION-DATE ≥ the specified date.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = <date>(...)
Deletes only those files for which the EXPIRATION-DATE ≥ the specified date.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = *LATEST / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)
Only files for which the retention period expires on or before the specified date (EXPIRATION-DATE ≤ specified date) will be deleted.

TO = *TOMORROW(...)
Deletes only those files for which the EXPIRATION-DATE ≤ date of next day.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = *TODAY(...)
Deletes only those files for which the EXPIRATION-DATE ≤ current date.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = *YESTERDAY(...)
Deletes only those files for which the EXPIRATION-DATE ≤ date of preceding day.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = <integer -99999..991231>(...)
Deletes only those files for which the EXPIRATION-DATE ≤ the specified date.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = <date>(...)
Deletes only those files for which the EXPIRATION-DATE ≤ the specified date.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

LAST-ACCESS-DATE = *ANY / *NONE / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)
The user can select the files to be deleted by the date when they were last accessed. On a given date, the selection can be restricted to a time interval within which the file was last accessed (see the relevant TIME operands in each structure). On deleting file generation groups and file generations see the DELETE-FILE-GROUP and DELETE-FILE-GENERATION commands. See also the *ACC-DATE* and *ACC-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

LAST-ACCESS-DATE = *ANY
The date of last access is not to be used as a selection criterion.

LAST-ACCESS-DATE = *NONE
Deletes only those files which have the value NONE entered in the *ACC-DATE* field in their catalog entry, i.e. files which have never been opened.

LAST-ACCESS-DATE = *TODAY(...)
Deletes only those files for which today’s date has been entered as the LAST-ACCESS-DATE in the catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the selection to a period of time relative to today’s date.

TIME = *INTERVAL(...)
Returns information on files that were last accessed within the specified time interval which follows.

FROM = <u>00:00:00</u> / <time>
Selects files that were last accessed after the specified time.

TO = <u>23:59:59</u> / <time>
Selects files that were last accessed before the specified time.

LAST-ACCESS-DATE = *YESTERDAY(...)
Deletes only those files for which yesterday's date has been entered as the LAST-ACCESS-DATE in the catalog entry.

TIME = INTERVAL(...)
Returns information on files that were last accessed within the specified time interval which follows.

FROM = <u>00:00:00</u> / <time>
Selects files that were last accessed after the specified time.

TO = <u>23:59:59</u> / <time>
Selects files that were last accessed before the specified time.

LAST-ACCESS-DATE = <integer -99999..991231>(...)
Deletes only those files for which the specified date has been entered as the LAST-ACCESS-DATE in the catalog entry. Here, the user can specify the date of last access in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value*: (6 digits, with preceding sign) the number of days from today's date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ ±0)

TIME = *INTERVAL(...)
Returns information on files that were last accessed within the specified time interval which follows.

FROM = <u>00:00:00</u> / <time>
Selects files that were last accessed after the specified time.

TO = <u>23:59:59</u> / <time>
Selects files that were last accessed before the specified time.

LAST-ACCESS-DATE = <date>(...)
Deletes only those files for which the specified date has been entered as the LAST-ACCESS-DATE in the catalog entry. The user can specify the creation date in the form [yy]yy-mm-dd.

TIME = *INTERVAL(...)
Returns information on files that were last accessed within the specified time interval which follows.

FROM = 00:00:00 / <time>
Selects files that were last accessed after the specified time.

TO = 23:59:59 / <time>
Selects files that were last accessed before the specified time.

LAST-ACCESS-DATE = *INTERVAL(...)
Only files which were last accessed within the specified time period will be deleted. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified using <integer> values). It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for deletion can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = *EARLIEST / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)
Only files which have been accessed since the specified date (i.e. for which LAST-ACCESS-DATE ≥ specified date) will be deleted. If EARLIEST is specified, the system will use the earliest possible date.

FROM = *TODAY(...)
Returns information on files for which the LAST-ACCESS-DATE ≥ date of the current day.

TIME = 00:00:00 / <time>
Time on the specified date. All files which were last accessed after the specified time are selected.

FROM = *YESTERDAY(...)
Returns information on files for which the LAST-ACCESS-DATE ≥ date of the preceding day.

TIME = 00:00:00 / <time>
Time on the specified date. All files which were last accessed after the specified time are selected.

FROM = <integer -99999..991231>(...)
Returns information on files for which the LAST-ACCESS-DATE ≥ the specified date.

TIME = 00:00:00 / <time>
Time on the specified date. All files which were last accessed after the specified time are selected.

FROM = <date>(...)
Returns information on files for which the LAST-ACCESS-DATE ≥ the specified date.

TIME = 00:00:00 / <time>
Time on the specified date. All files which were last accessed after the specified time are selected.

TO = *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)
Only those files which were last accessed on or before the specified date (i.e. files for which LAST-ACCESS-DATE ≤ the specified date) are deleted.

TO = *TODAY(...)
Returns information on files for which the LAST-ACCESS-DATE ≤ date of the current day.

TIME = 23:59:59 / <time>
Time on the specified date. All files which were last accessed before the specified time are selected.

TO = *YESTERDAY(...)
Returns information on files for which the LAST-ACCESS-DATE ≤ date of the preceding day.

TIME = 23:59:59 / <time>
Time on the specified date. All files which were last accessed before the specified time are selected.

TO = <integer -99999..991231>(...)
Returns information on files for which the LAST-ACCESS-DATE ≤ the specified date.

TIME = 23:59:59 / <time>
Time on the specified date. All files which were last accessed before the specified time are selected.

TO = <date>(...)
Returns information on files for which the LAST-ACCESS-DATE ≤ the specified date.

TIME = 23:59:59 / <time>
Time on the specified date. All files which were last accessed before the specified time are selected.

LAST-CHANGE-DATE = *ANY / *NONE / *TODAY(...) / *YESTERDAY(...) / <integer -99999..0>(...) / <date>(...) / *INTERVAL(...)
The user can select the files to be deleted based on the date on which they were last accessed for writing (i.e. the date they were last changed). On a given date, the selection can be restricted to a time interval within which the file was last changed (see the relevant TIME operands in each structure). On deleting file generation groups and file generations see the DELETE-FILE-GROUP and DELETE-FILE-GENERATION commands. See also the *CHANG-DATE* and *CHANG-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

LAST-CHANGE-DATE = *ANY
The date of the last write access is not to be used as a selection criterion.

LAST-CHANGE-DATE = *NONE
Deletes only those files for which no LAST-CHANGE-DATE has been entered in the catalog entry, i.e. files which have never been opened.

LAST-CHANGE-DATE = *TODAY(...)
Deletes only those files for which today’s date has been entered as the LAST-CHANGE-DATE in the catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the selection to a period of time relative to today’s date.

TIME = *INTERVAL(...)
Deletes only those files which were last changed within the specified time interval.

FROM = 00:00:00 / <time>
Selects files for which the LAST-CHANGE-DATE ≥ the specified time.

TO = 23:59:59 / <time>
Selects files for which the LAST-CHANGE-DATE ≤ the specified time.

LAST-CHANGE-DATE = *YESTERDAY(...)
Deletes only those files for which yesterday’s date has been entered as the LAST-CHANGE-DATE in the catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified LAST-CHANGE-DATE.

TIME = *INTERVAL(...)
Deletes only those files which were last changed within the specified time interval.

FROM = 00:00:00 / <time>
Selects files for which the LAST-CHANGE-DATE ≥ the specified time.

TO = 23:59:59 / <time>
Selects files for which the LAST-CHANGE-DATE ≤ the specified time.

LAST-CHANGE-DATE = <integer -99999..0>(...)
Deletes only those files for which the specified date has been entered as the LAST-CHANGE-DATE in the catalog entry. Here, the user can specify the date of the last write access relative to the current date (in the form -n).

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified LAST-CHANGE-DATE.

TIME = *INTERVAL(...)
Deletes only those files which were last changed within the specified time interval.

FROM = 00:00:00 / <time>
Selects files for which the LAST-CHANGE-DATE ≥ the specified time.

TO = 23:59:59 / <time>
Selects files for which the LAST-CHANGE-DATE ≤ the specified time.

LAST-CHANGE-DATE = <date>(...)
Deletes only those files for which the specified date has been entered as the LAST-CHANGE-DATE in the catalog entry. The user can specify the date in the form [yy]yy-mm-dd.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified LAST-CHANGE-DATE.

TIME = *INTERVAL(...)
Deletes only those files which were last changed within the specified time interval.

FROM = 00:00:00 / <time>
Selects files for which the LAST-CHANGE-DATE ≥ the specified time.

TO = 23:59:59 / <time>
Selects files for which the LAST-CHANGE-DATE ≤ the specified time.

LAST-CHANGE-DATE = *INTERVAL(...)
Only files which were last changed within the specified time period will be deleted. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified for the operand LAST-CHANGE-DATE= <integer...>. It is also possible to specify only the FROM operand (lower limit) or the TO operand (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for information output can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = *EARLIEST / *TODAY(...) / *YESTERDAY(...) / <integer -99999..0>(...) / <date>(...)
Only those files which have been changed since the specified date (i.e. LAST-CHANGE-DATE ≥ specified date) are deleted.

FROM = *TODAY(...)
Deletes only those files for which the LAST-CHANGE-DATE ≥ current date.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have a LAST-CHANGE-DATE after the specified time are selected.

FROM = *YESTERDAY(...)
Deletes only those files for which the LAST-CHANGE-DATE ≥ date of preceding day.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have a LAST-CHANGE-DATE after the specified time are selected.

FROM = <integer -99999..0>(...)
Deletes only those files for which the LAST-CHANGE-DATE ≥ the specified date.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have a LAST-CHANGE-DATE after the specified time are selected.

FROM = <date>(...)
Deletes only those files for which the LAST-CHANGE-DATE ≥ the specified date.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have a LAST-CHANGE-DATE after the specified time are selected.

TO = *TODAY(...) / *YESTERDAY(...) / <integer -99999..0>(...) / <date>(...)
Only those files which were changed on or before the specified date (i.e. LAST-CHANGE-DATE ≤ specified date) are deleted.

TO = *TODAY(...)
Deletes only those files for which the LAST-CHANGE-DATE ≤ current date.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have a LAST-CHANGE-DATE before the specified time are selected.

TO = *YESTERDAY(...)
Deletes only those files for which the LAST-CHANGE-DATE ≤ date of preceding day.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have a LAST-CHANGE-DATE before the specified time are selected.

TO = <integer -99999..0>(...)
Returns information on files for which the LAST-CHANGE-DATE ≤ the specified date.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have a LAST-CHANGE-DATE before the specified time are selected.

TO = <date>(...)
Deletes only those files for which the LAST-CHANGE-DATE ≤ the specified date.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have a LAST-CHANGE-DATE before the specified time are selected.

SUPPORT = *ANY / list-poss(3): *PUBLIC-DISK / *PRIVATE-DISK / *TAPE
The user can select which files are to be deleted by the DELETE-FILE command by the type of their volumes. On deleting file generation groups and file generations see the DELETE-FILE-GROUP and DELETE-FILE-GENERATION commands.

SUPPORT = *ANY
The volume type is not to be used as a selection criterion.

SUPPORT = *PUBLIC-DISK
Only files on disks of the pubset or on Net-Storage volumes are deleted.

SUPPORT = *PRIVATE-DISK
Only files on private disks are to be deleted.

SUPPORT = *TAPE
Only files on tape or tape cartridge are to be deleted.

STORAGE-TYPE = *ANY / *PUBLIC-SPACE / *NET-STORAGE
The selection criterion is the assigned storage type.

STORAGE-TYPE = *ANY
The storage type is not a selection criterion.

STORAGE-TYPE = *PUBLIC-SPACE
Only files residing on disks of the pubset are deleted.

STORAGE-TYPE = *NET-STORAGE(...)
Only files residing on Net-Storage volumes are deleted.

FILE-TYPE = *ANY / *BS2000 / *NODE-FILE
The selection criterion is the file type.

FILE-TYPE = *ANY
The file type is not a selection criterion.

FILE-TYPE = *BS2000
Only BS2000 files on Net-Storage volumes are deleted.

FILE-TYPE = *NODE-FILE
Only node files on Net-Storage volumes are deleted.

VOLUME = *ANY / <vsn 1..6>
For files on private volumes: by specifying a volume serial number (VSN) the user can limit the deletion of files by the DELETE-FILE command to those files only which are stored on this volume.

VOLUME = *ANY
The volume is not to be used as a selection criterion.

VOLUME = <vsn 1..6>
All files which are stored on the specified volume, or which have dummy entries on the volume, are to be deleted. In this case, no file name must be specified under the FILE-NAME operand.

SIZE = *ANY / *FREESIZE / <integer 0..2147483647> / *INTERVAL(...)

Only for files on disks and Net-Storage volumes: the user can select which files are to be deleted by the DELETE-FILE command according to their size, or the size of their reserved storage area (= number of PAM pages). On deleting file generation groups and file generations see the DELETE-FILE-GROUP and DELETE-FILE-GENERATION commands.
The value of the SIZE operand specifies the number of PAM pages: the upper and lower limits are both included in the range specified.

SIZE = *ANY

The file size is not to be used as a selection criterion.

SIZE = *FREESIZE

Only files for which storage space has been reserved but which do not yet occupy any of it (*HIGH-US-PA=0)* are to be deleted.

SIZE = <integer 0..2147483647>

Only files for which the number of PAM pages reserved is exactly equal to the number specified here are to be deleted.

SIZE = *INTERVAL(...)

Only files for which the number of pages lies within the specified range are to be deleted. The upper and lower limits are both included in the range specified (see description of SIZE=<integer...>). It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for deletion can only be meaningful if the chosen lower limit ≤ the upper limit. Only files for which the number of PAM pages reserved is at least equal to the FROM value and at most equal to the TO value will be processed (FROM ≤ SIZE ≤ TO).

FROM = 0 / <integer 0..2147483647>

Only files for which the number of PAM pages reserved is at least equal to the specified number are to be deleted (SIZE ≥ specified value).

TO = 2147483647 / <integer 0..2147483647>

Only files for which the number of PAM pages reserved is at most equal to the specified number are to be deleted (SIZE ≤ specified value).

NUMBER-OF-EXTENTS = *ANY / <integer 0..65535> / *INTERVAL(...)
Only for files on disks and Net-Storage volumes: the user can select the files to be processed by DELETE-FILE by the number of extents into which a file is “split”. An extent is a contiguous area on a disk, occupied by one file; the output field *EXTENTS* shows how many extents the file has. On deleting file generation groups and file generations see the DELETE-FILE-GROUP and DELETE-FILE-GENERATION commands.

NUMBER-OF-EXTENTS = *ANY
The number of extents is not to be used as a selection criterion.

NUMBER-OF-EXTENTS = <integer 0..65535>
Only disk files with exactly the specified number of extents (*EXTENDS* = integer) are to be deleted.

NUMBER-OF-EXTENTS = *INTERVAL(...)
Selects all the files whose number of extents lies within the specified range. The upper and lower limits are both included in the range specified. It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for deletion can only be meaningful if the chosen lower limit ≤ the upper limit. Only disk files which have at least as many EXTENTS as specified by the FROM value and at most as many as given by the TO value will be processed (FROM ≤ *EXTENTS* ≤ TO).

FROM = 0 / <integer 0..65535>
Only files for which the number of extents is at least equal to the specified number are to be deleted (*EXTENTS* ≥ integer).

TO = 65535 / <integer 0..65535>
Only files for which the number of extents is at most equal to the specified number are to be exported (*EXTENTS* ≤ integer).

NUMBER-OF-FREE-PAGES = *ANY / *SIZE / <integer 0..2147483647> / INTERVAL(...)
Only for files on disks and Net-Storage volumes: the user can select the files to be processed according to their number of free PAM pages, i.e. the number of pages reserved for the file but unused.

NUMBER-OF-FREE-PAGES = *ANY
The number of free PAM pages is not to be used as a selection criterion.

NUMBER-OF-FREE-PAGES = *SIZE
Only files which actually occupy no storage space are to be deleted (i.e. no PAM page has been written yet).

NUMBER-OF-FREE-PAGES = <integer 0..2147483647>
Only files which have exactly the number of reserved but unused (=free) PAM pages specified by <integer...> are to be deleted.

NUMBER-OF-FREE-PAGES = *INTERVAL(...)
Selects all files for which the number of unused PAM pages lies within the specified range. The upper and lower limits are both included in the range specified (see description of NUMBER-OF-FREE-PAGES = <integer...>). It is also possible to specify limits using the operands FROM (lower limit) and TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for deletion can only be meaningful if the chosen lower limit ≤ the upper limit. Only disk files which have at least as many free PAM pages as specified by the FROM value and at most as many as given by the TO value will be deleted (FROM ≤ FREE ≤ TO).

FROM = 0 / <integer 0..2147483647>
Only files for which the number free PAM pages is at least equal to the specified number are to be deleted (FREE ≥ integer).

TO = 2147483647 / <integer 0..2147483647>
Only files for which the number free PAM pages is at most equal to the specified number are to be deleted (FREE ≤ integer).

HIGHEST-USED-PAGE = *ANY / <integer 0..2147483647> / *INTERVAL(...)
The number of PAM pages used serves as a selection criterion here, i.e. the last page pointer points to the specified page (see also the *HIGH-US-PA* output field of the SHOW-FILE-ATTRIBUTES command).

HIGHEST-USED-PAGE = *ANY
The number of PAM pages used is not used as a selection criterion.

HIGHEST-USED-PAGE = <integer 0..2147483647>
Only the files for which the specified number of PAM pages are used (i.e. files for which the last page pointer points to the specified page) are deleted.

HIGHEST-USED-PAGE = *INTERVAL(...)
Deletes only those files which use a number of pages that falls within the specified range.

FROM = 0 / <integer 0..2147483647>
Only the files in which the number of used pages ≥ the specified number are deleted.

TO = 2147483647 / <integer 0..2147483647>
Only the files in which the number of used pages ≤ the specified number are deleted.

BLOCK-COUNTER = *ANY / <integer 0..2147483647> / *INTERVAL(...)
The number of blocks occupied by a tape file serves as the selection criterion (see also the *BLK-COUNT* output field of the SHOW-FILE-ATTRIBUTES command).

BLOCK-COUNTER = *ANY
The number of occupied tape blocks is not to be used as a selection criterion.

BLOCK-COUNTER = <integer 0..2147483647>
Deletes only those tape files which occupy exactly the specified number of tape blocks.

BLOCK-COUNTER = *INTERVAL(...)
Deletes only those tape files for which the occupied number of tape blocks falls within the specified range.

FROM = 0 / <integer 0..2147483647>
Only the tape files for which the number of occupied tape blocks ≥ the specified number are deleted.

TO = 2147483647 / <integer 0..2147483647>
Only the tape files for which the number of occupied tape blocks ≤ the specified number are deleted.

ACCESS = *ANY / *READ / *WRITE
The specified access type will be used as a selection criterion (ACCESS value in the catalog entry.)

ACCESS = *ANY
The ACCESS value is not to be used as a selection criterion.

ACCESS = *READ
Only files for which write access is prevented by ACCESS=READ, i.e. for which only read access is permitted, will be deleted.

ACCESS = *WRITE
Only files for which write and read access are permitted will be deleted.

PASSWORD = *ANY / list-poss(4): *NONE / *READ-PASSWORD / *WRITE-PASSWORD / *EXEC-PASSWORD
The user can select files with specific password types for processing by the DELETE-FILE command.

PASSWORD = *ANY
Password protection is not to be used as a selection criterion.

PASSWORD = *NONE
Only files for which there is no password protection are to be deleted.

PASSWORD = *READ-PASSWORD
Only files which are protected by a read password are to be deleted.

PASSWORD = *WRITE-PASSWORD
Only files which are protected by a write password are to be deleted.

PASSWORD = *EXEC-PASSWORD
Only files which are protected by an execute password are to be deleted.

USER-ACCESS = <u>*ANY</u> / list-poss(3): *OWNER-ONLY / *ALL-USERS / *SPECIAL
The user can select files using their access authorizations as a criterion.

USER-ACCESS = <u>*ANY</u>
The access authorization is not to be used as a selection criterion.

USER-ACCESS = *OWNER-ONLY
Only files which the owner alone may access are to be deleted.

USER-ACCESS = *ALL-USERS
Only files which permit access to all users are to be deleted.

USER-ACCESS = *SPECIAL
Only files which can be accessed by all user IDs including the maintenance IDs (i.e. user IDs with HARDWARE-MAINTENANCE privileges) are to be deleted.

FILE-STRUCTURE = <u>*ANY</u> / list-poss(5): *PAM / *SAM / *ISAM / *BTAM / *NONE
The user can select files for processing according to their access method (*FILE-STRUC* output field).

FILE-STRUCTURE = <u>*ANY</u>
The access method is not to be used as a selection criterion.

FILE-STRUCTURE = *PAM
Only PAM files should be processed.

FILE-STRUCTURE = *SAM
Only SAM files are to be processed.

FILE-STRUCTURE = *ISAM
Only ISAM files are to be processed.

FILE-STRUCTURE = *BTAM
Only BTAM files are to be processed. BTAM files are tape files.

FILE-STRUCTURE = *NONE
Only files for which FILE-STRUCTURE=NONE, i.e. files which have not yet been opened, are to be deleted.

BACKUP-CLASS = *ANY / list-poss(5): *A / *B / *C / *D / *E
The user can select files for deletion by their BACKUP-CLASS level.

BACKUP-CLASS = *ANY
The BACKUP-CLASS level is not to be used as a selection criterion.

BACKUP-CLASS = *A
Only the files for which the value A is entered in the catalog as the BACKUP-CLASS (most frequent backup) are deleted.

BACKUP-CLASS = *B
Only the files for which the value B is entered in the catalog as the BACKUP-CLASS are deleted.

BACKUP-CLASS = *C
Only the files for which the value C is entered in the catalog as the BACKUP-CLASS are deleted.

BACKUP-CLASS = *D
Only the files for which the value D is entered in the catalog as the BACKUP-CLASS are deleted.

BACKUP-CLASS = *E
Only the files for which the value E is entered in the catalog as the BACKUP-CLASS are deleted.

BLOCK-CONTROL-INFO = *ANY / list-poss(9): *NONE / *NO / *WITHIN-DATA-BLOCK / *WITHIN-DATA-2K-BLOCK / *WITHIN-DATA-4K-BLOCK / *PAMKEY / *NK / *NK2 / *NK4
The user can select files for deletion by their block format (*BLK-CONTR* output field).

BLOCK-CONTROL-INFO = *ANY
The BLOCK-CONTROL entry is not to be used as a selection criterion.

BLOCK-CONTROL-INFO = *NONE
Only files for which no BLK-CNTRL value has been defined, i.e. files which have not yet been opened, are to be deleted.

BLOCK-CONTROL-INFO = *NO
Only files which contain no block control field are to be deleted.

BLOCK-CONTROL-INFO = *WITHIN-DATA-BLOCK
Only files which were created with BLOCK-CONTROL-INFO=WITHIN-DATA-BLOCK (i.e. files for which block control information is held in a block control field at the start and within the data block) are to be deleted.

BLOCK-CONTROL-INFO = *WITHIN-DATA-2K-BLOCK
Only files which were created with BLOCK-CONTROL-INFO=WITHIN-DATA-2K-BLOCK (i.e. files for which block control information is located at the start of each 2K block) are to be deleted.

BLOCK-CONTROL-INFO = *WITHIN-DATA-4K-BLOCK
Only files which were created with BLOCK-CONTROL-INFO=WITHIN-DATA-4K-BLOCK (i.e. files for which block control information is located at the start of each 4K block) are to be deleted.

BLOCK-CONTROL-INFO = *PAMKEY
Only files which use a separate PAM key for the block control field (i.e the block control information is held in a separate key field, outside the PAM block) are to be deleted.

BLOCK-CONTROL-INFO = *NK
Only the NK files are to be deleted, i.e. files which can also be stored on NK volumes (NK2 and NK4).

BLOCK-CONTROL-INFO = *NK2
Only files which can also be stored on NK2 volumes (but not NK4 volumes) are to be deleted.

BLOCK-CONTROL-INFO = *NK4
Only files which can also be stored on NK4 volumes are to be deleted.

MIGRATE = <u>*ANY</u> / list-poss(3): *ALLOWED / *INHIBITED / *FORBIDDEN
The user can specify which files are to be processed by the DELETE-FILE command by the migration entry in the catalog (see the CREATE-FILE command, MIGRATE operand).

MIGRATE = <u>*ANY</u>
The specified files are to be deleted, irrespective of the value in the MIGRATE operand in each of their catalog entries.

MIGRATE = *ALLOWED
Only files for which the catalog entry specifies the appropriate operand value, i.e. files which may be migrated to storage levels S1 and S2, are to be deleted.

MIGRATE = *INHIBITED
Only files for which the catalog entry specifies the appropriate operand value are to be processed, i.e. files for which a simple migration lock is declared.

MIGRATE = *FORBIDDEN
Only files for which the catalog entry specifies the appropriate operand value are to be processed, i.e. files for which an intensified migration lock is declared. The files may not even be migrated for a brief period (e.g. for reorganization purposes).

STORAGE-LEVEL = *<u>ANY</u> / list-poss(3): *S0 / *S1 / *S2
Only files at a specific HSMS storage level are to be deleted. Where HSMS (**H**ierarchical **S**torage **M**anagement **S**ystem) is used, there are three possible storage levels for files on shared storage media:

– The processing level, S0. This comprises all pubsets to which the user has access according to his user entry.

– The background level, S1. This comprises disks on which HSMS has stored files which have been migrated from the S0 level. When DMS requires access to them, they are fetched back to processing level S0 by HSMS.

– The archival level, S2. This is a tape archive in which HSMS stores files which have been migrated from S0 on a longer term basis. When DMS requires access to them, they are fetched back to the processing level S0, depending on the HSMS operating parameters which have been set, and the availability of tape units.

The user can select files for deletion according to the level in the storage hierarchy (STORAGE-LEVEL) at which they are held. Where HSMS (Hierarchical Storage Management System) is used, the following three storage levels are possible for files on shared storage media:

S0: implemented as fast access disk storage (for online processing); all the pubsets to which his user entry gives him access.

S1: implemented as high capacity disk storage (background level, available online). These are disks on which HSMS stores files which have been migrated from the S0 level. When DMS requires access to them, they are fetched back to processing level S0 by HSMS.

S2: implemented as a magnetic tape and tape cartridge archive (background level, available offline). Archive level S2 is a tape archive in which HSMS stores files which are to be migrated from S0 on a longer term basis. When DMS requires access to them, they are fetched back to the processing level S0, depending on the HSMS operating parameters which have been set, and the availability of tape units.

STORAGE-LEVEL = *<u>ANY</u>
The specified files are to be deleted, irrespective of the storage level at which they are being held.

STORAGE-LEVEL = *S0
Only files which are being held at level S0 are to be deleted.

STORAGE-LEVEL = *S1
Only files which are being held at level S1 are to be deleted.

STORAGE-LEVEL = *S2
Only files which are being held at level S2 are to be deleted.

BASIC-ACL = *<u>ANY</u> / *NONE / *YES / *PARAMETERS(...)
The BASIC-ACL entry in the file catalog is used as a selection criterion.

BASIC-ACL = *<u>ANY</u>
The BASIC-ACL entry is not to be used as a selection criterion.

BASIC-ACL = *NONE
Only the files that have no BASIC-ACL entry in the catalog are to be deleted.

BASIC-ACL = *YES
Only the files which have a BASIC-ACL entry in the catalog are to be deleted.

BASIC-ACL = *PARAMETERS(...)
Only the files which have the specified BASIC-ACL entry in the catalog are to be deleted. NO-ACCESS means that no access rights were granted.

Access rights specified with the OWNER, GROUP and OTHERS operands within the *PARAMETERS(...) structure are logically ORed.

OWNER = *ANY / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights must already be defined for the owner.

OWNER = *PARAMETERS(...)
Access rights that must be present for the owner (logically ORed together):

READ = *ANY / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = *ANY / *NO / *YES
Specifies whether write access authorization must be present.

EXEC = *ANY / *NO / *YES
Specifies whether execute access authorization must be present.

GROUP = *ANY / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights should already be defined for the owner’s user group.

GROUP = *PARAMETERS(...)
Access rights that must be present for the owner’s user group (logically ORed together):

READ = *ANY / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = *ANY / *NO / *YES
Specifies whether write access authorization must be present.

EXEC = *ANY / *NO / *YES
Specifies whether execute access authorization must be present.

OTHERS = *ANY / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights should already be defined for all other users.

OTHERS = *PARAMETERS(...)
Access rights that must be present for all other users (logically ORed together):

READ = *ANY / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = *ANY / *NO / *YES
Specifies whether write access authorization must be present.

EXEC = *ANY / *NO / *YES
Specifies whether execute access authorization must be present.

GUARDS = *ANY / *NO / *YES / *PARAMETERS(...)
The use of GUARDS for access control (see the CREATE-FILE or MODIFY-FILE-ATTRIBUTES command) is used as a selection criterion.

GUARDS = *ANY
Access control with GUARDS is not to be used as a selection criterion.

GUARDS = *NO
Only the files which are not protected by GUARDS against unauthorized access are to be deleted (see also the PROTECTION-LEVEL operand).

GUARDS = *YES
Only the files which are protected by GUARDS against unauthorized access (i.e. files for which access is controlled via the GUARDS function unit) are to be deleted.

GUARDS = *PARAMETERS(...)
Deletes only those files which are protected by GUARDS against unauthorized access as specified, i.e. files for which access control is implemented using GUARDS: Access to the file is controlled by a guard, i.e. a special object which contains all the conditions under which a user is granted access authorization (e.g. date, time, user ID). The GUARDS function unit of the chargeable software product SECOS (see the “SECOS” manual [35]) must be installed in order to create and maintain a guard. Each access mode can be controlled by a separate guard. If no guard (*NONE) is defined for a given access mode, no corresponding access is permitted. If a defined guard is not accessible, the mode of access protected by it is not permitted. If the GUARDS subsystem is not available at the time of accessing the job variable, no access of any kind is allowed for the job variable.

i The values specified for the following READ, WRITE and EXEC operands are logically ORed.

READ = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>
Only the files which are protected against unauthorized read access by the specified guard are deleted. The default value *ANY means that the selection of files is not based on read protection with a guard. *NONE selects files for which no read access is permitted.

WRITE = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>
Only the files which are protected against unauthorized write access by the specified guard are deleted. The default value *ANY means that the selection of files is not based on a guard. *NONE selects files for which no write access is permitted. These files can be deleted with IGNORE-PROTECTION=*ACCESS.

EXEC = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>
Only the files which are protected against unauthorized execution by the specified guard are deleted. The default value *ANY means that the selection of files is not based on execute access protection by a guard. *NONE selects files for which no execution is permitted.

PROTECTION-ACTIVE = *ANY / list-poss(3): *LEVEL-0 / *LEVEL-1 / *LEVEL-2
The highest activated access control method (protection level) is used as a selection criterion. Only the files which have access protection at specified protection level are deleted. When the file is accessed, the highest activated protection level applies. The following table shows the method used for access control, the protection attributes, and the job variable protection hierarchy (protection levels):

Access control	Protection attribute	Prot. level
Standard access control	ACCESS and USER-ACCESS	0
Basic access control list	BASIC-ACL	1
Access control via guards	PASSWORD	2

Table 45: Hierarchy of access control methods

All other protection attributes of the file (e.g. passwords) are evaluated independently, without regard to the implemented protection level.

PROTECTION-ACTIVE = *ANY
The access control method is not to be used as a selection criterion.

PROTECTION-ACTIVE = *LEVEL-0
Only the files for which access is controlled via standard access control are to be deleted.

PROTECTION-ACTIVE = *LEVEL-1
Only the files for which access is controlled via a basic access control list (BASIC-ACL protection) are to be deleted.

PROTECTION-ACTIVE = *LEVEL-2
Only the files for which access takes place via GUARDS are deleted.

ACCESS-COUNTER = *ANY / <integer 0..2147483647> / *INTERVAL(...)
The access counter for the file is used as a selection criterion (see the *ACC-COUNT* output field of the SHOW-FILE-ATTRIBUTES command).

ACCESS-COUNTER = *ANY
The access counter is not to be used as a selection criterion.

ACCESS-COUNTER = <integer 0..2147483647>
Only the files for which the access counter exactly matches the specified value are selected for deletion.

ACCESS-COUNTER = *INTERVAL(...)
Deletes only the files for which the access counter lies in the specified value range which follows.

FROM = 0 / <integer 0..2147483647>
Deletes only those files for which the access counter ≥ the specified value.

TO = 2147483647 / <integer 0..2147483647>
Deletes only those files for which the access counter ≤ the specified value.

CODED-CHARACTER-SET = *ANY / *NONE / <name 1..8>
The coded character set defined in the catalog entry is used as a selection criterion (see the CREATE-FILE command; if a CCS is defined, see also the *COD-CH-SET* output field of the SHOW-FILE-ATTRIBUTES command).

CODED-CHARACTER-SET = *ANY
The defined character set is not to be used as a selection criterion.

CODED-CHARACTER-SET = *NONE
Only the files for which no coded character set was explicitly defined are selected for deletion.

CODED-CHARACTER-SET = <name 1..8>
Only the files for which the specified character set was defined are selected for deletion.

STATUS = *ANY / *PARAMETERS(...)
The current file status is used as a selection criterion.

STATUS = *ANY
The file status is not to be used as a selection criterion.

STATUS = *PARAMETERS(...)
Only the files which have the specified status are selected for deletion. The following selection criteria are possible:

i The selection criteria within the *PARAMETERS(...) structure are logically ORed.

CLOSED-OUTPUT = *ANY / *YES / *NO
Specifies whether the “file closed” status is to be used as a selection criterion.

CLOSED-OUTPUT = *YES
Only the files which have already been closed are deleted.

CLOSED-OUTPUT = *NO
Only the output files which have been opened in a program (OPEN OUTIN, INOUT or OUTPUT) and files which were not closed in an earlier system run or because a job was aborted are selected for deletion.

CACHED = *ANY / *YES / *NO
Specifies whether file processing via a cache is to be used as a selection criterion.

CACHED = *ANY
Deletes files regardless of whether or not they are cached.

CACHED = *YES
Only the currently cached files are selected for deletion.

CACHED = *NO
Only the files for which no data has been cached are deleted.

REPAIR-NEEDED = *ANY / *YES / *NO
Specifies whether files which were not closed in an earlier system run and not reconstructed with REPAIR-DISK-FILES are to be selected.

REPAIR-NEEDED = *ANY
Deletes files regardless of whether or not they need to be repaired.

REPAIR-NEEDED = *YES
Only the files which were not closed in an earlier system run and which have not yet been reconstructed are selected for deletion.

REPAIR-NEEDED = *NO
Only the files which were closed normally or reconstructed with the REPAIR-DISK-FILES command are deleted.

OPEN-ALLOWED = *ANY / *YES / *NO
Specifies whether files which could not be opened due to data inconsistency are to be selected.

OPEN-ALLOWED = *ANY
Deletes files regardless of whether or not they can be opened.

OPEN-ALLOWED = *YES
Only the files that can be opened are deleted.

OPEN-ALLOWED = *NO
Only the files which cannot be opened due to data inconsistency are deleted.

DEFECT-REPORTED = *ANY / *YES
Specifies whether files which may contain defective disk blocks are to be used as a selection criterion.

DEFECT-REPORTED = *ANY
Files are deleted irrespective of whether or not they may contain defective blocks.

DEFECT-REPORTED = *YES
Only files which may contain defective blocks are deleted.

CACHE-NOT-SAVED = *ANY / *YES
Specifies whether to select files for which updates have not been written out from the cache.

CACHE-NOT-SAVED = *ANY
Files are deleted irrespective of whether or not they contain modified data which could not be written back from the cache.

CACHE-NOT-SAVED = *YES
Only files containing modified data which could not be written back from the cache are deleted.

TYPE-OF-FILES = *ANY / list-poss(2): *FILE / *PLAM-LIBRARY
Specifies whether files are to be selected on the basis of information specific to the file type. The files to be exported can restricted to normal files or to PLAM libraries.

TYPE-OF-FILES = *ANY
Deletes files without taking their file type into account.

TYPE-OF-FILES = *FILE
Only files are deleted.

TYPE-OF-FILES = *PLAM-LIBRARY
Only PLAM libraries are deleted.

SPACE-RELEASE-LOCK = *ANY / *NO / *YES
The lock to prevent the release of unused memory space is used as a selection criterion (see the *SP-REL-LOCK* output field in the SHOW-FILE-ATTRIBUTES command).

SPACE-RELEASE-LOCK = *ANY
The permission to release unused memory space is not used as a selection criterion.

SPACE-RELEASE-LOCK = *NO
Only the files for which unused memory space may be released are selected for deletion.

SPACE-RELEASE-LOCK = *YES
Only the files for which the release of unused memory space is not permitted are selected for deletion.

IO-ATTRIBUTES = *ANY / *PARAMETERS(...)
The performance attributes defined in the catalog are used as a selection criterion.

IO-ATTRIBUTES = *ANY
The performance attributes are not to be used as a selection criterion.

IO-ATTRIBUTES = *PARAMETERS(...)
Only the files for which the specified performance attributes which follow have been entered in the catalog are selected for deletion.

PERFORMANCE = <u>*ANY</u> / list-poss(3): *STD / *HIGH / *VERY-HIGH
Deletes files with the specified performance attribute (*IO(PERF)* output field).

PERFORMANCE = <u>*ANY</u>
The performance attribute is not a selection criterion.

PERFORMANCE = *STD
Only the files that can be processed without any special performance requirements are deleted.

PERFORMANCE = *HIGH
Only the files which should be processed via a cache (high performance priority) are deleted.

PERFORMANCE = *VERY-HIGH
Only the files which should be processed via a cache and which should be permanently maintained in the cache if possible (highest performance priority) are selected for deletion.

USAGE = <u>*ANY</u> / list-poss(3): *READ-WRITE / *WRITE / *READ
Only the files which have a performance attribute that applies to the specified I/O operations (*IO(USAGE)* output field) are to be deleted.

USAGE = <u>*ANY</u>
The type of I/O operations to which the performance attribute applies is not to be used as a selection criterion.

USAGE = *READ-WRITE
Only the files for which the performance attribute applies to read-write operations are selected for deletion.

USAGE = *WRITE
Deletes files for which the performance attribute applies to only write operations.

USAGE = *READ
Deletes files for which the performance attribute applies to only read operations.

DISK-WRITE = <u>*ANY</u> / *IMMEDIATE / *BY-CLOSE
The time at which data consistency is required after a write operation is used as a selection criterion (see also the *DISK-WRITE* output field in the SHOW-FILE-ATTRIBUTES command.

DISK-WRITE = <u>*ANY</u>
The point at which data consistency is required is not a selection criterion.

DISK-WRITE = *IMMEDIATE
Deletes files which must be in a consistent state immediately after a write operation.

DISK-WRITE = *BY-CLOSE
Deletes files for which consistency of data is required only after CLOSE processing.

FREE-FOR-DELETION = *ANY / *NONE / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) / <integer -99999..0>(...) / <date>(...) / *INTERVAL(...)
The date of release for deletion (free-for-deletion date) can be used as a selection criterion for the files which are to be deleted. The selection for a specified date can be restricted to a specific period of time on the free-for-deletion date (see the TIME operands in the various structures) On deleting file generation groups and file generations see the DELETE-FILE-GROUP and DELETE-FILE-GENERATION commands. See also the SHOW-FILE-ATTRIBUTES command, *FREE-DEL-D* and *FREE-DEL-T* output fields.

FREE-FOR-DELETION = *ANY
The free-for-deletion date is not used as a selection criterion.

FREE-FOR-DELETION = *NONE
Only deletes files for which no free-for-deletion date has yet been entered in the catalog.

FREE-FOR-DELETION = *TOMORROW(...)
Only deletes files for which the next day is entered as the free-for-deletion date in the catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the file selection to a period relative to the date of the next day.

TIME = *INTERVAL(...)
Only deletes files with a free-for-deletion date which falls within the period specified next.

FROM = 00:00:00 / <time>
Selects files with a free-for-deletion date ≥ the specified time.

TO = 23:59:59 / <time>
Selects files with a free-for-deletion date ≤ the specified time.

FREE-FOR-DELETION = *TODAY(...)
Only deletes files with today’s date recorded as free-for-deletion date in their catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the selection to a period of time relative to today’s date.

TIME = *INTERVAL(...)
Only deletes files with a free-for-deletion date which falls within the period specified next.

FROM = 00:00:00 / <time>
Selects files with a free-for-deletion date ≥ the specified time.

TO = 23:59:59 / <time>
Selects files with a free-for-deletion date ≤ the specified time.

FREE-FOR-DELETION = *YESTERDAY(...)
Only deletes files with yesterday's date recorded as free-for-deletion date in their catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the selection to a period of time relative to yesterday's date.

TIME = *INTERVAL(...)
Only deletes files with a free-for-deletion date which was changed within the period specified next.

FROM = 00:00:00 / <time>
Selects files with a free-for-deletion date ≥ the specified time.

TO = 23:59:59 / <time>
Selects files with a free-for-deletion date ≤ the specified time.

FREE-FOR-DELETION = <integer -99999..99999>(...)
Only deletes files with the specified free-for-deletion date recorded in their catalog entry. The free-for-deletion date is specified relative to today's date (in the form -n for the past or +n for the future).

TIME = *ANY / *INTERVAL(...)
Restricts the selection to a period of time relative to the specified date.

TIME = *INTERVAL(...)
Only deletes files with a free-for-deletion date which falls within the period specified next.

FROM = 00:00:00 / <time>
Selects files with a free-for-deletion date ≥ the specified time.

TO = 23:59:59 / <time>
Selects files with a free-for-deletion date ≤ the specified time.

FREE-FOR-DELETION = <date>(...)
Only deletes files with exactly the specified date recorded as free-for-deletion date in their catalog entry. The date is specified in the form [yy]yy-mm-dd. 20 is automatically prefixed to two-digit year specifications < 60, 19 to two-digit year specifications ≥ 60.

TIME = *ANY / *INTERVAL(...)
Restricts the selection to a period of time relative to the specified date.

TIME = *INTERVAL(...)
Only deletes files with a free-for-deletion date which falls within the period specified next.

FROM = 00:00:00 / <time>
Selects files with a free-for-deletion date ≥ the specified time.

TO = 23:59:59 / <time>
Selects files with a free-for-deletion date ≤ the specified time.

FREE-FOR-DELETION = *INTERVAL(...)
Only deletes files with a free-for-deletion date which falls within the specified period. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified for the FREE-FOR-DELETION= <integer...> operand. It is also possible to specify only the FROM operand (lower limit) or the TO operand (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for information output can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = *EARLIEST / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) / <integer -99999..99999>(...) / <date>(...)
Only deletes files with a free-for-deletion date ≥ the specified date.

FROM = *TOMORROW(...)
Only deletes files whose free-for-deletion date ≥ date of the next next day.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have a free-for-deletion date after the specified time are selected.

FROM = *TODAY(...)
Only deletes files with a free-for-deletion date ≥ today’s date.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have a free-for-deletion date after the specified time are selected.

FROM = *YESTERDAY(...)
Only deletes files with a free-for-deletion date ≥ yesterday’s date.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have a free-for-deletion date after the specified time are selected.

FROM = <integer -99999..99999>(...)
Deletes only those files whose free-for-deletion date ≥ the specified date (in the form -n for the past or +n for the future).

TIME = 00:00:00 / <time>
Time on the specified date. All files that have a free-for-deletion date after the specified time are selected.

FROM = <date>(...)
Only deletes files with a free-for-deletion date ≥ the specified date. The date is specified in the form [yy]yy-mm-dd. 20 is automatically prefixed to two-digit year specifications < 60, 19 to two-digit year specifications ≥ 60.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have a free-for-deletion date after the specified time are selected.

TO = *LATEST / *TODAY(...) / *TOMORROW(...) / *YESTERDAY(...) / <integer -99999..99999>(...) / <date>(...)
Only deletes files with a free-for-deletion date ≤ the specified date.

TO = *LATEST
There is no upper limit on the selection range based on the free-for-deletion date.

TO = *TODAY(...)
Only deletes file generations with a free-for-deletion date ≤ today's date.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have a free-for-deletion date before the specified time are selected.

TO = *TOMORROW(...)
Only deletes files whose free-for-deletion date ≤ date of the next day.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have a free-for-deletion date before the specified time are selected.

TO = *YESTERDAY(...)
Only deletes files with a free-for-deletion date ≤ yesterday's date.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have a free-for-deletion date before the specified time are selected.

TO = <integer -99999..99999>(...)
Deletes only those files whose free-for-deletion date ≤ the specified date (in the form -n for the past or +n for the future).

TIME = 23:59:59 / <time>
Time on the specified date. All files that have a free-for-deletion date before the specified time are selected.

TO = <date>(...)
Only deletes files with a free-for-deletion date ≤ the specified date. The date is specified in the form [yy]yy-mm-dd. 20 is automatically prefixed to two-digit year specifications < 60, 19 to two-digit year specifications ≥ 60.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have a free-for-deletion date before the specified time are selected.

STORAGE-CLASS = *ANY / *NONE / <composed-name 1..8>
The selection criterion is the assigned storage class.

STORAGE-CLASS = *ANY
The storage class is not a selection criterion.

STORAGE-CLASS = *NONE
Only deletes files with no storage class assigned to them.

STORAGE-CLASS = <composed-name 1..8>
Only deletes files with the specified storage class assigned to them.

MANAGEMENT-CLASS = *ANY / *NONE / <composed-name 1..8>
The selection criterion is the HSMS management class.

MANAGEMENT-CLASS = *ANY
The HSMS management class is not a selection criterion.

MANAGEMENT-CLASS = *NONE
Only deletes files with no HSMS management class assigned to them.

MANAGEMENT-CLASS = <composed-name 1..8>
Only deletes files with the specified HSMS management class assigned to them.

ADM-INFORMATION = *ANY / *NONE / <c-string 1..8 with-low>
The selection criterion is the information that systems support has added to the catalog.

ADM-INFORMATION = *ANY
The systems support information is not a selection criterion.

ADM-INFORMATION = *NONE
Only deletes files with no systems support information in their catalog.

ADM-INFORMATION = <c-string 1..8 with-low>
Only deletes files with the specified string as the systems support information in their catalog.

USER-INFORMATION = *ANY / *NONE / <c-string 1..8 with-low>
The selection criterion is the information that the user has added to the catalog.

USER-INFORMATION = *ANY
The user information is not a selection criterion.

USER-INFORMATION = *NONE
Only deletes files with no user information in their catalog.

USER-INFORMATION = <c-string 1..8 with-low>
Only deletes files with the specified string as the user information in their catalog.

VOLUME-SET = *ANY / <cat-id 1..4>
The selection criterion is the volume set in which the file is stored.

VOLUME-SET = <u>*ANY</u>
The volume set is not a selection criterion.

VOLUME-SET = <cat-id 1..4>
Only deletes files stored in the specified volume set.

AVAILABILITY = <u>*ANY</u> / list-poss(2): *STD / *HIGH
The selection criterion is the required availability.

AVAILABILITY = <u>*ANY</u>
Availability is not a selection criterion.

AVAILABILITY = *STD
Only deletes files with no high availability requirements.

AVAILABILITY = *HIGH
Only deletes files with high availability requirements. The files are typically on disks mirrored by DRV.

S0-MIGRATION = <u>*ANY</u> / list-poss(2): *ALLOWED / *FORBIDDEN
The selection criterion is the permissibility of migration within the processing level (S0).

S0-MIGRATION = <u>*ANY</u>
Migration within the processing level is not a selection criterion.

S0-MIGRATION = *ALLOWED
Only deletes files for which migration to other volume sets within the processing level is allowed.

S0-MIGRATION = *FORBIDDEN
Only deletes files for which migration to other volume sets within the processing level is forbidden.

WORK-FILE = <u>*ANY</u> / *NO / *YES
The selection criterion is the work file attribute.

WORK-FILE = <u>*ANY</u>
The work file attribute is not a selection criterion.

WORK-FILE = *NO
Only deletes files which are not marked as work files.

WORK-FILE = *YES
Only deletes files which are marked as work files.

FILE-PREFORMAT = <u>*ANY</u> / list-poss(4): *NONE / *K / *NK2 / *NK4
The selection criterion is the preferred file format.

FILE-PREFORMAT = <u>*ANY</u>
The preferred file format is not a selection criterion.

FILE-PREFORMAT = *NONE
Only deletes files for which no preferred file format has been indicated.

FILE-PREFORMAT = *K
Only deletes files which are intended to be K files.

FILE-PREFORMAT = *NK2
Only deletes files which are intended to be NK files in 2K format.

FILE-PREFORMAT = *NK4
Only deletes files which are intended to be NK files in 4K format.

ENCRYPTION = <u>*ANY</u> / list-poss(3): *NONE / *AES / *DES
The selection criterion is the encryption method.

ENCRYPTION = <u>*ANY</u>
The encryption method is not a selection criterion.

ENCRYPTION = *NONE
Only unencrypted files are deleted.

ENCRYPTION = *AES
Only files which were encrypted using the AES method are deleted.

ENCRYPTION = *DES
Only files which were encrypted using the DES method are deleted.

OPTION = <u>*ALL</u> / *SPACE / *DATA / *DATA-KEEP-ATTRIBUTES / *DESTROY-ALL
Controls the deletion of the file.
If *DESTROY-BY-DELETE=YES* was defined in the catalog entry of the file, the file or released storage space will *definitely* be overwritten with binary zeros. If this is not the case and if OPTION=DESTROY-ALL was also not specified, the file or released storage space will be deleted only logically.

OPTION = <u>*ALL</u>
All catalog entries for the selected files are deleted.
The storage space *for files on disks and Net-Storage volumes* is also released

OPTION = *SPACE
Only for files on pubset disks and Net-Storage volumes: the storage space for the files deleted by DELETE-FILE is released, but the catalog entry is retained, although amended: it is then identical with a catalog entry created by a CREATE-FILE SUPPORT=*NONE command.
For *tape files* the default value ALL applies; for *private disks* the operand SPACE will be rejected.

OPTION = *DATA
Only for files on disks and Net-Storage volumes (for tape files the default value ALL applies): the data in the selected files is “logically deleted”. After this, the user can no longer access the data, since he/she is no longer permitted to access the volume concerned physically. The catalog entry and the storage space allocation are retained.

OPTION = *DATA-KEEP-ATTRIBUTES

Only for files on disks and Net-Storage volumes (for tape files the default value ALL applies):
the data of the files involved is “logically deleted” as with OPTION=*DATA, but the data-related file attributes are retained. The data can no longer be addressed by the user.

OPTION = *DESTROY-ALL

Only for files on disks and Net-Storage volumes (for tape files the default value ALL applies):
the storage space for the files involved is released and the catalog entry is deleted; in addition, the storage space thus released is overwritten with binary zeros so that, if the space is allocated again, nobody can read the old data (data security). In the case of files on private disks, all the volumes on which the file was stored must be mounted at the time of deletion.
When a file is deleted, the action parameters are evaluated first. See the command EXPORT-FILE for details of file exportation. “Data destruction” when a file is deleted can also be permanently recorded in the catalog entry using the CREATE-FILE or MODIFY-FILE-ATTRIBUTES command (DESTROY-BY-DELETE=*YES operand); a “DESTROY” flag is then set in the catalog entry (*DESTROY=YES* output field). In this case storage space is released and it will be automatically overwritten.

MOUNT = <u>*FIRST-DISK</u> / *ALL-DISK

Only for files on disks and Net-Storage volume:
specifies whether or not to mount all private disks containing files which are to be deleted. The user can request that only the first of the required private disks be mounted, or all of them. The MOUNT operand should be specified if the options ALL or DESTROY-ALL are specified. For tape files, or files on public disks, any MOUNT specification will be ignored.

MOUNT = <u>*FIRST-DISK</u>

It is only necessary for the first of the private disks, containing the start of the file and its catalog entry, to be online. The associated private disks do not have to be online.

MOUNT = *ALL-DISKS

All the private disks on which parts of the file are held, must be online. If any of the disks is missing, the file will not be deleted; on completion of the DELETE-FILE processing, the spin-off mechanism will be triggered.

DIALOG-CONTROL = <u>*STD</u> / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE / *CATALOG-CHANGE / *USER-ID-CHANGE

Specifies whether and under what conditions a verification dialog is to be conducted with the user during the deletion process. A control dialog is only possible in dialog mode but, in this mode, may also be used in procedures. The only operand value that you can specify in batch mode is *STD or *NO.

The user can intervene with the following inputs:

- Y: the specified file or file set will then be deleted.
- N: the specified file or file set will not be deleted.
- T: processing of the command will be terminated.

- ?: the possible responses will be listed, with an explanation of each.

In addition, the following options can be specified, separated by commas:

- ,CHECK = NO
  The DIALOG-CONTROL mode will be changed to ‘*NO’.
- ,CHECK = PVS
  The DIALOG-CONTROL mode will be changed to ‘*CATALOG-CHANGE’.
- ,CHECK = MULTIPLE
  The DIALOG-CONTROL mode will be changed to ‘*MORE-THAN-ONE-FILE’.
- ,CHECK = SINGLE
  SINGLE The DIALOG-CONTROL mode will be changed to ‘*FILE-CHANGE’.
- ,CHECK = ERROR
  The DIALOG-CONTROL mode will be changed to ‘*ERROR’.
- ,IGNORE = list-poss(5): ACCESS / EXDATE / RDPASS / WRPASS / EXPASS Specifies which protection attributes are to be ignored during deletion. The specification only applies to a single file in the control dialog.
  RDPASS, WRPASS and EXPASS are available to privileged users only.
- ,PASSWORD = list-poss(3): <c-string 1..4> /
  <x-string 1..8> / <integer -2147483648..2147483647>
  Enables password-protected files to be deleted (maximum of 3 passwords). The specification only applies to a single file in the control dialog.

DIALOG-CONTROL = *STD
The default *STD setting is equivalent to *MORE-THAN-ONE-FILE in an interactive dialog (when SYSCMD is connected to the terminal) and to *NO in procedures and in batch mode.

DIALOG-CONTROL = *NO
The user cannot intervene in DELETE-FILE processing; all the specified files will be deleted (without a verification dialog).

DIALOG-CONTROL = *ERROR
If deletion of the selected files proceeds without error, they will be deleted immediately, as when *NO is specified (i.e. no verification dialog). However, if a user-correctable error occurs, then a verification dialog takes place as for DIALOG-CONTROL=*FILE-CHANGE. DIALOG-CONTROL=*ERROR applies implicitly if DIALOG-CONTROL=*FILE-CHANGE is set. In the event of an error, the user may acknowledge the error message, abort DELETE-FILE processing or attempt to rectify the error. If he wishes, he can also change the DIALOG-CONTROL mode (see also the possible forms of intervention listed under the first DIALOG-CONTROL operand).

DIALOG-CONTROL = *FILE-CHANGE
For each file which is to be deleted, the user has the intervention options described under the first DIALOG-CONTROL operand. For each file which is to be processed, the user can decide interactively whether it should be deleted or not (response: YES/NO). If in the verification dialog he specifies protection attributes under “IGNORE”, or one or more passwords under “PASSWORD”, these will be taken into account for any selected file and,

if satisfied, the file will be deleted without further queries (“YES” must also be specified). The user can also abort DELETE-FILE processing, or change the DIALOG-CONTROL mode.

DIALOG-CONTROL = *MORE-THAN-ONE-FILE
If exactly one file is specified, this will be deleted immediately. If the file is specified in partially qualified form, which means that more than one file is selected, or if the file name contains wildcards, the user can decide, each time the catalog ID changes, whether or not files from the new catalog are to be deleted (see the intervention options described for the first of the DIALOG-CONTROL operands). He must respond to the question issued by the system with “YES” or “NO”. DIALOG-CONTROL = *MORE-THAN-ONE-FILE is useful if wildcards are specified for “catid” in the FILE-NAME operand. In the verification dialog, DELETE-FILE processing can be terminated, or the DIALOG-CONTROL mode can be changed.

DIALOG-CONTROL = *CATALOG-CHANGE
As with DIALOG-CONTROL = *MORE-THAN-ONE-FILE, the DELETE-FILE processing routine branches to a verification dialog if files in different catalogs (PUBSETS) are affected. The user can determine whether the files in the current PUBSET should be deleted (YES/NO), DELETE-FILE processing should be terminated, or the DIALOG-CONTROL mode should be changed.

DIALOG-CONTROL = *USER-ID-CHANGE
Whenever the user ID is changed when deleting the files, a branch is made to guided dialog.

OUTPUT = <u>*STD</u> / *NO / *SYSOUT
The user can specify whether a message (`DMS0800`) with the name of the deleted file is to be output to SYSOUT for each successfully deleted file. The default setting *STD is equivalent to OUTPUT=*NO.

OUTPUT = *NO
No messages are output to SYSOUT for successfully deleted files.

OUTPUT = *SYSOUT
For each file that is successfully deleted, a message with the name of that file is output to SYSOUT.

IGNORE-PROTECTION = <u>*NONE</u> / list-poss(5): *ACCESS / *EXPIRATION-DATE / *WRITE-PASSWORD / *READ-PASSWORD / *EXEC-PASSWORD
The user can specify whether any defined file protection against write access or any defined retention period is to be ignored. Systems support staff can also ignore password protection. The specification IGNORE-PROTECTION in the DELETE-FILE command thus makes it unnecessary to issue MODIFY-FILE-ATTRIBUTES commands to reset the protection attributes before the files can be deleted.

IGNORE-PROTECTION = *NONE
The protection attributes “read-only” (ACCESS=READ or no write access with BASIC-ACL or GUARDS) and “retention period” (EXPIRATION-DATE) will be observed during deletion.

IGNORE-PROTECTION = *ACCESS
Files for which write access by the owner is not allowed at the highest activated protection level can still be deleted using DELETE-FILE (see the selection criteria PROTECTION-ACTIVE, ACCESS, BASIC-ACL orGUARDS).

IGNORE-PROTECTION = *EXPIRATION-DATE
Files which are still within their retention period (*EXPIR-DATE* > current date) may be deleted using DELETE-FILE.

IGNORE-PROTECTION = *WRITE-PASSWORD
System support staff are authorized to ignore the protection attribute **write password** when deleting the file.

IGNORE-PROTECTION = *READ-PASSWORD
System support staff are authorized to ignore the protection attribute **read password** when deleting the file.

IGNORE-PROTECTION = *EXEC-PASSWORD
System support staff are authorized to ignore the protection attribute **execute password** when deleting the file.

PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> / <integer -2147483648..2147483647>
The user can specify one or more passwords, which will permit files protected by these passwords to be deleted. The passwords entered here are not recorded in the password table for the job, and are valid only for the current DELETE-FILE processing. Up to 3 passwords may be specified in the form of a list.
In order to delete a password-protected file, the password at the highest access level must be specified (see also the ADD-PASSWORD command).
The operand has the following special characteristics:

- The password entered is not logged.
- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

PASSWORDS-TO-IGNORE = *NONE
No passwords are specified.

SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
In procedures the user can specify whether the spin-off mechanism or SDF-P error handling is to be triggered every time an error occurs (apart from syntax errors), or whether specific error conditions are to be ignored.

SUPPRESS-ERRORS = *NONE
All errors will trigger the spin-off mechanism or SDF-P error handling.

SUPPRESS-ERRORS = list-poss(3): <alphanum-name 7..7>
The user can define which errors are to be ignored by means of their DMS error codes (alphanum-name 7..7). If the specified error occurs, the spin-off mechanism will not be triggered. A maximum of 3 error codes may be specified. DMS error code: 7 characters, of which the first three are always “DMS”; the last 4 characters identify the error; the digits 0...9 and letters A...F are permitted. When error codes are entered, no check is made to verify that valid error codes have been specified.

A detailed list of valid DMS error codes can be found on the manual server (URL: *http://manuals.ts.fujitsu.com*) by means of an HTML application and on the “BS2000 SoftBooks” DVD.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed
1	0	CMD0001	No action required
2	0	DMS05F7	File generation does not exist, but group entry is updated
2	0	DMS06D6	Error on deleting certain files Guaranteed messages: DMS0800, DMS0801, DMS06D5, DMS0666, DMS05C6, DMS05BF, DMS05C3, DMS053F
	1	CMD0202	Syntax or semantic error in command
	32	DMS0584	A state that does not allow the function to continue was reported during processing
	64	CMD0102	Interrupted by K2 key
	64	CMD0216	Privilege errors
	64	DMS0501	Requested catalog not available
	64	DMS0512	Requested catalog not found
	64	DMS051B	Requested user ID not in pubset Guaranteed message: DMS051B
	64	DMS051C	User not authorized to access pubset Guaranteed message: DMS051C
	64	DMS0535	Specified file not shareable
	64	DMS055C	The catalog entry could not be found on the assigned volume
	64	DMS057B	Invalid operand for migrated file
	64	DMS057C	Processing not possible due to HSMS error
	64	DMS057D	File has been migrated and cannot be recalled without delay
	64	DMS057E	File has been migrated, and HSMS is not available
	64	DMS0585	Error detected when processing catalog or multiprocessor system
	64	DMS0586	It is not possible to access or reserve a volume at present
	64	DMS0587	Use of the specified command has been restricted by the system administrator

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	64	DMS05FC	Specified user ID not in HOME pubset
	64	DMS0609	No access to system file
	64	DMS0640	Access to Net-Storage is rejected by the ONETSTOR subsystem because of communication problems with the net client
	64	DMS0643	Net client reports access error
	64	DMS0644	Net client reports internal error
	64	DMS0645	File does not exist on Net-Storage
	64	DMS0649	Net server reports POSIX-ACL error
	64	DMS064A	Net client reports that access to files is forbidden on the Net-Storage volume
	64	DMS064B	Access to node files not supported by the net client
	64	DMS064C	Directory of the specified user ID does not exist on Net-Server
	64	DMS06FF	BCAM connection severed
	130	DMS0524	System address space exhausted
	130	DMS0582	File is currently locked or being used and cannot be processed
	130	DMS0585	Error detected when processing catalog or multiprocessor system
	130	DMS0586	It is not possible to access or reserve a volume at present
	130	DMS0594	Not enough virtual memory available

Notes

- If a file held on a private disk is to be deleted, the device on which this disk is mounted will be requested for the job. After deletion of the file generation, the device will be returned to the system.
- For files stored on private disks, if OPTION=*DESTROY-ALL or MOUNT=*ALL-DISK is specified then all the volumes which hold the file must be already mounted at the time the command is executed. Otherwise, only the first volume which holds the file has to be mounted. The following volumes will not be requested until the command is being executed. This rule also applies if a partially qualified file name is specified in the DELETE-FILE command, thus addressing a number of files. In this situation, it is not necessary that all the volumes for all of the files are mounted simultaneously. The system will determine which of the files requires the most devices, and will request the corresponding number of devices.

Examples

Example 1: Deleting selected files; in this case, files that have names starting with 'D.' followed by one of the numbers from '1' to '8'

```
/show-file-attr inf=*minimum                                                       (1)
%N NNN NW              3 :2OS2:$USER1.D.1
%N NNN NW              3 :2OS2:$USER1.D.10
%N NNN R--------       3 :2OS2:$USER1.D.2
%S NNN NW             60 :2OS2:$USER1.D.3
%N NNN NR              3 :2OS2:$USER1.D.4
%N NNN GUARDS          3 :2OS2:$USER1.D.5
%N NNN NW              3 :2OS2:$USER1.D.6
%N NYN NW              3 :2OS2:$USER1.D.7
%N NNN NW              3 :2OS2:$USER1.D.8
%N NNN NW              3 :2OS2:$USER1.D.9
%S NNN NW              6 :2OS2:$USER1.LST.ADDCMD
%S NNN NW            333 :2OS2:$USER1.LST.DOMAIN.D
%S NNN NW            333 :2OS2:$USER1.LST.DOMAIN.E
%S NNN NW             24 :2OS2:$USER1.LST.HELP
%S NNN NW             24 :2OS2:$USER1.LST.RFA.416
%S NNN NW             66 :2OS2:$USER1.LST.SDF.D.1
%S NNN NW             30 :2OS2:$USER1.LST.SDF.E
%N NNN NW              3*:2OS2:$USER1.MAX.DISK-FILE.1
%N NNN NW              3 :2OS2:$USER1.MAX.FILE.1
%N NNN RWX------       3 :2OS2:$USER1.MAX.FILE.2
%N NNN GUARDS          3 :2OS2:$USER1.MAX.FILE.3
%N NNN NW              0*:2OS2:$USER1.MAX.GROUP.1 (FGG)
%N YNN NW              0 :2OS2:$USER1.MAX.GROUP.2 (FGG)
%N NNN NW              0 :2OS2:$USER1.MAX.GROUP.3 (FGG)
%N NNN YW                :2OS2:$USER1.MAX.TAPE-FILE.1
%I NNN NW            384 :2OS2:$USER1.SF.NEU
%I NNN NW             48 :2OS2:$USER1.SF.ROBAR
%I NNN NW             123 :2OS2:$USER1.SF.TEST.DEV.1

/del-file :*:d.<1:8>,dialog-control=*catalog-change                                (2)
%  DMS0516 DELETE FILE(S) ':2OBU:$USER1.D.<1:8>' ? REPLY (Y=YES; N=NO; T=TER
MINATE COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y                                    (3)
%  DMS0801 ERROR WHEN DELETING FILE ':2OBU:$USER1.D.<1:8>'
%  DMS051B REQUESTED USER ID NOT IN PUBSET 2OBU
%  DMS0516 DELETE FILE(S) ':2ORZ:$USER1.D.<1:8>' ? REPLY (Y=YES; N=NO; T=TER
MINATE COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y                                    (4)
%  DMS0801 ERROR WHEN DELETING FILE ':2ORZ:$USER1.D.<1:8>'
%  DMS051B REQUESTED USER ID NOT IN PUBSET 2ORZ
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.<1:8>' ? REPLY (Y=YES; N=NO; T=TER
MINATE COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)??                                    (5)
%  ? The user can decide, in interactive mode, whether or not to execute the
%    specified delete task.
%    (&00): partially qualified file name, or name of a file, a file generation
%           or a file generation group.
%    Y: the file, file generation or file generation group will be deleted.
%    N: the displayed file, file generation or file generation group will
%       not be deleted.
%    T: processing of the entered delete command will be terminated.
%    ?: explanations of the additional options will be displayed, followed
%       by renewed display of the task.
%       The meanings of the operands of the possible additional options are
%       analogous to those of the corresponding operands in the command and
%       may be looked up in the 'Control System Command Language' manual.
%       The following additional options may be added, separated by a comma,
%       to the response ('Y' or 'N' or 'T'):
%    ,CHECK:    update the current dialog form:
%               ,CHECK=NO or =PVS or =MULTIPLE or =SINGLE or =ERROR.
%    ,IGNORE:   redefine the protection attribute before processing the delete
%               task:
```

```
%                ,IGNORE=ACCESS or =EXDATE or ,IGNORE=(ACCESS,EXDATE).
%    ,PASSWORD: reassign the passwords before processing the delete task.
%               A maximum of 3 (hexadecimal or numeric) passwords are
%    permitted.
%               ,PASSWORD=password or =(password,...).
%  ! Enter the appropriate reply for the explanation in the 'Meaning' text.
%    If the reply is invalid or if an empty string is entered, the task
%    will not be processed.
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.<1:8>' ? REPLY (Y=YES; N=NO; T=TERM
INATE COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y,check=single   ─────────────────────  (6)
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.1' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.2' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y
%  DMS0801 ERROR WHEN DELETING FILE ':2OS2:$USER1.D.2'
%  DMS0666 REQUESTED ACCESS TO FILE NOT PERMITTED DUE TO EXISTING FILE PROTEC
TION. COMMAND NOT PROCESSED                                   ─────────────────────  (7)
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.3' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y
%  DMS0801 ERROR WHEN DELETING FILE ':2OS2:$USER1.D.3'
%  DMS05C3 FILE TO BE DELETED IS IN USE. RETRY COMMAND LATER ──────────────────────  (8)

%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.4' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y
%  DMS0801 ERROR WHEN DELETING FILE ':2OS2:$USER1.D.4'
%  DMS06D5 FILE IS READ-ONLY. CHECK FILE. MODIFY ACCESS TYPE AND REENTER COMM
AND
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.4' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y,ignore=access     ──────────────────────  (9)
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.5' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y
%  DMS0801 ERROR WHEN DELETING FILE ':2OS2:$USER1.D.5'       ────────────────────── (10)
%  DMS0666 REQUESTED ACCESS TO FILE NOT PERMITTED DUE TO EXISTING FILE PROTEC
TION. COMMAND NOT PROCESSED
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.6' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.7' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y                   ────────────────────── (11)
%  DMS0801 ERROR WHEN DELETING FILE ':2OS2:$USER1.D.7'
%  DMS05BF FILE PASSWORD-PROTECTED. FIRST ENTER CORRECT PASSWORD VIA APPROPRI
ATE COMMAND, THEN REENTER COMMAND FOR DELETION
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.7' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y,password=1234        ─────────────────── (12)
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.8' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y
/show-file-attr d.                  ──────────────────────────────────────────────── (13)
%        3 :2OS2:$USER1.D.10
%        3 :2OS2:$USER1.D.2
%       60 :2OS2:$USER1.D.3
%        3 :2OS2:$USER1.D.5
%        3 :2OS2:$USER1.D.9
%:2OS2: PUBLIC:      5 FILES RES=        72 FRE=       36  REL=        36 PAGES
/show-system-file-assignment        ──────────────────────────────────────────────── (14)
%PROCEDURE LEVEL NUMBER 0
%SYSCMD  : (PRIMARY)
%SYSDTA  : (PRIMARY)
%SYSIPT  : NOT ASSIGNED
%SYSOUT  : (PRIMARY)
%SYSLST  : :2OS2:$USER1.D.3
%SYSOPT  : (PRIMARY)
%TASKLIB : (PRIMARY)
```

(1) Lists all files of the user ID *USER1* on the default pubset *2OS2*. The most important file attributes are shown in condensed form (INFORMATION=*MINIMUM).

(2) All files that have names which begin with *D.*, end with one of the numbers from 1 to 8, and which comprise a total of 3 characters are to be deleted. In order to do this, all file catalogs that can be accessed by the user are searched (with the wildcard * in the catalog ID). In addition, an interactive prompt is to be issued for all selected files to be deleted (DIALOG-CONTROL=*CATALOG-CHANGE).

(3) (Prompt to verify whether the searched file set on pubset *2OBU* is to be deleted. The positive response that was indicated with *Y* is rejected, since no files which match the sequence of wildcards exist on the pubset.

(4) (Prompt to verify whether the searched file set on pubset *2ORZ* is to be deleted. The positive response that was indicated with *Y* is rejected, since no files which match the sequence of wildcards exist on the pubset.

(5) (Prompt to verify whether the searched file set on pubset *2OS2* is to be deleted. The *?* is used to request an explanation for the various responses that can be made.

(6) The response to the prompt to verify whether the searched file set on pubset *2OS2* is to be deleted is *Y*. In addition, the dialog control mode is set to *CHECK=SINGLE*, which means that a prompt is to be displayed for each file that is found.

(7) The file *D.2* cannot be deleted, since it is protected by a BASIC-ACL and the user (OWNER) has no write authorization, see item (1).

(8) The file *D.3* cannot be deleted because it is still open, see item (14).

(9) The file *D.4* is protected against write access using ACCESS=READ. The response to the second prompt for the file is *Y,IGNORE=ACCESS*, so the file can now be deleted.

(10) The file *D.5* cannot be deleted because it is protected using GUARDS and because the user has no write authorization (see example 2).

(11) The file *D.7* cannot be deleted, since it is protected against write access by a password and the user has not yet entered the password into his password table.

(12) When the second prompt appears, the required password is also specified. This allows the file to be deleted; however, the specified password is not entered in the password table of the task. The only possible way to do that is by using the ADD-PASSWORD command.

(13) Output of all file names beginning with *D.*

(14) Output of the system files along with their assignments: The file *D.3* is assigned to the system file SYSDTA and could not be deleted as a result (see item 8).

Example 2: Deleting files which are protected against write access by means of a BASIC-ACL or GUARDS

```
/show-file-attr d.<2,5>,inf=(security=*yes)  ————————————————————————————————————  (1)
%0000000003 :2OS2:$USER1.D.2
%  ------------------------------ SECURITY     ------------------------------
%  READ-PASS  = NONE        WRITE-PASS = NONE        EXEC-PASS  = NONE
%  USER-ACC   = OWNER-ONLY  ACCESS     = WRITE       ACL        = NO
%  OWNER      = R - -       GROUP      = - - -       OTHERS     = - - -
%  AUDIT      = NONE        FREE-DEL-D = *NONE       EXPIR-DATE = NONE
%  DESTROY    = NO          FREE-DEL-T = *NONE       EXPIR-TIME = NONE
%  SP-REL-LOCK= NO          ENCRYPTION = *NONE
%0000000003 :2OS2:$USER1.D.5
%  ------------------------------ SECURITY     ------------------------------
%  READ-PASS  = NONE        WRITE-PASS = NONE        EXEC-PASS  = NONE
%  USER-ACC   = OWNER-ONLY  ACCESS     = WRITE       ACL        = NO
%  OWNER      = R W X       GROUP      = - - -       OTHERS     = - - -
%  AUDIT      = NONE        FREE-DEL-D = *NONE       EXPIR-DATE = NONE
%  DESTROY    = NO          FREE-DEL-T = *NONE       EXPIR-TIME = NONE
%  SP-REL-LOCK= NO          ENCRYPTION = *NONE
%  GUARD-READ = $USER1.OWN-PROT
%  GUARD-WRIT = NONE
%  GUARD-EXEC = NONE
%:2OS2: PUBLIC:      2 FILES RES=         6 FRE=         3 REL=         3 PAGES
/delete-file d.<2,5>,dialog-control=*file-change,ignore-protection=*access ————————  (2)
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.2' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y      ———————————————————————————————————  (3)
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.D.5' ? REPLY (Y=YES; N=NO; T=TERMINAT
E COMMAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y      ———————————————————————————————————  (4)
/show-file-attr d.<2,5>                         ———————————————————————————————————  (5)
%  DMS06CC NO FILE CORRESPONDING TO SPECIFIED OPERANDS
```

(1) Shows the protection attributes of the files *D.2* and *D.5*. Access control for the file *D.2* is implemented by a BASIC-ACL. The user (OWNER) has no write authorization, i.e. cannot delete the file without a change in the protection attributes. Access control for the file *D.5* is implemented using GUARDS. Since no guard has been defined for write access, no write access is granted to any user, i.e. the user cannot delete his or her file unless the protection attributes are changed.

(2) The two files are to be deleted by ignoring any access protection that may have been defined (with ACCESS, BASIC-ACL or GUARDS). The deletion of each file is to be verified by means of a prompt.

(3) On confirmation of the prompt with *Y*, the file *D.2* is deleted.

(4) On confirmation of the prompt with *Y*, the file *D.5* is deleted.

(5) The file catalog now contains no information on these files.

Example 3: Selecting files using wildcards in the file name

```
/show-file-attr  <lst,max>.  ──────────────────────────────────────────────────  (1)
%        6 :20S2:$USER1.LST.ADDCMD
%        3 :20S2:$USER1.LST.BSP.2
%      333 :20S2:$USER1.LST.DOMAIN.D
%      333 :20S2:$USER1.LST.DOMAIN.E
%       24 :20S2:$USER1.LST.HELP
%       24 :20S2:$USER1.LST.RFA.416
%       66 :20S2:$USER1.LST.SDF.D.1
%       30 :20S2:$USER1.LST.SDF.E
%        3*:20S2:$USER1.MAX.DISK-FILE.1
%        3 :20S2:$USER1.MAX.FILE.1
%        3 :20S2:$USER1.MAX.FILE.2
%        3 :20S2:$USER1.MAX.FILE.3
%        0*:20S2:$USER1.MAX.GROUP.1 (FGG)
%        0 :20S2:$USER1.MAX.GROUP.2 (FGG)
%        0 :20S2:$USER1.MAX.GROUP.3 (FGG)
%          :20S2:$USER1.MAX.TAPE-FILE.1
%:20S2: PUBLIC:     13 FILES RES=       828 FRE=        26 REL=        18 PAGES
%:20S2: PRDISC:      2 FILES RES=         3 FRE=         3 REL=         0 PAGES
%:20S2: TAPE  :      1 FILE
/delete-file ///.file.2,dialog-control=*file-change  ───────────────────────────  (2)
/show-file-attr ///.file.  ─────────────────────────────────────────────────────  (3)
%         3 :20S2:$USER1.MAX.FILE.1
%         3 :20S2:$USER1.MAX.FILE.3
%:20S2: PUBLIC:      2 FILES RES=         6 FRE=         6 REL=         6 PAGES
```

(1) Shows all files for which the first partial name is *MAX* or *LST*.

(2) Selects files for deletion if their first partial name consists of exactly three characters and if the file name ends in *FILE.2*.

(3) Only one file was selected, i.e. the file *MAX.FILE.2*. Since this was the only file available for selection, it was deleted without confirmation.

DELETE-FILE-GENERATION

Delete file generations from file generation group

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE-GENERATION-GROUP
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE TSOS
Routing code:	$ (with NBCONOPI=N) or E (with NBCONOPI=Y)

Function

With the DELETE-FILE-GENERATION command, a user can delete file generations which are cataloged under his own user ID, and selected according to criteria which must be specified in the command. Specific deletion operations allow the user to release storage space. The user is assisted in doing this by a guided dialog. The operands of the DELETE-FILE-GENERATION command can be divided into four groups, which correspond to the different functional levels of the command (see “Overview of functions” on page 3-74).

Selection

Using the selection parameters, the user specifies which files generations/ catalog entries are to be processed. The attributes stored in the group entries are used as selection criteria. For this purpose, some of the operands in the DELETE-FILE-GENERATION command correspond with those of the SHOW-FILE-ATTRIBUTES command.

File protection

File protection parameters make it unnecessary for the user to input a MODIFY-FILE-GROUP-ATTRIBUTES or ADD-PASSWORD command in order to delete files for which file protection attributes such as passwords, retention period, etc. are defined.

Action parameters

Action parameters control the internal execution of the DELETE-FILE-GENERATION command. The user can, on the one hand, determine the scope of the deletions, and on the other hand can define conditions for deletion.

Control parameters

Control parameters permit the user to define his own user interface, within certain limits. For example, he can use interactive prompts in the dialog, or can request a SYSOUT log, or have DMS errors ignored (for the spin-off mechanism or SDF-P error handling).

Privileged functions

The privileged functions for system support staff include specification of any user ID, the ability to ignore protection attributes of the file generations being deleted and, in the case of a user ID change, to branch to guided dialog.

By default, systems support (TSOS privilege) is a co-owner of all the files (and can therefore delete file generations under any user ID). When SECOS is used, this co-ownership can be restricted.

In conjunction with the SECOS software product a user can allow other user IDs to act as co-owners of the file generation groups of his user ID. Co-owners of a user ID are also allowed to delete file generations under that ID.

Overview of functions

	Function / Meaning	**Level 1 operands**	**Level 2/3 operands**
3-81	The name of the file generation which is to be deleted	GENERATION-NAME	
3-81	File generation to be deleted, specified in relative form	DELETE =*GENERATIONS-BEFORE =*GENERATIONS-AFTER	
3-82	Criteria for selection of the file generations to be deleted	SELECT= *BY-ATTRIBUTES	
	Date specification		
3-82	– Expiration date (implicit retention period)		EXPIRATION-DATE
	– Time		TIME
3-93	– Date when file is freed for deletion		FREE-FOR-DELETION
	– Time		TIME
	Storage space		
3-87	– Type of volume		SUPPORT
	– Public disk or private disk		=*ANY
	– Private disk		=*PRIVATE-DISK

Table 46: Overview of DELETE-FILE-GENERATION command functions (Part 1 of 3)

	Function / Meaning	Level 1 operands	Level 2/3 operands
	File security/file protection		
3-87	– Access		ACCESS
3-88	– Shareability		USER-ACCESS
3-90	– Basic access control list		BASIC-ACL
3-91	– Password protection		PASSWORD
3-88	– Access control using GUARDS		GUARDS
3-90	– protection against release of storage space		SPACE-RELEASE-LOCK
3-89	– Code table (CCS)		CODED-CHARACTER-SET
3-88	– BACKUP level		BACKUP-CLASS
3-89	– Ability to migrate from processing level (S0) to background level (S1 or S2) when the HSMS software product is in use		MIGRATE
3-92	– HSMS management class (MANAGEMENT-CLASS operand)		MANAGEMENT-CLASS
3-97	– Work file attribute		WORK-FILE-GROUP
	Deletion parameters	OPTION	
3-98	– Delete the catalog entry and release storage space	=*ALL	
3-98	– Release storage space only, retain catalog entry	=*SPACE	
3-98	– Logical deletion: delete the data-related attributes of the file, change the catalog entry accordingly, retain the storage space allocation	=*DATA	
3-98	– Logical deletion as with *DATA, but the data-related attributes of the file are retained	=*DATA-KEEP-ATTRIBUTES	
3-99	– Delete the catalog entry, release the storage space and overwrite its contents	=*DESTROY-ALL	
3-99	Mount request	MOUNT	

Table 46: Overview of DELETE-FILE-GENERATION command functions (Part 2 of 3)

	Function / Meaning	Level 1 operands	Level 2/3 operands
	Control parameters	DIALOG-CONTROL	
3-100	– User intervention not permitted (default setting for procedures and batch mode)	=*NO	
3-101	– Dialog when more than one file is referenced if the file name specified is not fully qualified (default setting for interactive mode)	=*MORE-THAN-ONE-FILE	
3-100	– Dialog when an error occurs	=*ERROR	
3-101	– Dialog when catalog ID changes	=*CATALOG-CHANGE	
3-101	– For each file selected, there will be an interactive dialog with the user to determine whether the current command should be processed	=*FILE-CHANGE	
3-101	– Dialog when user ID changes	=*USER-ID-CHANGE	
3-101	Message for successfully deleted files	OUTPUT	
	– Suppress	=*NO	
	– Output to SYSOUT	=*SYSOUT	
3-102	Ignore protection attributes	IGNORE-PROTECTION	
	– Ignore write protection provided by ACCESS=READ	=*ACCESS	
	– Ignore retention periods	=*EXPIRATION-DATE	
	– Ignore password protection	=*WRITE-PASSWORD / *READ-PASSWORD	
3-102	Ignore password protection provided by specified passwords	PASSWORDS-TO-IGNORE	
3-103	Suppress error situations	SUPPRESS-ERRORS	

Table 46: Overview of DELETE-FILE-GENERATION command functions (Part 3 of 3)

Format

(Part 1 of 5)

```
DELETE-FILE-GENERATION                                                     Alias: DLFGN

 GENERATION-NAME = *DUMMY / <filename 1..54 without-vers with-wild(80)>
,DELETE = *GENERATIONS-BEFORE / *GENERATIONS-AFTER
,SELECT = *ALL / [*BY-ATTRIBUTES](...)
   [*BY-ATTRIBUTES](...)
     |  EXPIRATION-DATE = *ANY / *NONE / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
     |                    <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)
     |    *TOMORROW(...)
     |      |  TIME = *ANY / [*INTERVAL](...)
     |      |    [*INTERVAL](...)
     |      |      |  FROM = 00:00:00 / <time>
     |      |      | ,TO = 23:59:59 / <time>
     |    *TODAY(...)
     |      |  TIME = *ANY / [*INTERVAL](...)
     |      |    [*INTERVAL](...)
     |      |      |  FROM = 00:00:00 / <time>
     |      |      | ,TO = 23:59:59 / <time>
     |    *YESTERDAY(...)
     |      |  TIME = *ANY / [*INTERVAL](...)
     |      |    [*INTERVAL](...)
     |      |      |  FROM = 00:00:00 / <time>
     |      |      | ,TO = 23:59:59 / <time>
     |    <integer -99999..991231>(...)
     |      |  TIME = *ANY / [*INTERVAL](...)
     |      |    [*INTERVAL](...)
     |      |      |  FROM = 00:00:00 / <time>
     |      |      | ,TO = 23:59:59 / <time>
     |    <date>(...)
     |      |  TIME = *ANY / [*INTERVAL](...)
     |      |    [*INTERVAL](...)
     |      |      |  FROM = 00:00:00 / <time>
     |      |      | ,TO = 23:59:59 / <time>
```

(Part 2 of 5)

```
        *INTERVAL(...)
            FROM = *EARLIEST / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
                        <integer -99999..991231>(...) / <date>(...)
               *TOMORROW(...)
                    TIME = 00:00:00 / <time>
               *TODAY(...)
                    TIME = 00:00:00 / <time>
               *YESTERDAY(...)
                    TIME = 00:00:00 / <time>
               <integer -99999..991231>(...)
                    TIME = 00:00:00 / <time>
               <date>(...)
                    TIME = 00:00:00 / <time>
           ,TO = *LATEST / TODAY(...) / *TOMORROW(...) / *YESTERDAY(...) /
                    <integer -99999..991231>(...) / <date>(...) / *ANY
               *TODAY(...)
                    TIME = 23:59:59 / <time>
               *TOMORROW(...)
                    TIME = 23:59:59 / <time>
               *YESTERDAY(...)
                    TIME = 23:59:59 / <time>
               <integer -99999..991231>(...)
                    TIME = 23:59:59 / <time>
               <date>(...)
                    TIME = 23:59:59 / <time>
  ,SUPPORT = *ANY / *PRIVATE-DISK
  ,ACCESS = *ANY / *READ / *WRITE
  ,PASSWORD = *ANY / list-poss(3): *NONE / *READ-PASSWORD / *WRITE-PASSWORD
  ,USER-ACCESS = *ANY / list-poss(2): *OWNER-ONLY / *ALL-USERS
  ,BACKUP-CLASS = *ANY / list-poss(5): *A / *B / *C / *D / *E
  ,MIGRATE = *ANY / list-poss(3): *ALLOWED / *INHIBITED / *FORBIDDEN
  ,CODED-CHARACTER-SET = *ANY / *NONE / <name 1..8>
  ,SPACE-RELEASE-LOCK = *ANY / *NO / *YES
```

(Part 3 of 5)

```
,BASIC-ACL = *ANY / *NONE / *YES / [*PARAMETERS](...)

   [*PARAMETERS](...)

        OWNER = *ANY / *NO-ACCESS / [*PARAMETERS](...)

           [*PARAMETERS](...)

                READ = *ANY / *NO / *YES
               ,WRITE = *ANY / *NO / *YES

       ,GROUP = *ANY / *NO-ACCESS / [*PARAMETERS](...)

           [*PARAMETERS](...)

                READ = *ANY / *NO / *YES
               ,WRITE = *ANY / *NO / *YES

       ,OTHERS = *ANY / *NO-ACCESS / [*PARAMETERS](...)

           [*PARAMETERS](...)

                READ = *ANY / *NO / *YES
               ,WRITE = *ANY / *NO / *YES

,GUARDS = *ANY / *YES / *NO / [*PARAMETERS](...)

   [*PARAMETERS](...)

        READ = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>
       ,WRITE = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>

,MANAGEMENT-CLASS = *ANY / *NONE / <composed-name 1..8>

,FREE-FOR-DELETION = *ANY / *NONE / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
                        <integer -99999..99999>(...) / <date>(...) / *INTERVAL(...)

   *TOMORROW(...)

        TIME = *ANY / *INTERVAL(...)

           *INTERVAL(...)
                FROM = 00:00:00 / <time>
               ,TO = 23:59:59 / <time>

   *TODAY(...)
        TIME = *ANY / *INTERVAL(...)

           *INTERVAL(...)
                FROM = 00:00:00 / <time>
               ,TO = 23:59:59 / <time>
```

(Part 4 of 5)

```
*YESTERDAY(...)
    TIME = *ANY / *INTERVAL(...)

      *INTERVAL(...)
          FROM = 00:00:00 / <time>
         ,TO = 23:59:59 / <time>

<integer -99999..99999>(...)
    TIME = *ANY / *INTERVAL(...)

      *INTERVAL(...)
          FROM = 00:00:00 / <time>
         ,TO = 23:59:59 / <time>

<date>(...)
    TIME = *ANY / *INTERVAL(...)

      *INTERVAL(...)
          FROM = 00:00:00 / <time>
         ,TO = 23:59:59 / <time>

*INTERVAL(...)

    FROM = *EARLIEST / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
                <integer -99999..99999>(...) / <date>(...)
      *TOMORROW(...)
          TIME = 00:00:00 / <time>

      *TODAY(...)
          TIME = 00:00:00 / <time>
      *YESTERDAY(...)
          TIME = 00:00:00 / <time>

      <integer -99999..99999>(...)
          TIME = 00:00:00 / <time>

      <date>(...)
          TIME = 00:00:00 / <time>

   ,TO = *LATEST / *TODAY(...) / *TOMORROW(...) / *YESTERDAY(...) /
             <integer -99999..99999>(...) / <date>(...)

      *TODAY(...)
          TIME = 23:59:59 / <time>

      *TOMORROW(...)
          TIME = 23:59:59 / <time>
```

(Part 5 of 5)

```
        |         |    *YESTERDAY(...)
        |         |       |  TIME = 23:59:59 / <time>
        |         |
        |         |    <integer -99999..99999>(...)
        |         |       |  TIME = 23:59:59 / <time>
        |         |
        |         |    <date>(...)
        |         |       |  TIME = 23:59:59 / <time>
        |
        |  ,WORK-FILE-GROUP = *ANY / *NO / *YES
,OPTION = *ALL / *SPACE / *DATA / *DATA-KEEP-ATTRIBUTES / *DESTROY-ALL
,MOUNT = *FIRST-DISK / *ALL-DISKS
,DIALOG-CONTROL = *STD / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE /
                  *CATALOG-CHANGE / *USER-ID-CHANGE
,OUTPUT = *STD / *NO / *SYSOUT
,IGNORE-PROTECTION = *NONE / list-poss(4): *ACCESS / *EXPIRATION-DATE / *WRITE-PASSWORD /
                     *READ-PASSWORD
,PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> /
                       <integer -2147483648..2147483647>
,SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
```

Operands

GENERATION-NAME = *DUMMY / <filename 1..54 without-vers with-wild(80)>
The name of a reference generation, from which or up to which the files are to be deleted (see the DELETE operand). Only the user’s own user ID or a user ID for which the user is co-owner may be specified.

GENERATION-NAME = *DUMMY
Designates the dummy generation *DUMMY, which “always exists”, and which satisfies all selection criteria. Any other operands (except for DIALOG-CONTROL, OUTPUT and SUPPRESS-ERRORS) will be checked for their formal correctness, but otherwise ignored. The main use of *DUMMY is in testing procedures.

DELETE = *GENERATIONS-BEFORE / GENERATIONS-AFTER
Specifies which file generations are to be deleted: all those whose number is less than the reference generation, or all those whose number is greater than it.

SELECT = *ALL
All the files in the set specified by GENERATION-NAME are selected.

SELECT = *BY-ATTRIBUTES(...)
Restricts the files in the set specified in GENERATION-NAME to files which satisfy the criteria which follow. The default values, *ANY and ANY, each indicate that the set of file generations is not to be restricted to any particular values for the attribute concerned.

EXPIRATION-DATE = *ANY / *NONE / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)
The user can select the file generation to be deleted on the basis of the expiration date. A retention period exists if the expiration date is greater than the current date. The selection can be restricted for a specified date to a time interval in which the last file update occurred (see the TIME operand for each structure). Note that at present the time 00:00:00 is always entered in the file catalog as the time of the expiration date. See also the *EXPIR-DATE* and *EXPIR-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

EXPIRATION-DATE = *ANY
The expiration date is not to be used as a selection criterion.

EXPIRATION-DATE = *NONE
Only the file generations for which no release date (NONE) has been entered as the EXPIRATION-DATE in the group entry will be deleted, i.e. file generations which have never been opened.

EXPIRATION-DATE = *TOMORROW(...)
Only those file generations which have tomorrow’s date entered as the EXPIRATION-DATE in their group entry are deleted.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes only those file generations whose expiration date lies within the specified time interval that follows.

FROM = 00:00:00 / <time>
Selects file generations for which the time of expiration ≥ the specified time.

TO = 23:59:59 / <time>
Selects file generations for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = *TODAY(...)
Only those file generations which have today’s date entered as the EXPIRATION-DATE in their group entry are deleted.

TIME = _*ANY_ / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes only those file generations whose expiration date lies within the specified time interval that follows.

FROM = _00:00:00_ / <time>
Selects file generations for which the time of expiration ≥ the specified time.

TO = _23:59:59_ / <time>
Selects file generations for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = *YESTERDAY(...)
Only those file generations which have yesterday’s date entered as the EXPIRATION-DATE in their group entry are deleted.

TIME = _*ANY_ / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes only those file generations whose expiration date lies within the specified time interval that follows.

FROM = _00:00:00_ / <time>
Selects file generations for which the time of expiration ≥ the specified time.

TO = _23:59:59_ / <time>
Selects file generations for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = <integer -99999..991231>(...)
Only those file generations which have the specified date entered as the EXPIRATION-DATE in their group entry are deleted. Here, the user can specify the expiration date in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value*: (6 digits, with preceding sign) the number of days from today's date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ +0 or TOMORROW ≙ +1)

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = INTERVAL(...)
Deletes only those file generations whose expiration date lies within the specified time interval that follows.

FROM = 00:00:00 / <time>
Selects file generations for which the time of expiration ≥ the specified time.

TO = 23:59:59 / <time>
Selects file generations for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = <date>(...)
Only those file generations which have the specified date entered as the EXPIRATION-DATE in their group entry are deleted. The user can specify the expiration date in the form [yy]yy-mm-dd.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of files to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes only those file generations whose expiration date lies within the specified time interval that follows.

FROM = 00:00:00 / <time>
Selects file generations for which the time of expiration ≥ the specified time.

TO = 23:59:59 / <time>
Selects file generations for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = *INTERVAL(...)
Only file generations whose release date lies within the specified time period, i.e. whose retention period ends within the specified time period, will be deleted. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified for EXPIRATION-DATE = <integer...>. It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). Whichever of the operands is not specified will be replaced by the default value for use as the limit of the range. The use of range limits for deletion can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = <u>*EARLIEST</u> / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)
If EARLIEST is specified, the system will use the earliest possible date. Only those file generations for which the retention period expires on or after the specified date (EXPIRATION-DATE ≥ specified date) are deleted.

FROM = *TOMORROW(...)
Deletes only those file generations for which the EXPIRATION-DATE ≥ date of next day.

TIME = <u>00:00:00</u> / <time>
Time on the specified date. All file generations that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = *TODAY(...)
Deletes only those file generations for which the EXPIRATION-DATE ≥ current date.

TIME = <u>00:00:00</u> / <time>
Time on the specified date. All file generations that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = *YESTERDAY(...)
Deletes only those file generations for which the EXPIRATION-DATE ≥ date of preceding day.

TIME = <u>00:00:00</u> / <time>
Time on the specified date. All file generations that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = <integer -99999..991231>(...)
Deletes only those file generations for which the EXPIRATION-DATE ≥ the specified date.

TIME = 00:00:00 / <time>
Time on the specified date. All file generations that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = <date>(...)
Deletes only those file generations for which the EXPIRATION-DATE ≥ the specified date.

TIME = 00:00:00 / <time>
Time on the specified date. All file generations that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = *LATEST / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)
Only those file generations for which the retention period expires on or before the specified date (EXPIRATION-DATE ≤ specified date) will be deleted.

TO = *TOMORROW(...)
Deletes only those file generations for which the EXPIRATION-DATE ≤ date of next day.

TIME = 23:59:59 / <time>
Time on the specified date. All file generations that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = *TODAY(...)
Deletes only those file generations for which the EXPIRATION-DATE ≤ current date.

TIME = 23:59:59 / <time>
Time on the specified date. All file generations that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = *YESTERDAY(...)
Deletes only those file generations for which the EXPIRATION-DATE ≤ date of preceding day.

TIME = 23:59:59 / <time>
Time on the specified date. All file generations that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = <integer -99999..991231>(...)
Deletes only those file generations for which the EXPIRATION-DATE ≤ the specified date.

TIME = 23:59:59 / <time>
Time on the specified date. All file generations that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = <date>(...)
Deletes only those file generations for which the EXPIRATION-DATE ≤ the specified date.

TIME = 23:59:59 / <time>
Time on the specified date. All file generations that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

SUPPORT = *ANY / *PRIVATE-DISK
The user can select which file generations are to be deleted by the type of their volumes.

SUPPORT = *ANY
The volume type is not to be used as a selection criterion.

SUPPORT = *PRIVATE-DISK
Only file generations on private disks are to be deleted.

ACCESS = *ANY / *READ / *WRITE
The specified access type will be used as a selection criterion (ACCESS value in the group entry.)

ACCESS = ANY
The ACCESS value is not to be used as a selection criterion.

ACCESS = *READ
Only file generations for which write access is prevented by ACCESS=READ, i.e. for which only read access is permitted, will be deleted.

ACCESS = *WRITE
Only file generations for which write access is permitted will be deleted.

PASSWORD = <u>*ANY</u> / list-poss(4): *NONE / *READ-PASSWORD / *WRITE-PASSWORD / *EXEC-PASSWORD
The user can select file generations by specific password types for processing.

PASSWORD = <u>*ANY</u>
Password protection is not to be used as a selection criterion.

PASSWORD = *NONE
Only file generations for which there is no password protection are to be deleted.

PASSWORD = *READ-PASSWORD
Only file generations which are protected by a read password are to be deleted.

PASSWORD = *WRITE-PASSWORD
Only file generations which are protected by a write password are to be deleted.

USER-ACCESS = <u>*ANY</u> / list-poss(2): *OWNER-ONLY / *ALL-USERS
The user can select file generations using their access authorizations as a criterion.

USER-ACCESS = <u>*ANY</u>
The access authorization is not to be used as a selection criterion.

USER-ACCESS = *OWNER-ONLY
Only file generations which the owner alone may access are to be deleted.

USER-ACCESS = *ALL-USERS
Only file generations which permit access to all users are to be deleted.

BACKUP-CLASS = <u>*ANY</u> / list-poss(5): *A / *B / *C / *D / *E
The user can select file generations for deletion by their BACKUP-CLASS level.

BACKUP-CLASS = <u>*ANY</u>
The BACKUP-CLASS level is not to be used as a selection criterion.

BACKUP-CLASS = *A
Only file generations for which the group entry contains the value BACKUP=A (most frequent backup) are to be processed.

BACKUP-CLASS = *B
Only file generations for which the group entry contains the value BACKUP=B are to be processed.

BACKUP-CLASS = *C
Only file generations for which the group entry contains the value BACKUP=C are to be processed.

BACKUP-CLASS = *D
Only file generations for which the group entry contains the value BACKUP=D are to be processed.

BACKUP-CLASS = *E
Only file generations for which the group entry contains the value BACKUP=E are to be processed.

MIGRATE = <u>*ANY</u> / list-poss(3): *ALLOWED / *INHIBITED / *FORBIDDEN
The user can select which file generations are to be processed by the migration entry for the group (see the CREATE-FILE-GROUP command, MIGRATE operand).

MIGRATE = <u>*ANY</u>
The specified file generations are to be deleted, irrespective of the value in the MIGRATE operand in each of their group entries.

MIGRATE = *ALLOWED
Only file generations for which the group entry specifies the appropriate operand value, i.e. file generations which may be migrated to storage levels S1 and S2, are to be deleted.

MIGRATE = *INHIBITED
Only file generations for which the group entry specifies the appropriate operand value are to be processed, i.e. file generations which may not be migrated.

MIGRATE = *FORBIDDEN
Only file generations for which the catalog entry specifies the appropriate operand value are to be processed, i.e. file generations for which an intensified migration lock is declared. The file generations may not even be migrated for a brief period (e.g. for reorganization purposes).

CODED-CHARACTER-SET = <u>*ANY</u> / *NONE / <name 1..8>
The coded character set defined in the group entry is used as a selection criterion (see the CREATE-FILE command for an explanation; if a CCS is defined, see also the *COD-CH-SET* output field of the SHOW-FILE-ATTRIBUTES command).

CODED-CHARACTER-SET = <u>*ANY</u>
The defined character set is not to be used as a selection criterion.

CODED-CHARACTER-SET = *NONE
Only the file generations for which no coded character set was explicitly defined are selected for deletion.

CODED-CHARACTER-SET = <name 1..8>
Only the files for which the specified character set was defined are selected for deletion.

SPACE-RELEASE-LOCK = *ANY / *NO / *YES
The lock to prevent the release of unused memory space is used as a selection criterion (see the *SP-REL-LOCK* output field in the SHOW-FILE-ATTRIBUTES command).

SPACE-RELEASE-LOCK = *ANY
The permission to release unused memory space is not used as a selection criterion.

SPACE-RELEASE-LOCK = *NO
Only the file generations for which unused memory space may be released are selected for deletion.

SPACE-RELEASE-LOCK = *YES
Only the file generations for which the release of unused memory space is not permitted are selected for deletion.

BASIC-ACL = *ANY / *NONE / *YES / *PARAMETERS(...)
The BASIC-ACL entry in the file catalog is used as a selection criterion.

BASIC-ACL = *ANY
The BASIC-ACL entry is not to be used as a selection criterion.

BASIC-ACL = *NONE
Only the file generations that have no BASIC-ACL entry in the catalog are to be deleted.

BASIC-ACL = *YES
Only the file generations which have a BASIC-ACL entry in the catalog are to be deleted.

BASIC-ACL = *PARAMETERS(...)
Only the file generations which have the specified BASIC-ACL entry in the catalog are to be deleted. NO-ACCESS means that no access rights were granted.

i Access rights specified with the OWNER, GROUP and OTHERS operands within the *PARAMETERS(...) structure are logically ORed.

OWNER = *ANY / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights must already be defined for the owner.

OWNER = *PARAMETERS(...)
Access rights that must be present for the owner (logically ORed together):

READ = *ANY / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = *ANY / *NO / *YES
Specifies whether write access authorization must be present.

GROUP = *ANY / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights should already be defined for the owner's user group.

GROUP = *PARAMETERS(...)
Access rights that must be present for the owner's user group (logically ORed together):

READ = *ANY / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = *ANY / *NO / *YES
Specifies whether write access authorization must be present.

OTHERS = *ANY / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights should already be defined for all other users.

OTHERS = *PARAMETERS(...)
Access rights that must be present for all other users (logically ORed together):

READ = *ANY / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = *ANY / *NO / *YES
Specifies whether write access authorization must be present.

GUARDS = *ANY / *NO / *YES / *PARAMETERS(...)
The use of GUARDS for access control (see the CREATE-FILE-GROUP or MODIFY-FILE-GROUP-ATTRIBUTES command) is used as a selection criterion.

GUARDS = *ANY
Access control with GUARDS is not to be used as a selection criterion.

GUARDS = *NO
Only the file generations which are not protected by GUARDS against unauthorized access are to be deleted (see also the PROTECTION-LEVEL operand).

GUARDS = *YES
Only the file generations which are protected by GUARDS against unauthorized access (i.e. files for which access is controlled via the GUARDS function unit) are to be deleted.

GUARDS = *PARAMETERS(...)
Deletes only those file generations which are protected by GUARDS against unauthorized access as specified, i.e. files for which access control is implemented using GUARDS: Access to the file generation is controlled by a guard, i.e. a special object which contains all the conditions under which a user is granted access authorization (e.g. date, time, user ID). The GUARDS function unit of the chargeable software product SECOS (see the “SECOS” manual [35]) must be installed in order to create and maintain a guard.
Each access mode can be controlled by a separate guard. If no guard (*NONE) is defined for a given access mode, no corresponding access is permitted. If a defined guard is not accessible, the mode of access protected by it is not permitted. If the GUARDS subsystem is not available at the time of accessing the job variable, no access of any kind is allowed for the job variable.

The values specified for the following READ and WRITE operands are logically ORed.

READ = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>
Only the file generations which are protected against unauthorized read access by the specified guard are deleted. The default value *ANY means that the selection of file generations is not based on read access protection by a guard. *NONE selects file generations for which no read access is permitted.

WRITE = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>
Only the file generations which are protected against unauthorized write access by the specified guard are deleted. The default value *ANY means that the selection of file generations is not based on write access protection by a guard. *NONE selects file generations for which no write access is permitted. These file generations can be deleted with IGNORE-PROTECTION=*ACCESS.

MANAGEMENT-CLASS = *ANY / *NONE / <composed-name 1..8>
The selection criterion is the HSMS management class.

MANAGEMENT-CLASS = *ANY
The HSMS management class is not a selection criterion.

MANAGEMENT-CLASS = *NONE
Only deletes file generations with no HSMS management class assigned to them.

MANAGEMENT-CLASS = <composed-name 1..8>
Only deletes file generations with the specified HSMS management class assigned to them.

FREE-FOR-DELETION = *ANY / *NONE / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) / <integer -99999..0>(...) / <date>(...) / *INTERVAL(...)
The date of release of deletion (free-for-deletion date) can be used as a selection criterion for the file generations which are to be deleted. The selection for a specified date can be restricted to a specific period of time on the free-for-deletion date (see the TIME operands in the various structures) See also the SHOW-FILE-ATTRIBUTES command, *FREE-DEL-D* and *FREE-DEL-T* output fields.

FREE-FOR-DELETION = *ANY
The free-for-deletion date is not used as a selection criterion.

FREE-FOR-DELETION = *NONE
Only deletes file generations for which no free-for-deletion date has yet been entered in the catalog.

FREE-FOR-DELETION = *TOMORROW(...)
Only deletes file generations for which the next day is entered as the free-for-deletion date in the catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the file selection to a period relative to the date of the next day.

TIME = *INTERVAL(...)
Only deletes file generations with a free-for-deletion date which falls within the period specified next.

FROM = 00:00:00 / <time>
Selects files with a free-for-deletion date ≥ the specified time.

TO = 23:59:59 / <time>
Selects files with a free-for-deletion date ≤ the specified time.

FREE-FOR-DELETION = *TODAY(...)
Only deletes file generations with today’s date recorded as free-for-deletion date in their catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the selection to a period of time relative to today’s date.

TIME = *INTERVAL(...)
Only deletes file generations with a free-for-deletion date which falls within the period specified next.

FROM = <u>00:00:00</u> / <time>
Selects file generations with a free-for-deletion date ≥ the specified time.

TO = <u>23:59:59</u> / <time>
Selects file generations with a free-for-deletion date ≤ the specified time.

FREE-FOR-DELETION = *YESTERDAY(...)
Only deletes file generations with yesterday's date recorded as free-for-deletion date in their catalog entry.

TIME = <u>*ANY</u> / *INTERVAL(...)
Restricts the selection to a period of time relative to yesterday's date.

TIME = *INTERVAL(...)
Only deletes file generations with a free-for-deletion date which was changed within the period specified next.

FROM = <u>00:00:00</u> / <time>
Selects file generations with a free-for-deletion date ≥ the specified time.

TO = <u>23:59:59</u> / <time>
Selects file generations with a free-for-deletion date ≤ the specified time.

FREE-FOR-DELETION = <integer -99999..99999>(...)
Only deletes file generations with the specified free-for-deletion date recorded in their catalog entry. The free-for-deletion date is specified relative to today's date (in the form -n for the past or +n for the future).

TIME = <u>*ANY</u> / *INTERVAL(...)
Restricts the selection to a period of time relative to the specified date.

TIME = *INTERVAL(...)
Only deletes file generations with a free-for-deletion date which falls within the period specified next.

FROM = 00:00:00 / <time>
Selects file generations with a free-for-deletion date ≥ the specified time.

TO = 23:59:59 / <time>
Selects file generations with a free-for-deletion date ≤ the specified time.

FREE-FOR-DELETION = <date>(...)
Only deletes file generations with exactly the specified date recorded as free-for-deletion date in their catalog entry. The date is specified in the form [yy]yy-mm-dd. 20 is automatically prefixed to two-digit year specifications < 60, 19 to two-digit year specifications ≥ 60.

TIME = *ANY / *INTERVAL(...)
Restricts the selection to a period of time relative to the specified date.

TIME = *INTERVAL(...)
Only deletes file generations with a free-for-deletion date which falls within the period specified next.

FROM = 00:00:00 / <time>
Selects file generations with a free-for-deletion date ≥ the specified time.

TO = 23:59:59 / <time>
Selects file generations with a free-for-deletion date ≤ the specified time.

FREE-FOR-DELETION = *INTERVAL(...)
Only deletes file generations with a free-for-deletion date which falls within the specified period. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified for the FREE-FOR-DELETION= <integer...> operand. It is also possible to specify only the FROM operand (lower limit) or the TO operand (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for information output can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = *EARLIEST / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) / <integer -99999..99999>(...) / <date>(...)
Only deletes file generations with a free-for-deletion date ≥ the specified date.

FROM = *TOMORROW(...)
Only deletes file generations whose free-for-deletion date ≥ date of the next day.

TIME = 00:00:00 / <time>
Time on the specified date. All files that have a free-for-deletion date after the specified time are selected.

FROM = *TODAY(...)
Only deletes file generations with a free-for-deletion date ≥ today’s date.

TIME = 00:00:00 / <time>
Time on the specified date. All file generations that have a free-for-deletion date after the specified time are deleted.

FROM = *YESTERDAY(...)
Only deletes file generations with a free-for-deletion date ≥ yesterday’s date.

TIME = 00:00:00 / <time>
Time on the specified date. All file generations that have a free-for-deletion date after the specified time are deleted.

FROM = <integer -99999..99999>(...)
Deletes only those file generations whose free-for-deletion date ≥ the specified date (in the form -n for the past or +n for the future).

TIME = 00:00:00 / <time>
Time on the specified date. All file generations that have a free-for-deletion date after the specified time are deleted.

FROM = <date>(...)
Only deletes file generations with a free-for-deletion date ≥ the specified date. The date is specified in the form [yy]yy-mm-dd. 20 is automatically prefixed to two-digit year specifications < 60, 19 to two-digit year specifications ≥ 60.

TIME = 00:00:00 / <time>
Time on the specified date. All file generations that have a free-for-deletion date after the specified time are deleted.

TO = *LATEST / *TODAY(...) / *TOMORROW(...) / *YESTERDAY(...) / <integer -99999..99999>(...) / <date>(...)
Only deletes file generations with a free-for-deletion date ≤ the specified date.

TO = *LATEST
There is no upper limit on the selection range based on the free-for-deletion date.

TO = *TODAY(...)
Only deletes file generations with a free-for-deletion date ≤ today’s date.

TIME = 23:59:59 / <time>
Time on the specified date. All file generations that have a free-for-deletion date before the specified time are deleted.

TO = *TOMORROW(...)
Only deletes file generations whose free-for-deletion date ≤ date of the next day.

TIME = 23:59:59 / <time>
Time on the specified date. All files that have a free-for-deletion date before the specified time are selected.

TO = *YESTERDAY(...)
Only deletes file generations with a free-for-deletion date ≤ yesterday's date.

TIME = 23:59:59 / <time>
Time on the specified date. All file generations that have a free-for-deletion date before the specified time are deleted.

TO = <integer -99999..99999>(...)
Deletes only those file generations whose free-for-deletion date ≥ the specified date (in the form -n for the past or +n for the future).

TIME = 23:59:59 / <time>
Time on the specified date. All file generations that have a free-for-deletion date before the specified time are deleted.

TO = <date>(...)
Only deletes file generations with a free-for-deletion date ≤ the specified date. The date is specified in the form [yy]yy-mm-dd. 20 is automatically prefixed to two-digit year specifications < 60, 19 to two-digit year specifications ≥ 60.

TIME = 23:59:59 / <time>
Time on the specified date. All file generations that have a free-for-deletion date before the specified time are deleted.

WORK-FILE-GROUP = *ANY / *NO / *YES
The selection criterion is the work file attribute.

WORK-FILE-GROUP = *ANY
The work file attribute is not a selection criterion.

WORK-FILE-GROUP = *NO
Only deletes file generations which are not marked as work files.

WORK-FILE-GROUP = *YES
Only deletes file generations which are marked as work files.

OPTION = <u>*ALL</u> / *SPACE / *DATA / *DATA-KEEP-ATTRIBUTES / *DESTROY-ALL
Controls the deletion of the file.
If `DESTROY-BY-DELETE=YES` was defined in the catalog entry of the file, the file or released storage space will *definitely* be overwritten with binary zeros. If this is not the case and if OPTION=DESTROY-ALL was also not specified, the file or released storage space will be deleted only logically, rather than being overwritten.

OPTION = <u>*ALL</u>
The default setting for disk files.
The catalog entries for the selected file generations are to be deleted and their storage space will also be released.

OPTION = *SPACE
Only for file generations on public disks (for tape files the default setting ALL applies): the storage space for the file generations deleted by DELETE-FILE-GENERATION is released, but the catalog entry is retained, although amended: it is then identical with a catalog entry created by a CREATE-FILE-GENERATION command and SUPPORT=*NONE.
For private disks the SPACE operand will be rejected.

OPTION = *DATA
Only for disk files (for tape files the default value ALL applies): The data in the selected files is “logically deleted”. After this, the user can no longer access the data, since he is no longer permitted to physically access the volume concerned. The group entry and storage space allocation still exist. The catalog entry is identical to that for a file generation which was created using CREATE-FILE-GENERATION but which has not yet been opened (*FILE-STRUC=NONE*, *CRE-DATE=NONE*).

OPTION = *DATA-KEEP-ATTRIBUTES
Only for disk files (the default ALL applies for tape files):
The data of the file generations involved is “logically deleted” as with OPTION=*DATA, but the data-related file attributes are retained. The data can no longer be addressed by the user.

OPTION = *DESTROY-ALL

Only for disk files (for tape files the default value ALL applies):
the storage space for the affected file generations is released and the group entry is deleted; in addition, the storage space thus released is overwritten with binary zeros so that, if the space is allocated again, nobody can read the old data (data protection). In the case of files on private disks, all the volumes on which the file was stored must be mounted at the time of deletion.
When a file is deleted, the action parameters are evaluated first. See the EXPORT-FILE command for details of exporting file generations. “Data destruction” when a file is deleted can also be permanently recorded in the group entry for the file using the command CREATE-FILE-GROUP; in the group entry a “DESTROY” indicator is then set (DESTROY=YES). In this case storage space is released and it will be automatically overwritten.

MOUNT = <u>*FIRST-DISK</u> / *ALL-DISKS

Only for files on private disks:
Specifies whether or not to mount all the private disks which hold any part of the file which is to be deleted.
The user can request that only the first of the required private disks be mounted, or all of them. The MOUNT operand should be specified if the operands ALL or DESTROY-ALL are specified. For tape files, or files on public disks, any MOUNT specification will be ignored.

MOUNT = <u>*FIRST-DISK</u>

It is only necessary for the first of the private disks, containing the start of the file generation and its group entry, to be online. The associated private disks do not have to be online.

MOUNT = *ALL-DISKS

All the private disks on which parts of the file are held, must be online. If any of the disks is missing, the file generation will not be deleted; on completion of the DELETE-FILE-GENERATION processing, the spin-off mechanism will be triggered.

DIALOG-CONTROL = *STD / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE / *CATALOG-CHANGE / *USER-ID-CHANGE
Specifies whether and under what conditions a verification dialog is to be conducted with the user during the deletion process. The verification dialog mechanism is available only in interactive mode, though that does include procedures. The only operand value that you can specify in batch mode is *STD or *NO.

The user can intervene with the following inputs:
- Y: the specified file or file set will then be deleted.
- N: the specified file or file set will not be deleted.
- T: processing of the command will be terminated.
- ?: the possible responses will be listed, with an explanation of each.

In addition, the following options can be specified, separated by commas:
- ,CHECK = NO The DIALOG-CONTROL mode will be changed to ‘*NO’.
- ,CHECK = PVS The DIALOG-CONTROL mode will be changed to ‘*CATALOG-CHANGE’.
- ,CHECK = MULTIPLE The DIALOG-CONTROL mode will be changed to ‘*MORE-THAN-ONE-FILE’.
- ,CHECK = SINGLE The DIALOG-CONTROL mode will be changed to ‘*FILE-CHANGE’.
- ,CHECK = ERROR The DIALOG-CONTROL mode will be changed to ‘*ERROR’.
- ,IGNORE = list-poss(4): ACCESS / EXDATE / RDPASS / WRPASS Specifies which protection attributes are to be ignored during deletion. Only applies to a verification dialog for a single file. RDPASS and WRPASS are available to privileged users only.
- ,PASSWORD = list-poss(3): <c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647> Enables password-protected files to be deleted (maximum of 3 passwords). Only applies to a verification dialog for a single file.

DIALOG-CONTROL = *STD
The default value *STD corresponds to *MORE-THAN-ONE-FILE in interactive mode (SYSCMD is allocated to the terminal), and in procedures or in batch mode it corresponds to *NO.

DIALOG-CONTROL = *NO
The user cannot intervene in DELETE-FILE-GENERATION processing; all the specified file generations will be deleted (without a verification dialog).

DIALOG-CONTROL = *ERROR
If deletion of the selected file generations proceeds without error, they will be deleted immediately, as when NO is specified (i.e. no verification dialog). However, if a user-correctable error occurs, then a verification dialog takes place as for DIALOG-CONTROL = *FILE-CHANGE. DIALOG-CONTROL = *ERROR applies implicitly if DIALOG-CONTROL = *FILE-CHANGE is set. In the event of an error, the user may acknowledge the error message, abort DELETE-FILE-GENERATION processing or attempt to rectify the error. If he wishes, he can also change the DIALOG-CONTROL mode.

DIALOG-CONTROL = *FILE-CHANGE
For each file generation which is to be deleted, the user has the intervention options described under the first DIALOG-CONTROL operand. For each file generation which is to be processed, the user can decide in a dialog whether it should be deleted or not (response: YES/NO). If in the dialog he specifies protection attributes under “IGNORE”, or one or more passwords under “PASSWORD”, these will be taken into account for any selected file generation and, if satisfied, the file generation will be deleted without further queries (“YES” must also be specified). The user can also abort DELETE-FILE-GENERATION processing, or change the DIALOG-CONTROL mode.
The file generations which have been deleted will be listed in alphanumeric order.

DIALOG-CONTROL = *MORE-THAN-ONE-FILE
If exactly one file generation is specified, this will be deleted immediately. If GENERATION-NAME is partially qualified, which means that more than one file generation is selected, or if GENERATION-NAME contains wildcards, the user can decide, each time the catalog ID changes, whether or not file generations from the new catalog are to be deleted (see the intervention options described for the first of the DIALOG-CONTROL operands). He must respond to the question issued by the system with “YES” or “NO”. DIALOG-CONTROL = *MORE-THAN-ONE-FILE is useful if wildcards are specified for “catid” in the GENERATION-NAME operand. In the dialog, DELETE-FILE-GENERATION processing can be terminated, or the DIALOG-CONTROL mode can be changed.

DIALOG-CONTROL = *CATALOG-CHANGE
As for DIALOG-CONTROL = *MORE-THAN-ONE-FILE, the system starts a dialog if file generations in different catalogs (pubsets) are affected by DELETE-FILE-GENERATION processing. The user can determine whether the file generations in the current pubset should be deleted (YES/NO), DELETE-FILE-GENERATION processing should be terminated, or the DIALOG-CONTROL mode should be changed.

DIALOG-CONTROL = *USER-ID-CHANGE
Whenever the user ID is changed when deleting the file generations, a branch is made to guided dialog.

OUTPUT = <u>*STD</u> / *NO / *SYSOUT
The user can specify whether a message (`DMS0800`) with the name of the deleted file generation is to be output to SYSOUT for each successful deletion. The default setting *STD is equivalent to OUTPUT=*NO.

OUTPUT = *NO
No messages are output to SYSOUT for successfully deleted file generations.

OUTPUT = *SYSOUT
For each file generation that is successfully deleted, a message with the name of that file generation is output to SYSOUT.

IGNORE-PROTECTION = *NONE / list-poss(4): *ACCESS / *EXPIRATION-DATE / *WRITE-PASSWORD / *READ-PASSWORD
The user can specify whether any defined protection against write access or any defined retention period is to be ignored. Systems support staff can also ignore password protection. The specification IGNORE-PROTECTION in the DELETE-FILE-GENERATION command thus makes it unnecessary to issue MODIFY-FILE-GENERATION-ATTRIBUTES commands to reset the protection attributes before the file generations can be deleted.

IGNORE-PROTECTION = *NONE
The protection attributes “read-only” (ACCESS=READ) and “retention period” (EXPIRATION-DATE) will be observed during deletion.

IGNORE-PROTECTION = *ACCESS
File generations for which protection against writing is defined by ACCESS=READ can still be deleted using DELETE-FILE-GENERATION.

IGNORE-PROTECTION = EXPIRATION-DATE
File generations which are still within their retention period (EXDATE > current date) may nevertheless be deleted using DELETE-FILE-GENERATION.

IGNORE-PROTECTION = *WRITE-PASSWORD
System support staff are authorized to ignore the protection attribute **write password** when deleting the file generation.

IGNORE-PROTECTION = *READ-PASSWORD
System support staff are authorized to ignore the protection attribute **read password** when deleting the file generation.

PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> / <integer -2147483648..2147483647>
The user can specify one or more passwords, which will permit files protected by these passwords to be deleted. The passwords entered here are not recorded into the password table for the job, and are valid only for the current DELETE-FILE-GENERATION processing. Up to 3 passwords may be specified in the form of a list.
In order to delete a password-protected file, the password at the highest access level must be specified (see the ADD-PASSWORD command).
The operand has the following special characteristics:

- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .
- The password entered is not logged.

PASSWORDS-TO-IGNORE = *NONE
No passwords are specified.

SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
In procedures the user can specify whether the spin-off mechanism or SDF-P error handling is to be triggered every time an error occurs (apart from syntax errors), or whether specific error conditions are to be ignored.

SUPPRESS-ERRORS = *NONE
All errors will trigger the spin-off mechanism or SDF-P error handling.

SUPPRESS-ERRORS = list-poss(3): <alphanum-name 7..7>
The user can define which errors are to be ignored by means of their DMS error codes (alphanum-name 7..7). If the specified error occurs, the spin-off mechanism will not be triggered. A maximum of 3 error codes may be specified.
DMS error code: 7 characters, of which the first three are always “DMS”; the last 4 characters identify the error; the digits 0...9 and letters A...F are permitted. When error codes are entered, no check is made to verify that valid error codes have been specified.

A detailed list of valid DMS error codes can be found on the manual server (URL: *http://manuals.ts.fujitsu.com*) by means of an HTML application and on the “BS2000 SoftBooks” DVD.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed
1	0	CMD0001	No action required
2	0	DMS05F7	File generation does not exist, but group entry is updated
2	0	DMS06D6	Error on deleting certain files Guaranteed messages: DMS0800, DMS0801, DMS06D5, DMS0666, DMS05C6,DMS05BF, DMS05C3, DMS053F
	1	CMD0202	Syntactical or semantic error in command
	32	DMS0584	A state that does not allow the function to continue was reported during processing
	64	CMD0102	Interrupted by K2 key
	64	CMD0216	Privileges error
	64	DMS0501	Requested catalog not available
	64	DMS0512	Requested catalog not available
	64	DMS051B	Requested user ID not in pubset
	64	DMS051C	User not authorized to access pubset
	64	DMS0535	Specified file not shareable
	64	DMS055C	Catalog entry not found on specified private disk
	64	DMS057B	Invalid operand for migrated file
	64	DMS057C	Processing not possible due to HSMS error
	64	DMS057D	File has been migrated and cannot be recalled without delay
	64	DMS057E	File has been migrated, and HSMS is not available
	64	DMS0585	Error detected during catalog processing or multihost processing
	64	DMS0586	Currently not possible to access or reserve volume
	64	DMS0587	Use of specified command restricted by system administrator
	64	DMS05FC	Specified user ID not in HOME pubset
	64	DMS0609	Access to system file not permitted
	64	DMS06FF	BCAM connection severed
	130	DMS0524	System address space full
	130	DMS0582	File is currently locked or in use and cannot be processed

Notes

- If it is necessary to delete a file generation which is held on a private disk, then the device on which this disk is mounted will be requested for the job. After deletion of the file generation, the device will be returned to the system.
- For file generations stored on private disks, if DESTROY-ALL or MOUNT=*ALL-DISK is specified then all the volumes which hold the file generation must be already mounted at the time the command is executed. Otherwise, only the first volume which holds the file generation has to be mounted. The following volumes will not be requested until the command is being executed.

  This rule also applies if a partially qualified file generation name is specified in the DELETE-FILE-GENERATION command, thus addressing a number of files. In this situation, it is not necessary that all the volumes for all of the file generations are mounted simultaneously. The system will determine which of the file generations requires the most devices, and will request the corresponding number of devices.

Example

```
/show-file-attr max.group.4,select=(generation=*yes)  ————————————————————————————  (1)
%        0 :2OS2:$USER1.MAX.GROUP.4 (FGG)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0001)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0002)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0003)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0004)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0005)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0006)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0007)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0008)
%:2OS2: PUBLIC:      9 FILES RES=       24 FRE=       24 REL=       24 PAGES
/del-file-gen gen-name=max.group.4(*2),delete=*generation-before,output=*sysout
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.GROUP.4(*0001)' DELETED  —————————————  (2)
/del-file-gen gen-name=max.group.4(*6),delete=*generation-after,output=*sysout
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.GROUP.4(*0008)' DELETED  —————————————  (3)
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.GROUP.4(*0007)' DELETED
/del-file-gen gen-name=max.group.4(*2)  ——————————————————————————————————————————  (4)
%  CMD0051 INVALID OPERAND 'DELETE'
%  CMD0099 MANDATORY OPERAND INVALID OR MISSING       ————————————————————————————  (5)
/show-file-attr max.group.4,select=(generation=*yes)  ————————————————————————————  (6)
%        0 :2OS2:$USER1.MAX.GROUP.4 (FGG)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0002)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0003)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0004)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0005)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0006)
%:2OS2: PUBLIC:      6 FILES RES=       15 FRE=       15 REL=       15 PAGES
```

(1) The SHOW-FILE-ATTRIBUTES command returns information on the file generation group MAX.GROUP and all associated generations.

(2) The DELETE-FILE-GENERATION command is used to delete all generations that were created prior to generation *0002. The message sent to SYSOUT indicates that generation *0001 was deleted (see the operand OUTPUT=*SYSOUT).

(3) The DELETE-FILE-GENERATION command is used to delete all generations that were created after generation *0006. The message sent to SYSOUT indicates that generations *0007 and *0008 were deleted (see the operand OUTPUT=*SYSOUT).

(4) The DELETE-FILE-GENERATION command is issued for generation *0002 without a specification for the DELETE operand.

(5) The above command is rejected, since the DELETE operand is mandatory. The DELETE-FILE-GENERATION command is used to delete generations that were created before or after the specified generation.

(6) The SHOW-FILE-INFORMATION command returns information on the current catalog entries for the file generation group and for its existing generations.

DELETE-FILE-GROUP

Delete file generation group and generations

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE-GENERATION-GROUP
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE TSOS
Routing code:	$ (with NBCONOPI=N) or E (with NBCONOPI=Y)

Function

With the DELETE-FILE-GROUP command, a user can delete file generation groups, and their associated file generations, which are cataloged under his own user ID, and selected according to criteria which must be specified in the command. If the specified file is neither a file generation group nor a file generation, it will still be deleted. Specific deletion operations allow the user to release storage space. The user is assisted in doing this by a guided dialog. The operands of the DELETE-FILE-GROUP command can be divided into four groups, which correspond to the different functional levels of the command (see “Overview of functions” on page 3-108).

Selection

Using the selection parameters, the user specifies which file generation groups/catalog entries are to be processed. The attributes stored in the catalog entries are used as selection criteria. For this purpose, some of the operands in the DELETE-FILE-GROUP command correspond with those of the SHOW-FILE-ATTRIBUTES command.

File protection

File protection parameters make it unnecessary for the user to input a MODIFY-FILE-GROUP-ATTRIBUTES or ADD-PASSWORD command in order to delete files for which file protection attributes such as passwords, retention period, etc. are defined.

Action parameters

Action parameters control the internal execution of the DELETE-FILE-GROUP command. The user can, on the one hand, determine the scope of the deletions, and on the other hand can define conditions for deletion.

Control parameters

Control parameters permit the user to define his own user interface, within certain limits. For example, he can use interactive prompts in the dialog, or can request a SYSOUT log, or have DMS errors ignored (for the spin-off mechanism or SDF-P error handling).

Privileged functions

The privileged functions for system support staff include specification of any user ID, the ability to ignore protection attributes of the file generations being deleted and, in the case of a user ID change, to branch to guided dialog.

By default, systems support (TSOS privilege) is a co-owner of all the files (and can therefore delete file generation groups under any user ID). When SECOS is used, this co-ownership can be restricted.

In conjunction with the SECOS software product a user can allow other user IDs to act as co-owners of the file generation groups of his user ID. Co-owners of a user ID are then also allowed to delete file generation groups.

Overview of functions

	Function / Meaning	Level 1 operands	Level 2/3 operands
3-115	The name of the file generation group to be deleted	GROUP-NAME	
3-116	Criteria for selection of the file generations to be deleted	SELECT= *BY-ATTRIBUTES	
	Date specification		
	– Expiration date (implicit retention period) – Time		EXPIRATION-DATE TIME
	– Date when file is freed for deletion – Time		FREE-FOR-DELETION TIME
	Storage space		
	– Type of volume		SUPPORT
	– Public disk or private disk – Private disk		=*ANY =*PRIVATE-DISK

Table 47: Overview of DELETE-FILE-GROUP command functions (Part 1 of 3)

Function / Meaning	Level 1 operands	Level 2/3 operands
File security/file protection		
– Access		ACCESS
– Shareability		USER-ACCESS
– Basic access control list		BASIC-ACL
– Password protection		PASSWORD
– Access control using GUARDS		GUARDS
– protection against release of storage space		SPACE-RELEASE-LOCK
– Code table (CCS)		CODED-CHARACTER-SET
– BACKUP level		BACKUP-CLASS
– Ability to migrate from processing level (S0) to background level (S1 or S2) when the HSMS software product is in use		MIGRATE
– HSMS management class (MANAGEMENT-CLASS operand)		MANAGEMENT-CLASS
– Work file attribute		WORK-FILE-GROUP
3-116 Deletion parameters	OPTION	
– Delete the catalog entry and release storage space	=*ALL	
– Release storage space only, retain catalog entry	=*SPACE	
– Logical deletion: delete the data-related attributes of the file, change the catalog entry accordingly, retain the storage space allocation	=*DATA	
– Logical deletion as with *DATA, but the data-related attributes of the file are retained	=*DATA-KEEP-ATTRIBUTES	
– Delete the catalog entry, release the storage space and overwrite its contents	=*DESTROY-ALL	
3-117 Mount request	MOUNT	

Table 47: Overview of DELETE-FILE-GROUP command functions (Part 2 of 3)

	Function / Meaning	Level 1 operands	Level 2/3 operands
3-118	Control parameters	DIALOG-CONTROL	
	– User intervention not permitted (default setting for procedures and batch mode)	=*NO	
	– Dialog when more than one file is referenced if the file name specified is not fully qualified (default setting for interactive mode)	=*MORE-THAN-ONE-FILE	
	– Dialog when an error occurs	=*ERROR	
	– Dialog when catalog ID changes	=*CATALOG-CHANGE	
	– For each file selected, there will be an interactive dialog with the user to determine whether the current command should be processed	=*FILE-CHANGE	
	– Dialog when user ID changes	=*USER-ID-CHANGE	
3-119	Message for successfully deleted files	OUTPUT	
	– Suppress	=*NO	
	– Output to SYSOUT	=*SYSOUT	
3-120	Ignore protection attributes	IGNORE-PROTECTION	
	– Ignore write protection provided by ACCESS=READ	=*ACCESS	
	– Ignore retention periods	=*EXPIRATION-DATE	
	– Ignore password protection	=*WRITE-PASSWORD / *READ-PASSWORD	
3-121	Ignore password protection provided by specified passwords	PASSWORDS-TO-IGNORE	
3-121	Suppress error situations	SUPPRESS-ERRORS	

Table 47: Overview of DELETE-FILE-GROUP command functions (Part 3 of 3)

Format

(Part 1 of 5)

```
DELETE-FILE-GROUP                                                         Alias: DLFGP

 GROUP-NAME = *DUMMY / <filename 1..54 without-gen-vers with-wild(80)> /
,SELECT = *ALL / [*BY-ATTRIBUTES](...)
   [*BY-ATTRIBUTES](...)
        EXPIRATION-DATE = *ANY / *NONE / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
                          <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)
          *TOMORROW(...)
               TIME = *ANY / [*INTERVAL](...)
                  [*INTERVAL](...)
                      FROM = 00:00:00 / <time>
                     ,TO = 23:59:59 / <time>
          *TODAY(...)
               TIME = *ANY / [*INTERVAL](...)
                  [*INTERVAL](...)
                      FROM = 00:00:00 / <time>
                     ,TO = 23:59:59 / <time>
          *YESTERDAY(...)
               TIME = *ANY / [*INTERVAL](...)
                  [*INTERVAL](...)
                      FROM = 00:00:00 / <time>
                     ,TO = 23:59:59 / <time>
          <integer -99999..991231>(...)
               TIME = *ANY / [*INTERVAL](...)
                  [*INTERVAL](...)
                      FROM = 00:00:00 / <time>
                     ,TO = 23:59:59 / <time>
          <date>(...)
               TIME = *ANY / [*INTERVAL](...)
                  [*INTERVAL](...)
                      FROM = 00:00:00 / <time>
                     ,TO = 23:59:59 / <time>
```

(Part 2 of 5)

```
   *INTERVAL(...)
      FROM = *EARLIEST / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
                  <integer -99999..991231>(...) / <date>(...)
         *TOMORROW(...)
            TIME = 00:00:00 / <time>
         *TODAY(...)
            TIME = 00:00:00 / <time>
         *YESTERDAY(...)
            TIME = 00:00:00 / <time>
         <integer -99999..991231>(...)
            TIME = 00:00:00 / <time>
         <date>(...)
            TIME = 00:00:00 / <time>
     ,TO = *LATEST / TODAY(...) / *TOMORROW(...) / *YESTERDAY(...) /
              <integer -99999..991231>(...) / <date>(...) / *ANY
         *TODAY(...)
            TIME = 23:59:59 / <time>
         *TOMORROW(...)
            TIME = 23:59:59 / <time>
         *YESTERDAY(...)
            TIME = 23:59:59 / <time>
         <integer -99999..991231>(...)
            TIME = 23:59:59 / <time>
         <date>(...)
            TIME = 23:59:59 / <time>
,SUPPORT = *ANY / *PRIVATE-DISK
,ACCESS = *ANY / *READ / *WRITE
,PASSWORD = *ANY / list-poss(3): *NONE / *READ-PASSWORD / *WRITE-PASSWORD
,USER-ACCESS = *ANY / list-poss(2): *OWNER-ONLY / *ALL-USERS
,BACKUP-CLASS = *ANY / list-poss(5): *A / *B / *C / *D / *E
,MIGRATE = *ANY / list-poss(3): *ALLOWED / *INHIBITED / *FORBIDDEN
```

(Part 3 of 5)

```
,CODED-CHARACTER-SET = *ANY / *NONE / <name 1..8>

,SPACE-RELEASE-LOCK = *ANY / *NO / *YES

,BASIC-ACL = *ANY / *NONE / *YES / [*PARAMETERS](...)

   [*PARAMETERS](...)

        OWNER = *ANY / *NO-ACCESS / [*PARAMETERS](...)

           [*PARAMETERS](...)

                READ = *ANY / *NO / *YES
               ,WRITE = *ANY / *NO / *YES

       ,GROUP = *ANY / *NO-ACCESS / [*PARAMETERS](...)

           [*PARAMETERS](...)

                READ = *ANY / *NO / *YES
               ,WRITE = *ANY / *NO / *YES

       ,OTHERS = *ANY / *NO-ACCESS / [*PARAMETERS](...)

           [*PARAMETERS](...)

                READ = *ANY / *NO / *YES
               ,WRITE = *ANY / *NO / *YES

,GUARDS = *ANY / *YES / *NO / [*PARAMETERS](...)

   [*PARAMETERS](...)

        READ = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>
       ,WRITE = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>

,MANAGEMENT-CLASS = *ANY / *NONE / <composed-name 1..8>

,FREE-FOR-DELETION = *ANY / *NONE / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
                     <integer -99999..99999>(...) / <date>(...) / *INTERVAL(...)

   *TOMORROW(...)

        TIME = *ANY / *INTERVAL(...)

           *INTERVAL(...)

                FROM = 00:00:00 / <time>
               ,TO = 23:59:59 / <time>
```

(Part 4 of 5)

```
  |   *TODAY(...)
  |      |   TIME = *ANY / *INTERVAL(...)
  |      |
  |      |      *INTERVAL(...)
  |      |
  |      |         |   FROM = 00:00:00 / <time>
  |      |         |  ,TO = 23:59:59 / <time>
  |
  |   *YESTERDAY(...)
  |      |   TIME = *ANY / *INTERVAL(...)
  |      |
  |      |      *INTERVAL(...)
  |      |
  |      |         |   FROM = 00:00:00 / <time>
  |      |         |  ,TO = 23:59:59 / <time>
  |
  |   <integer -99999..99999>(...)
  |      |   TIME = *ANY / *INTERVAL(...)
  |      |
  |      |      *INTERVAL(...)
  |      |
  |      |         |   FROM = 00:00:00 / <time>
  |      |         |  ,TO = 23:59:59 / <time>
  |
  |   <date>(...)
  |      |   TIME = *ANY / *INTERVAL(...)
  |      |
  |      |      *INTERVAL(...)
  |      |
  |      |         |   FROM = 00:00:00 / <time>
  |      |         |  ,TO = 23:59:59 / <time>
  |
  |   *INTERVAL(...)
  |
  |      |   FROM = *EARLIEST / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
  |      |                <integer -99999..99999>(...) / <date>(...)
  |      |      *TOMORROW(...)
  |      |         |   TIME = 00:00:00 / <time>
  |      |
  |      |      *TODAY(...)
  |      |         |   TIME = 00:00:00 / <time>
  |      |
  |      |      *YESTERDAY(...)
  |      |         |   TIME = 00:00:00 / <time>
  |      |
  |      |      <integer -99999..99999>(...)
  |      |         |   TIME = 00:00:00 / <time>
  |      |
  |      |      <date>(...)
  |      |         |   TIME = 00:00:00 / <time>
```

(Part 5 of 5)

```
       |       | ,TO = *LATEST / *TODAY(...) / *TOMORROW(...) / *YESTERDAY(...) /
       |       |       <integer -99999..99999>(...) / <date>(...)
       |       |
       |       |    *TODAY(...)
       |       |       |  TIME = 23:59:59 / <time>
       |       |
       |       |    *TOMORROW(...)
       |       |       |  TIME = 23:59:59 / <time>
       |       |
       |       |    *YESTERDAY(...)
       |       |       |  TIME = 23:59:59 / <time>
       |       |
       |       |    <integer -99999..99999>(...)
       |       |       |  TIME = 23:59:59 / <time>
       |       |
       |       |    <date>(...)
       |       |       |  TIME = 23:59:59 / <time>
       |
       | ,WORK-FILE-GROUP = *ANY / *NO / *YES

,OPTION = *ALL / *SPACE / *DATA / *DATA-KEEP-ATTRIBUTES / *DESTROY-ALL

,MOUNT = *FIRST-DISK / *ALL-DISKS

,DIALOG-CONTROL = *STD / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE /
                  *CATALOG-CHANGE / *USER-ID-CHANGE

,OUTPUT = *STD / *NO / *SYSOUT

,IGNORE-PROTECTION = *NONE / list-poss(4): *ACCESS / *EXPIRATION-DATE / *WRITE-PASSWORD /
                     *READ-PASSWORD

,PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> /
                       <integer -2147483648..2147483647>

,SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
```

Operands

GROUP-NAME = *DUMMY / <filename 1..54 without-gen-vers with-wild(80)>

The name of the file generation group (FGG) which is to be deleted. If the specified file is not a file generation group (and also not a file generation), then it will still be deleted. Only the user’s own user ID or a user ID for which the user is co-owner may be specified.

GROUP-NAME = *DUMMY

Designates the dummy file generation group *DUMMY, which “always exists”, and which satisfies all selection criteria. Any other operands (except for DIALOG-CONTROL, OUTPUT and SUPPRESS-ERRORS) will be checked for their formal correctness, but otherwise ignored. If *DUMMY is specified, no catalog or data access is necessary. The main use of *DUMMY is in testing procedures.

SELECT = *ALL
All the files in the set specified by GROUP-NAME are selected.

SELECT = *BY-ATTRIBUTES(...)
Restricts the files in the set specified by GROUP-NAME to FGGs for which the fields in the group entry satisfy the criteria which follow. The default values, *ANY and ANY, each indicate that the file set is not to be restricted to any particular values for the attribute concerned.

For descriptions of the selection criteria see the DELETE-FILE-GENERATION command, which is used to delete file generations only.

OPTION = *ALL / *SPACE / *DATA / DATA-KEEP-ATTRIBUTES / *DESTROY-ALL
Controls the deletion of the file.
If `DESTROY-BY-DELETE=YES` was defined in the catalog entry of the file, the file or released storage space will *definitely* be overwritten with binary zeros. If this is not the case and if OPTION=DESTROY-ALL was also not specified, the file or released storage space will be deleted only logically, rather than being overwritten.

OPTION = *ALL
The default setting for disk files.
The group entry and the catalog entries for the selected file generations are to be deleted and their storage space will also be released.

OPTION = *SPACE
*Only for file generations on public disks (for tape files the default value *ALL applies):*
The storage space for the file generations deleted by DELETE-FILE-GROUP is released, but the catalog entry is retained, although amended: it is then identical with a catalog entry created by a CREATE-FILE-GROUP command.
For private disks the SPACE operand will be rejected.

OPTION = *DATA
*Only for disk files (for tape files the default value *ALL applies):*
The data in the selected files is “logically deleted”. After this, the user can no longer access the data, since he is no longer permitted to physically access the volume concerned. The group entry and storage space allocation still exist. The catalog entries for the file generation are identical to those for a file generation which was created using CREATE-FILE-GENERATION but which has not yet been opened (*FILE-STRUC=NONE*, *CRE-DATE=NONE*).

OPTION = *DATA-KEEP-ATTRIBUTES
*Only for disk files (for tape files the default value *ALL applies):*
The data of the file generations involved is “logically deleted” as with OPTION=*DATA, but the data-related file attributes are retained. The data can no longer be addressed by the user.

OPTION = *DESTROY-ALL
Only for disk files (for tape files the default value ALL applies): The storage space for the affected file generations is released and the group entry is deleted and the catalog entries of the file generation are deleted; in addition, the storage space thus released is overwritten with binary zeros so that, if the space is allocated again, nobody can read the old data (data protection). In the case of files on private disks, all the volumes on which the file was stored must be mounted at the time of deletion.
When a file is deleted, the action parameters are evaluated first. See the command EXPORT-FILE for details of exportation of file generations. “Data destruction” when a file is deleted can also be permanently recorded in the group entry for the file using the command CREATE-FILE-GROUP; in the group entry a “DESTROY” indicator is then set (DESTROY=YES). In this case storage space is released and it will be automatically overwritten.

MOUNT = <u>*FIRST-DISK</u> / *ALL-DISKS
Only for files on private disks: Specifies whether or not to mount all the private disks which hold any part of the file which is to be deleted. The user can request that only the first of the required private disks be mounted, or all of them. The MOUNT operand should be specified if the operands ALL or DESTROY-ALL are specified. For tape files, or files on public disks, any MOUNT specification will be ignored.

MOUNT = <u>*FIRST-DISK</u>
It is only necessary for the first of the private disks, containing the start of the file generation and its group entry, to be online. The associated private disks do not have to be online.

MOUNT = *ALL-DISKS
All the private disks on which parts of the file are held, must be online. If any of the disks is missing, the file will not be deleted; on completion of the DELETE-FILE-GROUP processing, the spin-off mechanism will be triggered.

DIALOG-CONTROL = *STD / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE / *CATALOG-CHANGE / *USER-ID-CHANGE
Specifies whether and under what conditions a verification dialog is to be conducted with the user during the deletion process. The verification dialog mechanism is available only in interactive mode, though that does include procedures. The only operand value that you can specify in batch mode is *STD or *NO.

The user can intervene with the following inputs:

- Y: the specified file or file set will then be deleted.
- N: the specified file or file set will not be deleted.
- T: processing of the command will be terminated.
- ?: the possible responses will be listed, with an explanation of each.

In addition, the following options can be specified, separated by commas:

- ,CHECK = NO The DIALOG-CONTROL mode will be changed to ‘*NO’.
- ,CHECK = PVS The DIALOG-CONTROL mode will be changed to ‘*CATALOG-CHANGE’.
- ,CHECK = MULTIPLE The DIALOG-CONTROL mode will be changed to ‘*MORE-THAN-ONE-FILE’.
- ,CHECK = SINGLE The DIALOG-CONTROL mode will be changed to ‘*FILE-CHANGE’.
- ,CHECK = ERROR The DIALOG-CONTROL mode will be changed to ‘*ERROR’.
- ,IGNORE = list-poss(4): ACCESS / EXDATE / RDPASS/ WRPASS Specifies which protection attributes are to be ignored during deletion. Only applies to a verification dialog for a single file generation group. RDPASS and WRPASS are available to privileged users only.
- ,PASSWORD = list-poss(3): <c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647> Enables password-protected files to be deleted (maximum of 3 passwords). Only applies to a verification dialog for a single file generation group.

DIALOG-CONTROL = *STD
The default value *STD corresponds to *MORE-THAN-ONE-FILE in interactive mode (SYSCMD is allocated to the terminal), and in procedures or in batch mode it corresponds to *NO.

DIALOG-CONTROL = *NO
The user cannot intervene in DELETE-FILE-GROUP processing; all the specified file generation groups will be deleted (without a verification dialog).

DIALOG-CONTROL = *ERROR
If deletion of the selected file generation groups proceeds without error, they will be deleted immediately, as when *NO is specified (i.e. no verification dialog). However, if a user-correctable error occurs, then a verification dialog takes place as for DIALOG-CONTROL = *FILE-CHANGE. DIALOG-CONTROL = *ERROR applies implicitly if DIALOG-CONTROL = *FILE-CHANGE is set. In the event of an error, the user may acknowledge the error message, abort processing of DELETE-FILE-GROUP or attempt to rectify the error. If he wishes, he can also change the DIALOG-CONTROL mode.

DIALOG-CONTROL = *FILE-CHANGE
For each file generation group which is to be deleted, the user has the intervention options described under the first DIALOG-CONTROL operand. For each file generation group which is to be processed, the user can decide in a verification dialog whether it should be deleted or not (response: YES/NO). If in the dialog he specifies protection attributes under “IGNORE”, or one or more passwords under “PASSWORD”, these will be taken into account for any selected file generation group and, if satisfied, the file generation group will be deleted without further queries (“YES” must also be specified). The user can also abort DELETE-FILE-GROUP processing, or change the DIALOG-CONTROL mode.
The file generations which have been deleted will be listed in alphanumeric order.

DIALOG-CONTROL = *MORE-THAN-ONE-FILE
If exactly one file generation group is specified, this will be deleted immediately. If GROUP-NAME is partially qualified, which means that more than one group is selected, or if GROUP-NAME contains wildcards, the user can decide, each time the catalog ID changes, whether or not file generation groups from the new catalog are to be deleted (see also the intervention options described under the first DIALOG-CONTROL operand). He must respond to the question issued by the system with “YES” or “NO”. DIALOG-CONTROL = *MORE-THAN-ONE-FILE is useful if wildcards are specified for the catalog ID in the GROUP-NAME. In the dialog, DELETE-FILE-GROUP processing can be terminated, or the DIALOG-CONTROL mode can be changed.

DIALOG-CONTROL = *CATALOG-CHANGE
As for DIALOG-CONTROL = *MORE-THAN-ONE-FILE, the system starts a dialog if file generation groups in different catalogs (pubsets) are affected by the DELETE-FILE-GROUP processing. The user can determine whether the file generation groups in the current pubset should be deleted (YES/NO), DELETE-FILE-GROUP processing should be terminated, or the DIALOG-CONTROL mode should be changed.

DIALOG-CONTROL = *USER-ID-CHANGE
Each time the user ID changes during the deletion of the file generation group, the system branches into the guided dialog.

OUTPUT = <u>*STD</u> / *NO / *SYSOUT
The user can specify whether a message (`DMS0800`) with the name of the deleted file generation group is to be output to SYSOUT for each successful deletion. The default setting *STD is equivalent to OUTPUT=*NO.

OUTPUT = *NO
No messages are output to SYSOUT for successfully deleted file generation groups.

OUTPUT = *SYSOUT
For each file generation group that is successfully deleted, a message with the name of that file generation group is output to SYSOUT.

IGNORE-PROTECTION = <u>*NONE</u> / list-poss(4): *ACCESS / *EXPIRATION-DATE / *WRITE-PASSWORD / *READ-PASSWORD
The user can specify whether any defined protection against write access or any defined retention period is to be ignored. Systems support staff can also ignore password protection. The specification of IGNORE-PROTECTION in the DELETE-FILE-GROUP command thus makes it unnecessary to issue MODIFY-FILE-GROUP-ATTRIBUTES commands to reset the protection attributes before the files can be deleted.

IGNORE-PROTECTION = <u>*NONE</u>
The protection attributes “read-only” (ACCESS=READ) and “retention period” (EXPIRATION-DATE) will be observed during deletion.

IGNORE-PROTECTION = *ACCESS
File generation groups for which protection against writing is defined by ACCESS=READ can still be deleted using DELETE-FILE-GROUP.

IGNORE-PROTECTION = *EXPIRATION-DATE
File generation groups which are still within their retention period (EXDATE > current date) may nevertheless be deleted using DELETE-FILE-GROUP.

IGNORE-PROTECTION = *WRITE-PASSWORD
Systems support staff are authorized to ignore the protection attribute **write password** when deleting the file generation group.

IGNORE-PROTECTION = *READ-PASSWORD
System support staff are authorized to ignore the protection attribute **read password** when deleting the file generation group.

PASSWORDS-TO-IGNORE = _*NONE_ / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> / <integer -2147483648..2147483647>
The user can specify one or more passwords, which will permit file generation groups protected by these passwords to be deleted. The passwords entered here are not recorded into the password table for the job, and are valid only for the current DELETE-FILE-GROUP processing. Up to 3 passwords may be specified in the form of a list. In order to delete a password-protected file generation group, the password at the highest access level must be specified (see the ADD-PASSWORD command).
The operand has the following special characteristics:

- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .
- The password entered is not logged.

PASSWORDS-TO-IGNORE = _*NONE_
No passwords are specified.

SUPPRESS-ERRORS = _*NONE_ / list-poss(3): <alphanum-name 7..7>
In procedures the user can specify whether the spin-off mechanism or SDF-P error handling is to be triggered every time an error occurs (apart from syntax errors), or whether specific error conditions are to be ignored.

SUPPRESS-ERRORS = _*NONE_
All errors will trigger the spin-off mechanism or SDF-P error handling.

SUPPRESS-ERRORS = list-poss(3): <alphanum-name 7..7>
The user can define which errors are to be ignored by means of their DMS error codes (alphanum-name 7..7). If the specified error occurs, the spin-off mechanism will not be triggered. A maximum of 3 error codes may be specified.
DMS error code: 7 characters, of which the first three are always “DMS”; the last 4 characters identify the error; the digits 0...9 and letters A...F are permitted. When error codes are entered, no check is made to verify that valid error codes have been specified.

A detailed list of valid DMS error codes can be found on the manual server (URL: *http://manuals.ts.fujitsu.com*) by means of an HTML application and on the “BS2000 SoftBooks” DVD.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed
1	0	CMD0001	No action required
2	0	DMS05F7	File generation does not exist, but group entry is updated
2	0	DMS06D6	Error on deleting certain files Guaranteed messages: DMS0800, DMS0801, DMS06D5, DMS0666, DMS05C6,DMS05BF, DMS05C3, DMS053F
	1	CMD0202	Syntactical or semantic error in command
	32	DMS0584	A state that does not allow the function to continue was reported during processing
	64	CMD0102	Interrupted by K2 key
	64	CMD0216	Privileges error
	64	DMS0501	Requested catalog not available
	64	DMS0512	Requested catalog not available
	64	DMS051B	Requested user ID not in pubset
	64	DMS051C	User not authorized to access pubset
	64	DMS0535	Specified file not shareable
	64	DMS055C	Catalog entry not found on specified private disk
	64	DMS057B	Invalid operand for migrated file
	64	DMS057C	Processing not possible due to HSMS error
	64	DMS057D	File has been migrated and cannot be recalled without delay
	64	DMS057E	File has been migrated, and HSMS is not available
	64	DMS0585	Error detected during catalog processing or multihost processing
	64	DMS0586	Currently not possible to access or reserve volume
	64	DMS0587	Use of specified command restricted by system administrator
	64	DMS05FC	Specified user ID not in HOME pubset
	64	DMS0609	Access to system file not permitted
	64	DMS06FF	BCAM connection severed
	130	DMS0524	System address space full
	130	DMS0582	File is currently locked or in use and cannot be processed
	130	DMS0585	Error detected during catalog processing or multihost processing
	130	DMS0586	Currently not possible to access or reserve volume
	130	DMS0594	Not enough virtual memory available

Notes

– If it is required to delete an FGG which is held on a private disk, then the device on which this disk is mounted will be requested for the job.

– For files stored on private disks, if OPTION=*DESTROY-ALL is specified then all the volumes which hold the file group must be already mounted at the time the command is executed or, if DATA is specified, only the first volume which holds the file group has to be mounted. This rule also applies if a partially qualified file name is specified in the DELETE-FILE-GROUP command, thus addressing a number of files. In this situation, it is not necessary that all the volumes for all of the files are mounted simultaneously. The system will determine which of the files requires the most devices, and will request the corresponding number of devices.

– If an error arises during the deletion of a file generation group or some of the generations in a group, then any remaining file generations which are present will not be deleted, and the catalog entry for the file generation group will be updated.

Example

```
/show-file-attr max.group.4,select=(generation=*yes)     ——————————————————————————  (1)
%        0 :2OS2:$USER1.MAX.GROUP.4 (FGG)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0002)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0003)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0004)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0005)
%        3 :2OS2:$USER1.MAX.GROUP.4(*0006)
%:2OS2: PUBLIC:      6 FILES RES=       15 FRE=       15 REL=       15 PAGES
/del-file-gr group-name=max.group.4,output=*sysout     ——————————————————————————————  (2)
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.GROUP.4(*0002)' DELETED  ——————————————  (3)
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.GROUP.4(*0003)' DELETED
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.GROUP.4(*0004)' DELETED
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.GROUP.4(*0005)' DELETED
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.GROUP.4(*0006)' DELETED
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.GROUP.4' DELETED
/show-file-attr max.group.4,select=(gen=*yes)   ———————————————————————————————————  (4)
%  DMS0533 REQUESTED FILE NOT CATALOGED IN PUBSET '2OS2'.COMMAND TERMINATED
```

(1) The SHOW-FILE-INFORMATION command returns information on the catalog entries for the file generation group *MAX.GROUP.4* and its existing generations.

(2) The DELETE-FILE-GROUP command is to be used to delete the file generation group, i.e. all generations as well as the group entry. A separate message is to be output to SYSOUT for each deleted catalog entry (operand OUTPUT=*SYSOUT).

(3) The messages sent to SYSOUT indicate the generations *0002 to *0006 and the group entry for the file generation group *MAX.GROUP.4* have been deleted.

(4) The SHOW-FILE-ATTRIBUTES command confirms that all catalog entries for the file generation group *MAX.GROUP.4* have been deleted.

DELETE-GS-PARTITION

Delete partition from GS

Description status:	GSMAN V19.0A
Functional area:	Caching media control Global storage administration
Domain:	DEVICE
Privileges:	TSOS

Function

The DELETE-GS-PARTITION command enables systems support to delete a partition from the global storage (GS) medium. GS is a high-capacity non-volatile storage medium used for buffer storage of data (see also “The global storage (GS) medium” in the “Introduction to System Administration” [14]).

In global GS operation in an XCS network the command applies to all the systems on the network. The GSMAN subsystem is not available until XCS has been started.

In order for a partition to be successfully deleted, use of it must have ended and it must no longer contain any valid data. Otherwise the command will be rejected and the partition retained. In this case it is possible to force the partition to be deleted by using the FORCE-DESTROY-GS-PARTITION command.

Systems support can use the SHOW-GS-STATUS command to check up on the current GS configuration.

Format

DELETE**-GS-PART**ITION
PARTITION**-ID** = <name 1..8>

Operands

PARTITION-ID = <name 1..8>
Name of the GS partition to be deleted from GS. If there is no partition with this name, the command will be rejected and a message to that effect will be issued.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
	1	CMD0202	Syntax error
	32	CMD0221	Internal error
	64	CMD0216	No authorization to invoke command
	64	EGC0112	No GS available
	64	EGC0301	The specified partition does not exist
	64	EGC0302	The partition is still in use
	64	EGC0303	The partition still contains data
	64	EGC1000	GS is not available
	128	EGC0010	GSMAN subsystem is not ready
	128	EGC0110	Command temporarily not executable

DELETE-GS-VOLUME

Delete GS volume

Description status:	GSVOL V1.3B
Functional area:	Caching media control Global storage administration
Domain:	DEVICE SYSTEM-TUNING STORAGE-MANAGEMENT
Privileges:	TSOS

Function

The DELETE-GS-VOLUME command allows systems support to delete a GS volume. The GS partition on which the GS volume was set up and the device mnemonic of the GS volume are available for subsequent reuse (CREATE-GS-VOLUME command).
If necessary, the GS partition can be removed using a suitable GSMAN command.

In the case of global GS operation within an XCS network, the command applies equally to all the systems within the network. That means that it needs to be issued just once for each GS volume that is to be deleted.

The SHOW-GS-VOLUME-ATTRIBUTES command allows systems support to find out about available GS volumes.

Format

```
DELETE-GS-VOLUME

 DEVICE-UNIT = <x-text 4..4>
,FORCE = *NO / *YES
```

Operands

DEVICE-UNIT = <x-text 4..4>
Specifies the device mnemonic of the GS volume being deleted.

FORCE = *NO / *YES
Determines whether the GS volume is to be deleted unconditionally.

FORCE = *NO
The GS volume is to be deleted on condition that there are no more systems attached to it. If this is not the case, the command is rejected.

FORCE = *YES
The GS volume is to be deleted whatever the circumstances, even if there are still systems attached to it.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed
	1	CMD0202	Syntax error in command
	32	CMD0221	Internal error
	64	CMD0216	No authorization to issue command
	64	NDG0003	Volume cannot be deleted
	64	NDG0005	Device mnemonic not in allowable range 0300 - 03FF
	64	NDG0006	Error accessing partition
	64	NDG0007	Volume does not exist
	64	NDG0101	Invalid response
	64	NDG0103	Command execution was cancelled by the caller

Example

```
/DELETE-GS-VOLUME DEVICE-UNIT=0314
```

DELETE-ISAM-POOL

Delete ISAM pool or link to ISAM pool

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING

Function

The DELETE-ISAM-POOL command enables users to delete task-specific ISAM pools which they have created or to cancel links between their jobs and shareable ISAM pools (with scopes of USER-ID, USER-GROUP and HOST-SYSTEM). If the link between a shareable ISAM pool and the last job to which it was connected is canceled, the ISAM pool is automatically deleted.

An ISAM pool is uniquely identified by the specified name, the catalog ID (see the CAT-ID operand) and the scope (see the SCOPE operand). If an ISAM pool named in the command does not exist, the command is rejected with an error message. The user can check the ISAM pools to which his or her job is linked by using the SHOW-ISAM-POOL-ATTRIBUTES command.

Before it is possible to delete an ISAM pool, or the link between a job and a pool, the command REMOVE-ISAM-POOL-LINK must be issued to delete all entries for the pool concerned from the pool table of the job. Otherwise, if a pool is still linked to the pool table by its pool link name, the DELETE-ISAM-POOL command will be rejected with an error message. Entries in the pool table can be displayed by means of the SHOW-ISAM-POOL-LINK command.

Note on ISAM pools

As of BS2000/OSD V6.0B cross-task ISAM pools are automatically created on a file-specific basis when a file is opened in a data space. The CREATE-ISAM-POOL command with SCOPE=*USER-ID/*USER-GROUP is only supported for reasons of compatibility and has the same effect as SCOPE=*HOST-SYSTEM (for details on ISAM pools in data spaces see the “Introductory Guide to DMS” [13]).

A detailed description of the ISAM pool can be found in the manuals “Introductory Guide to DMS” [13] and “DMS Macros” [12].

Format

```
DELETE-ISAM-POOL

POOL-NAME = *ALL / <name 1..8>(...)

   <name 1..8>(...)

      |  CAT-ID = *DEFAULT-PUBSET / <cat-id 1..4>

      | ,SCOPE = *TASK / *HOST-SYSTEM / *USER-ID / *USER-GROUP
```

Operands

POOL-NAME =*ALL / <name 1..8>(...)
Specifies whether it is only particular ISAM pools, or all those which are linked to the job, which are to be released.

POOL-NAME = *ALL
All the ISAM pools which are linked to the calling job are to be deleted or – for shareable pools – all the links to the ISAM pool are to be canceled. If the pool link names still define links between pools and files for the calling job, the command will be rejected with an error message.

POOL-NAME = <name 1..8>(...)
The specified ISAM pool is to be deleted. The ISAM pool is uniquely identified by the name, the catalog ID (see the CAT-ID operand) and the scope (see the SCOPE operand). If no ISAM pool was created as specified or no link to the pool was established (see the CREATE-ISAM-POOL command), the command is rejected with an error message.

CAT-ID = *DEFAULT-PUBSET / <alphanum-name 1..4>
Catalog ID of the pubset to which the ISAM pool is assigned.

CAT-ID = *DEFAULT-PUBSET
The ISAM pool is assigned to the catalog that was set with the `ISPLDEFC` system parameter (**IS**AM-**P**OO**L**-**DEF**AULT-**C**ATID):

X'00': default catalog ID from the user entry (see the SHOW-USER-ATTRIBUTES command, *DEFAULT-PUBSET* output field*)*

X'01': catalog ID of the home pubset

CAT-ID = <alphanum-name 1..4>
Catalog ID of the specified ISAM pool.

SCOPE = *TASK / *HOST-SYSTEM / *USER-ID / *USER-GROUP
Scope of the specified ISAM pool.

SCOPE = *TASK
The corresponding task-local ISAM pool, which can only be used by its own task, is to be deleted.

SCOPE = *HOST-SYSTEM
The corresponding cross-task ISAM pool, which can be used by all tasks, is to be deleted.

SCOPE = *USER-ID / USER-GROUP
These scopes are only supported for reasons of compatibility (see the "Note on ISAM pools" on page 3-128).

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without errors
	32	DMS0A17	Internal system error
	64	DMS0A0E	Syntax error in ISAM pool command
	64	DMS0A11	Specified catalog ID does not exist
	64	DMS0A13	Specified pool name is syntactically invalid
	64	DMS0A19	Specified ISAM pool does not exist
	64	DMS0A1A	Pool links to ISAM pool still exist
	64	DMS0A21	ISAM pool limit reached
	64	DMS0A22	User group does not exist
	130	DMS0A12	Specified catalog ID not available

Examples

See the ADD-ISAM-POOL-LINK command.

DELETE-JV

Delete job variable

Description status:	JV V15.1A
Functional area:	Job variables
Domain:	JOB-VARIABLES
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION TSOS
Routing code:	$ (with NBCONOPI=N) or J (with NBCONOPI=Y)

This function is only available to the user if the chargeable software product JV has been loaded as a subsystem.

Function

The DELETE-JV command is used to delete job variables cataloged under the user's own ID. Job variables that monitor a job (SET-LOGON-PARAMETERS, ENTER-JOB, ENTER-PROCEDURE) cannot be deleted until the job is completed (status indicator: $T or $A) or unless job monitoring is prematurely terminated with the MODIFY-JV-ATTRIBUTES command (MONJV-PROTECTION operand).

Job variables that are used in conditional job control (CJC) commands and macros are likewise protected against deletion (information can be obtained with the SHOW-CJC-STATUS command), i.e. a condition formulated in the ADD-CJC-ACTION or WAIT-EVENT commands or in the ONEVT macro continues to apply.

Privileged functions

The privileged functions for systems support (TSOS privilege) include specification of temporary or cataloged files of any user ID, the ability to ignore protection attributes of the job variable being deleted and, in the case of a user ID change, to branch to guided dialog.

By default, systems support (TSOS privilege) is a co-owner of all the job variables (and can therefore delete job variables under any user ID). When SECOS is used, this co-ownership can be restricted for permanent job variables.

Format

(Part 1 of 5)

```
DELETE-JV                                                                                    Alias: DLJV

JV-NAME = <filename 1..54 without-gen-vers with-wild(80)> / *LINK(...)

  *LINK(...)
    |
    |  LINK-NAME = <alphanum-name 1..7>

,IGNORE-PROTECTION = *NONE / list-poss(4): *ACCESS  / *EXPIRATION-DATE /
                                    *READ-PASSWORD / *WRITE-PASSWORD

,PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <c-string 1..4> / <x-string 1..8> /
                                        <integer -2147483648..2147483647>

,DIALOG-CONTROL = *STD / *JV-CHANGE / *USER-ID-CHANGE / *CATALOG-CHANGE / *NO

,OPTION = *ALL / *DATA

,OUTPUT = *NO / *SYSOUT

,SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>

,SELECT = *ALL / [*BY-ATTRIBUTES](...)

  [*BY-ATTRIBUTES](...)
    |
    |  ACCESS = *ANY / *READ / *WRITE
    |
    | ,USER-ACCESS = *ANY / list-poss(2): *OWNER-ONLY / *ALL-USERS
    |
    | ,PASSWORD = *ANY / list-poss(3): *NONE / *READ-PASSWORD / *WRITE-PASSWORD
    |
    | ,CREATION-DATE = *ANY / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) /
    |                        <date>(...) / *INTERVAL(...)
    |
    |    *TODAY(...)
    |      |
    |      |  TIME = *ANY / [*INTERVAL](...)
    |      |
    |      |    [*INTERVAL](...)
    |      |      |
    |      |      |  FROM = 00:00:00 / <time>
    |      |      | ,TO = 23:59:59 / <time>
    |
    |    *YESTERDAY(...)
    |      |
    |      |  TIME = *ANY / [*INTERVAL](...)
    |      |
    |      |    [*INTERVAL](...)
    |      |      |
    |      |      |  FROM = 00:00:00 / <time>
    |      |      | ,TO = 23:59:59 / <time>
```

(Part 2 of 5)

```
       |     <integer -99999..991231>(...)
       |        |   TIME = *ANY / [*INTERVAL](...)
       |        |
       |        |      [*INTERVAL](...)
       |        |         |   FROM = 00:00:00 / <time>
       |        |         |  ,TO = 23:59:59 / <time>
       |
       |     <date>(...)
       |        |   TIME = *ANY / [*INTERVAL](...)
       |        |
       |        |      [*INTERVAL](...)
       |        |         |   FROM = 00:00:00 / <time>
       |        |         |  ,TO = 23:59:59 / <time>
       |
       |     *INTERVAL(...)
       |
       |        |   FROM = 1950-01-01 / <integer -99999..991231>(...) / <date>(...) / *TODAY(...) /
       |        |                  *YESTERDAY(...)
       |        |
       |        |      <integer -99999..991231>(...)
       |        |         |   TIME = 00:00:00 / <time>
       |        |
       |        |      <date>(...)
       |        |         |   TIME = 00:00:00 / <time>
       |        |
       |        |      *TODAY(...)
       |        |         |   TIME = 00:00:00 / <time>
       |        |
       |        |      *YESTERDAY(...)
       |        |         |   TIME = 00:00:00 / <time>
       |        |
       |        |  ,TO = *TODAY (...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)
       |        |
       |        |      *TODAY(...)
       |        |         |   TIME = 23:59:59 / <time>
       |        |
       |        |      *YESTERDAY(...)
       |        |         |   TIME = 23:59:59 / <time>
       |        |
       |        |      <integer -99999..991231>(...)
       |        |         |   TIME = 23:59:59 / <time>
       |        |
       |        |      <date>(...)
       |        |         |   TIME = 23:59:59 / <time>
```

(Part 3 of 5)

```
   | ,EXPIRATION-DATE = *ANY / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) /
   |                        <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)
   |
   |    *TOMORROW(...)
   |     |   TIME = *ANY / [*INTERVAL](...)
   |     |
   |     |      [*INTERVAL](...)
   |     |       |   FROM = 00:00:00 / <time>
   |     |       |  ,TO = 23:59:59 / <time>
   |
   |    *TODAY(...)
   |     |   TIME = *ANY / [*INTERVAL](...)
   |     |
   |     |      [*INTERVAL](...)
   |     |       |   FROM = 00:00:00 / <time>
   |     |       |  ,TO = 23:59:59 / <time>
   |
   |    *YESTERDAY(...)
   |     |   TIME = *ANY / [*INTERVAL](...)
   |     |
   |     |      [*INTERVAL](...)
   |     |       |   FROM = 00:00:00 / <time>
   |     |       |  ,TO = 23:59:59 / <time>
   |
   |    <integer -99999..991231>(...)
   |     |   TIME = *ANY / [*INTERVAL](...)
   |     |
   |     |      [*INTERVAL](...)
   |     |       |   FROM = 00:00:00 / <time>
   |     |       |  ,TO = 23:59:59 / <time>
   |
   |    <date>(...)
   |     |   TIME = *ANY / [*INTERVAL](...)
   |     |
   |     |      [*INTERVAL](...)
   |     |       |   FROM = 00:00:00 / <time>
   |     |       |  ,TO = 23:59:59 / <time>
   |
   |    *INTERVAL(...)
   |     |   FROM = 1950-01-01 / <integer -99999..991231>(...) / *TOMORROW(...) / *TODAY(...) /
   |     |               *YESTERDAY(...) / <date>(...)
   |     |
   |     |      <integer -99999..991231>(...)
   |     |       |   TIME = 00:00:00 / <time>
   |     |
   |     |      *TOMORROW(...)
   |     |       |   TIME = 00:00:00 / <time>
   |     |
   |     |      *TODAY(...)
   |     |       |   TIME = 00:00:00 / <time>
```

(Part 4 of 5)

```
                  *YESTERDAY(...)
                       TIME = 00:00:00 / <time>

                  <date>(...)
                       TIME = 00:00:00 / <time>

               ,TO = *ANY / TODAY(...) / *TOMORROW(...) / *YESTERDAY(...) /
                     <integer -99999..991231>(...) / <date>(...) / *ANY

                  *TODAY(...)
                       TIME = 23:59:59 / <time>

                  *TOMORROW(...)
                       TIME = 23:59:59 / <time>

                  *YESTERDAY(...)
                       TIME = 23:59:59 / <time>

                  <integer -99999..991231>(...)
                       TIME = 23:59:59 / <time>

                  <date>(...)
                       TIME = 23:59:59 / <time>

       ,BASIC-ACL = *ANY / *NONE / *YES / [*PARAMETERS](...)

          [*PARAMETERS](...)

               OWNER = *ANY / *NO-ACCESS / [*PARAMETERS](...)

                  [*PARAMETERS](...)
                       READ = *ANY / *NO / *YES
                      ,WRITE = *ANY / *NO / *YES

              ,GROUP = *ANY / *NO-ACCESS / [*PARAMETERS](...)

                  [*PARAMETERS](...)
                       READ = *ANY / *NO / *YES
                      ,WRITE = *ANY / *NO / *YES

              ,OTHERS = *ANY / *NO-ACCESS / [*PARAMETERS](...)

                  [*PARAMETERS](...)
                       READ = *ANY / *NO / *YES
                      ,WRITE = *ANY / *NO / *YES

       ,GUARDS = *ANY / *YES / *NONE / [*PARAMETERS](...)

          [*PARAMETERS](...)

               READ = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>

              ,WRITE = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>
```

(Part 5 of 5)

```
   ,MANAGEMENT-CLASS = *ANY / *NONE / <composed-name 1..8>
   ,PROTECTION-ACTIVE = *ANY / list-poss(3): *LEVEL-0 / *LEVEL-1 / *LEVEL-2
   ,SIZE = *ANY / <integer 0..256> / [*INTERVAL](...)
      [*INTERVAL](...)
           FROM = 0 / <integer 0..256>
          ,TO = 256 / <integer 0..256>
```

Operands

JV-NAME = <filename 1..54 without-gen-vers with-wild(80)> / *LINK(...)
Specification identifying the job variable to be deleted. JVs to be deleted can be identified by name or link name.
If the JV is protected by a password, the password must be specified (in the ADD-PASSWORD command). The password with the higher protection function should always be specified (write before read password).

JV-NAME = <filename 1..54 without-gen-vers with-wild(80)>
Name of the JV to be deleted. Nonprivileged users may only specify their own user ID without using wildcards. If no catalog ID is specified, the default catalog ID of the user ID concerned is selected. If the first character in the string is “*”, then it has to be entered twice.

JV-NAME = *LINK(...)
The JV to be deleted is identified by a link name.

> **LINK-NAME = <alphanum-name 1..7>**
> Link name of the JV to be deleted.
> The JV to be deleted must be cataloged under the user’s own ID.

IGNORE-PROTECTION = *NONE / list-poss(4): *ACCESS / *EXPIRATION-DATE / *READ-PASSWORD / *WRITE-PASSWORD
Specifies which protection attributes are to be ignored during deletion of a JV. The *ACCESS entry also applies for BASIC-ACL and GUARDS protection.
*Only systems support (TSOS privilege) can delete JVs by entering the operand values *READ-PASSWORD and *WRITE-PASSWORD without entering the password.*

PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> / <integer -2147483648..2147483647>
Allows the deletion of job variables that are protected by the specified passwords (write before read password). No entry is made in the password table.
The operand has the following special characteristics:

- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

– The password entered is not logged.

DIALOG-CONTROL =
Specifies whether and under what conditions a verification dialog is to be conducted with the user during the deletion process. Verification is advisable if the job variable name is partially qualified or specified with wildcards, thus enabling more than one job variable to be deleted (also unintentionally).
The verification dialog mechanism is available only in interactive mode, though that does include procedures.
Each verification query can be answered with `response[,CHECK=mode]`.

Answer	**Effects**
Y	The JV or JV set displayed in the query is to be deleted.
N	The JV or JV set displayed is not to be deleted.
T	The verification dialog is aborted. The job variables concerned are not deleted.
Other values	Same effect as "N"
Optional suffix:	**Effects**
,CHECK=mode	CHECK enables the setting for the verification dialog to be changed (corresponds to DIALOG-CONTROL). The following values can be specified for <mode>: SINGLE : corresponds to DIALOG-CONTROL=*JV-CHANGE MULTIPLE : corresponds to DIALOG-CONTROL=*USER-ID-CHANGE PVS : corresponds to DIALOG-CONTROL=*CATALOG-CHANGE NO : corresponds to DIALOG-CONTROL=*NO STD : corresponds to DIALOG-CONTROL=*STD

DIALOG-CONTROL = <u>*STD</u>
Default value: corresponds to *USER-ID-CHANGE in true interactive mode and to *NO in procedures and in batch mode.

DIALOG-CONTROL = *NO
All selected job variables are deleted without a verification dialog.

DIALOG-CONTROL = *JV-CHANGE
A verification query occurs for each job variable to be deleted (message JVS0469).

DIALOG-CONTROL = *USER-ID-CHANGE
A verification query occurs in each case for all job variables under one user ID that are to be deleted (message JVS0465). The query does not occur if only one job variable is to be deleted.

DIALOG-CONTROL = *CATALOG-CHANGE
A verification query occurs in each case for all job variables of a pubset that are to be deleted (message JVS0468). The query does not occur if only one job variable is to be deleted.

OPTION = *ALL / *DATA
Specifies whether the job variable entry or only the contents are to be deleted.

OUTPUT = *NO / *SYSOUT
Specifies whether the names of the deleted job variables are to be output.

SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
In procedures the user can specify whether the spin-off mechanism or SDF-P error handling is to be triggered every time an error occurs (apart from syntax errors), or whether specific error conditions are to be ignored.

SUPPRESS-ERRORS = *NONE
All errors will trigger the spin-off mechanism or SDF-P error handling.

SUPPRESS-ERRORS = list-poss(3): <alphanum-name 7..7>
The user can define which errors are to be ignored by means of their DMS error codes (alphanum-name 7..7). If the specified error occurs, the spin-off mechanism will not be triggered. A maximum of 3 error codes may be specified. DMS error code: 7 characters, of which the first three are always “DMS”; the last 4 characters identify the error; the digits 0...9 and letters A...F are permitted. When error codes are entered, no check is made to verify that valid error codes have been specified.

A detailed list of valid JVS error codes can be found in the “JV” manual [20] or on the manual server (URL: *http://manuals.ts.fujitsu.com*) by means of an HTML application and on the “BS2000 SoftBooks” DVD.

SELECT = *ALL / *BY-ATTRIBUTES(...)
The job variable selection criteria. The following selection criteria are not relevant for displaying special job variables.

SELECT = *ALL
Deletes all the job variables which the user is authorized to access.

SELECT = *BY-ATTRIBUTES(...)
Restricts the job variables selected from the set specified by JV-NAME to those which satisfy the following specifications. The default value *ANY for an attribute means that the job variable set is not to be restricted to particular values of that attribute.

ACCESS = *ANY / *READ / *WRITE
Deletes job variables depending on the permitted access type.

ACCESS = *READ
Deletes only those job variables for which write access is prohibited by ACCESS=READ, i.e. to which only read access is permitted.

ACCESS = *WRITE
Deletes only those job variables for which write access is permitted.

USER-ACCESS = *ANY / list-poss(2): *OWNER-ONLY / *ALL-USERS
Deletes job variables depending on whether or not they are shareable. If a user ID other than the user’s own is specified, then implicitly USER-ACCESS=ALL-USERS.

USER-ACCESS = *OWNER-ONLY
Deletes job variables which only the owner or co-owner may access.

USER-ACCESS = *ALL-USERS
Deletes job variables which may also be accessed under other user IDs.

PASSWORD = *ANY / list-poss(3): *NONE / *READ-PASSWORD / *WRITE-PASSWORD
Deletes job variables depending on the password protection defined. If several types of password are specified as a list, the system will link them by logical Or, and will return information on all the job variables which satisfy any one of the specified conditions.

PASSWORD = *NONE
Deletes job variables which have no password protection.

PASSWORD = *READ-PASSWORD
Deletes job variables which are protected by a read password.

PASSWORD = *WRITE-PASSWORD
Deletes job variables which are protected by a write password

CREATION-DATE = *ANY / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)

Deletes job variables on the basis of their creation date (CREATION-DATE); range specifications are inclusive of the specified limit values. It is meaningless to specify a CREATION-DATE which lies in the future!

CREATION-DATE = *TODAY(...)

Deletes job variables which have today’s date entered as the CREATION-DATE in their catalog entry.

TIME = *ANY / *INTERVAL(...)

Restricts the selection of job variables to a time interval related to the specified creation date.

TIME = *INTERVAL(...)

Deletes job variables that were created on the specified day within the specified time interval which follows.

FROM = 00:00:00 / <time>

Deletes job variables for which the time of creation ≥ the specified time.

TO = 23:59:59 / <time>

Deletes job variables for which the time of creation ≥ the specified time.

CREATION-DATE = *YESTERDAY(...)

Deletes job variables which have yesterday’s date entered as the CREATION-DATE in their catalog entry.

TIME = *ANY / *INTERVAL(...)

Restricts the selection of job variables to a time interval related to the specified creation date.

TIME = *INTERVAL(...)

Deletes job variables that were created on the specified day within the specified time interval which follows.

FROM = 00:00:00 / <time>

Deletes job variables for which the time of creation ≥ the specified time.

TO = 23:59:59 / <time>

Deletes job variables for which the time of creation ≥ the specified time.

CREATION-DATE = <integer -99999..991231>(...)
Deletes job variables which were created on the specified date. Here, the user can specify the creation date in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value:* (6 digits, with preceding sign) the number of days from today's date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1 or TODAY ≙ ±0)

TIME = <u>*ANY</u> / *INTERVAL(...)
Restricts the selection of job variables to a time interval related to the specified creation date.

TIME = *INTERVAL(...)
Deletes job variables that were created on the specified day within the specified time interval which follows.

FROM = <u>00:00:00</u> / <time>
Deletes job variables for which the time of creation ≥ the specified time.

TO = <u>23:59:59</u> / <time>
Deletes job variables for which the time of creation ≥ the specified time.

CREATION-DATE = <date>(...)
Deletes job variables which were created on the specified date.
The user can specify the creation date in the form [yy]yy-mm-dd.

TIME = <u>*ANY</u> / *INTERVAL(...)
Restricts the selection of job variables to a time interval related to the specified creation date.

TIME = *INTERVAL(...)
Deletes job variables that were created on the specified day within the specified time interval which follows.

FROM = <u>00:00:00</u> / <time>
Deletes job variables for which the time of creation ≥ the specified time.

TO = <u>23:59:59</u> / <time>
Deletes job variables for which the time of creation ≥ the specified time.

CREATION-DATE = *INTERVAL(...)
Deletes job variables which were created within the specified time period. The upper and lower limits are both included in the range specified. See also the explanation of how dates are specified in the EXPIRATION-DATE=<integer ...> operand. It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). Whichever of the operands is not specified will be replaced by the default value for use as the limit of the range. The use of range limits for information output can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = 1950-01-01 / <integer -99999..991231>(...) / *TODAY(...) / *YESTERDAY(...) / <date>(...)
Deletes job variables which were created after the specified date (i.e. CREATION-DATE ≥ specified date).

FROM = <integer -99999..991231>(...)
Deletes job variables which were created after the specified date (i.e. CREATION-DATE ≥ specified date).

TIME = 00:00:00 / <time>
Time on the specified date. All job variables created at or after the specified time are selected.

FROM = *TODAY(...)
Deletes job variables which were created after the specified date (i.e. CREATION-DATE ≥ current date).

TIME = 00:00:00 / <time>
Time on the specified date. All job variables created at or after the specified time are selected.

FROM = *YESTERDAY(...)
Deletes job variables which were created after the specified date (i.e. CREATION-DATE ≥ date of preceding day).

TIME = 00:00:00 / <time>
Time on the specified date. All job variables created at or after the specified time are selected.

FROM = <date>(...)
Deletes job variables which were created after the specified date (i.e. CREATION-DATE ≥ specified date).

TIME = 00:00:00 / <time>
Time on the specified date. All job variables created at or after the specified time are selected.

TO = *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)
Deletes job variables which were created before the specified date (i.e. CREATION-DATE ≤ specified date).

TO = *TODAY(...)
Deletes job variables which were created before the specified date (i.e. CREATION-DATE ≤ current date).

TIME = 23:59:59 / <time>
Time on the specified date. All job variables created at or before the specified time are selected.

TO = *YESTERDAY(...)
Deletes job variables which were created before the specified date (i.e. CREATION-DATE ≤ date of preceding day).

TIME = 23:59:59 / <time>
Time on the specified date. All job variables created at or before the specified time are selected.

TO = <integer -99999..991231>(...)
Deletes job variables which were created before the specified date (i.e. CREATION-DATE ≤ specified date).

TIME = 23:59:59 / <time>
Time on the specified date. All job variables created at or before the specified time are selected.

TO = <date>(...)
Deletes job variables which were created before the specified date (i.e. CREATION-DATE ≤ specified date).

TIME = 23:59:59 / <time>
Time on the specified date. All job variables created at or before the specified time are selected.

EXPIRATION-DATE = *ANY / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...) / *INTERVAL(...)
The EXPIRATION-DATE operand deletes job variables as a function of the “expiration date” (*EXPIR-DATE* output field) , i.e. the date from which write accesses to the job variable are allowed. It is meaningful to specify a date in the future if retention periods are being queried.

EXPIRATION-DATE = *TOMORROW(...)
Deletes job variables which have tomorrow’s date entered as the EXPIRATION-DATE in their catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of job variables to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes job variables for which the expiration date lies within the specified interval that follows.

FROM = 00:00:00 / <time>
Deletes job variables for which the time of expiration ≥ the specified time.

TO = 23:59:59 / <time>
Deletes job variables for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = *TODAY(...)
Deletes only job variables which have today’s date entered as the EXPIRATION-DATE in their catalog entry.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of job variables to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes job variables for which the expiration date lies within the specified interval that follows.

FROM = 00:00:00 / <time>
Deletes job variables for which the time of expiration ≥ the specified time.

TO = 23:59:59 / <time>
Deletes job variables for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = *YESTERDAY(...)
Deletes only job variables which have yesterday’s date entered as the EXPIRATION-DATE in their catalog entry.

TIME = <u>*ANY</u> / *INTERVAL(...)
Restricts the selection of job variables to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes job variables for which the expiration date lies within the specified interval that follows.

FROM = <u>00:00:00</u> / <time>
Deletes job variables for which the time of expiration ≥ the specified time.

TO = <u>23:59:59</u> / <time>
Deletes job variables for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = <integer -99999..991231>(...)
Here, the user can specify the expiration date in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value:* (6 digits with preceding sign), the number of days from today’s date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ ?00 or TOMORROW ï +1)

TIME = <u>*ANY</u> / *INTERVAL(...)
Restricts the selection of job variables to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes job variables for which the expiration date lies within the specified interval that follows.

FROM = <u>00:00:00</u> / <time>
Deletes job variables for which the time of expiration ≥ the specified time.

TO = <u>23:59:59</u> / <time>
Deletes job variables for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = <date>(...)
Deletes job variables for which exactly the specified date is entered as the EXPIRATION-DATE in the catalog entry. The user can specify the expiration date in the form [yy]yy-mm-dd.

TIME = *ANY / *INTERVAL(...)
Restricts the selection of job variables to a time interval related to the specified expiration date. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TIME = *INTERVAL(...)
Deletes job variables for which the expiration date lies within the specified interval that follows.

FROM = 00:00:00 / <time>
Deletes job variables for which the time of expiration ≥ the specified time.

TO = 23:59:59 / <time>
Deletes job variables for which the time of expiration ≤ the specified time.

EXPIRATION-DATE = *INTERVAL(...)
Deletes only job variables whose expiration dates lie within the period specified below, i.e. files whose retention period expires after the specified period. The upper and lower limits are both included in the range specified. See also the explanation of how dates are specified in the EXPIRATION-DATE=<integer ...> operand. It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). Whichever of the operands is not specified will be replaced by the default value for use as the limit of the range. The use of range limits for information output can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = 1950-01-01 / <integer -99999..991231>(...) /
<date>(...) / *TOMORROW(...) / *TODAY(...) / *YESTERDAY(...)
Deletes job variables for which the EXPIRATION-DATE ≥ the specified date.

FROM = <integer -99999..991231>(...)
Deletes job variables for which the EXPIRATION-DATE ≥ the specified date.

TIME = 00:00:00 / <time>
Time on the specified date. All job variables that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = <date>(...)
Deletes job variables for which the EXPIRATION-DATE ≥ the specified date.

> **TIME = 00:00:00 / <time>**
> Time on the specified date. All job variables that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = TOMORROW(...)
Deletes job variables for which the EXPIRATION-DATE ≥ date of the next day.

> **TIME = 00:00:00 / <time>**
> Time on the specified date. All job variables that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = *TODAY(...)
Deletes job variables for which the EXPIRATION-DATE ≥ date of the current day.

> **TIME = 00:00:00 / <time>**
> Time on the specified date. All job variables that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

FROM = *YESTERDAY(...)
Deletes job variables for which the EXPIRATION-DATE ≥ date of the preceding day.

> **TIME = 00:00:00 / <time>**
> Time on the specified date. All job variables that have an EXPIRATION-DATE after the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = *ANY / *TODAY(...) / *TOMORROW(...) / *YESTERDAY(...) / <integer -99999..991231>(...) / <date>(...)
Deletes job variables for which the EXPIRATION-DATE ≤ the specified date.

TO = *ANY
Specifies an open interval that starts with the EXPIRATION-DATE but has no upper limit.

TO = *TODAY(...)
Deletes job variables for which the EXPIRATION-DATE ≤ date of the current day.

> **TIME = 23:59:59 / <time>**
> Time on the specified date. All job variables that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = *TOMORROW(...)
Deletes job variables for which the EXPIRATION-DATE ≤ date of the next day.

TIME = 23:59:59 / <time>
Time on the specified date. All job variables that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = *YESTERDAY(...)
Deletes job variables for which the EXPIRATION-DATE ≤ date of the preceding day.

TIME = 23:59:59 / <time>
Time on the specified date. All job variables that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = <integer -99999..991231>(...)
Deletes job variables for which the EXPIRATION-DATE ≤ the specified date.

TIME = 23:59:59 / <time>
Time on the specified date. All job variables that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

TO = <date>(...)
Deletes job variables for which the EXPIRATION-DATE ≤ the specified date.

TIME = 23:59:59 / <time>
Time on the specified date. All job variables that have an EXPIRATION-DATE before the specified time are selected. Note that the time stamp for the EXPIRATION-DATE is always set to 00:00:00 at present!

BASIC-ACL = *ANY / *NONE / *YES / *PARAMETERS(...)
Deletes job variables whose BASIC-ACL entry matches the specified values.

BASIC-ACL = *NONE
Deletes job variables that have no BASIC-ACL entry in the catalog.

BASIC-ACL = *YES
Deletes job variables that have a BASIC-ACL entry in the catalog.

BASIC-ACL = *PARAMETERS(...)
Deletes job variables for which the specified access rights are defined in the BASIC-ACL entry. NO-ACCESS means that no access rights have been defined.

i Access rights specified with the OWNER, GROUP and OTHERS operands within the *PARAMETERS(...) structure are logically ORed.

OWNER = <u>*ANY</u> / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights must already be defined for the owner.

OWNER = *PARAMETERS(...)
Access rights that must be present for the owner (logically ORed together):

READ = <u>*ANY</u> / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = <u>*ANY</u> / *NO / *YES
Specifies whether write access authorization must be present.

GROUP = <u>*ANY</u> / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights should already be defined for the owner’s user group.

GROUP = *PARAMETERS(...)
Access rights that must be present for the owner’s user group (logically ORed together):

READ = <u>*ANY</u> / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = <u>*ANY</u> / *NO / *YES
Specifies whether write access authorization must be present.

OTHERS = <u>*ANY</u> / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights should already be defined for all other users.

OTHERS = *PARAMETERS(...)
Access rights that must be present for all other users (logically ORed together):

READ = <u>*ANY</u> / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = <u>*ANY</u> / *NO / *YES
Specifies whether write access authorization must be present.

GUARDS = *ANY / *NONE / *YES / *PARAMETERS(...)
Deletes job variables for which access is controlled using GUARDS (see the CREATE-FILE or MODIFY-FILE-ATTRIBUTES command).

GUARDS = *NONE
Deletes job variables that are not protected by GUARDS against unauthorized access.

GUARDS = *YES
Deletes job variables that are protected by GUARDS against unauthorized access, i.e. for which access is controlled via GUARDS.

GUARDS = *PARAMETERS(...)
Deletes job variables which are protected by GUARDS against unauthorized access as specified, i.e. job variables for which access control is implemented using GUARDS: Access to the job variable is controlled by a guard, i.e. a special object which contains all the conditions under which a user is granted access authorization (e.g. date, time, user ID). The GUARDS function unit of the chargeable software product SECOS (see the “SECOS” manual [35]) must be installed in order to create and maintain a guard. Each access mode can be controlled by a separate guard. If no guard (*NONE) is defined for a given access mode, no corresponding access is permitted. If a defined guard is not accessible, the mode of access protected by it is not permitted. If the GUARDS subsystem is not available at the time of accessing the job variable, no access of any kind is allowed for the job variable.

ℹ The values specified for the following READ and WRITE operands will be linked by logical AND.

READ = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>
Deletes job variables which are protected against unauthorized read access by the specified guard. The default value *ANY means that the selection of job variables is not based on read protection with a guard. *NONE selects job variables for which no read guard was defined, i.e. job variables for which no read access is permitted.

WRITE = *ANY / *NONE / <filename 1..18 without-cat-gen-vers>
Deletes job variables which are protected against unauthorized write access by the specified guard. The default value *ANY means that the selection of job variables is not based on a guard. *NONE selects job variables for which no guard was defined, i.e. job variables for which no write access is permitted.

MANAGEMENT-CLASS = *ANY / *NONE / <composed-name 1..8>
Deletes job variables assigned the specified HSMS management class.

MANAGEMENT-CLASS = *NONE
Deletes job variables with no HSMS management class assigned to them

MANAGEMENT-CLASS = <composed-name 1..8>
Deletes job variables assigned the specified HSMS management class.

PROTECTION-ACTIVE = *ANY / list-poss(3): *LEVEL-0 / *LEVEL-1 / *LEVEL-2
Deletes job variables for which the specified protection level is the highest activated access control.
When the job variable is accessed, the highest activated protection level applies. The following table shows the method used for access control, the protection attributes, and the job variable protection hierarchy (protection levels):

Access control method	**Protection attribute**	**Prot. level**
Standard access control	ACCESS and USER-ACCESS	0
Basic access control list	BASIC-ACL	1
Access control via guards	PASSWORD	2

Table 48: Hierarchy of access control methods

All other protection attributes of the job variable (e.g. passwords) are evaluated independently, without regard to the implemented protection level.

PROTECTION-ACTIVE = *LEVEL-0
Deletes job variables for which access is controlled via standard access control.

PROTECTION-ACTIVE = *LEVEL-1
Deletes job variables for which access is controlled via a basic access control list (BASIC-ACL protection)

PROTECTION-ACTIVE = *LEVEL-2
Deletes job variables which are accessed via GUARDS.

SIZE =* ANY / <integer 0..256> / *INTERVAL(...)
Requests information on job variables depending on the length of their value.

SIZE = <integer 0..256>
Deletes job variables whose value is the specified number of bytes in length.

SIZE = *INTERVAL(...)

Deletes job variables for which the length of the job variable value lies within the specified range. The upper and lower limits are both included in the range specified. See also the explanation of the operand SIZE = <integer> It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for information output can only be meaningful if the chosen lower limit ≤ the upper limit. Information will only be returned about job variables for which the length of the value is at least equal to the FROM value and at most equal to the TO value (FROM ≤ SIZE ≤ TO).

FROM = 0 / <integer 0..256>

Deletes job variables for which the length of the value is at least the specified number of bytes (SIZE ≥ specified value).

TO = 256 / <integer 0..256>

Deletes job variables for which the length of the value is at most the specified number of bytes (SIZE ≤ specified value).

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
1	0	CMD0001	No action necessary
2	0	CMD0001	Command executed with a warning
	1	CMD0202	Syntax error
	32	CMD0221	System error
	64	JVS04E0	Command not executable in the call environment; if possible, remove cause of error (see SYSOUT message JVS04xx)
	130	JVS04E1	Command cannot be executed at this time; for cause see SYSOUT message JVS04xx
	130	CMD2282	Subsystem JV not available for indefinite time

DELETE-OPERATOR-ROLE

Delete operator role

Description status:	SRPMNUC V19.0A
Functional area:	Operator function control
Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

Function

This command is used to delete a list of routing (authorization) codes which has been compiled to form an “operator role” and stored in the user catalog of a pubset. An operator role corresponds to an area of work, and gives the exerciser of the role the right to issue the commands assigned to that area of work. Each area of work (= operator role) is represented by a set of authorization (routing) codes specified by the system administration; this may be any combination of the total of 40 authorization codes used in BS2000. The system administration uses the MODIFY-OPERATOR-ATTRIBUTES command to assign to each operator identification the operator roles which it may undertake.

The DELETE-OPERATOR-ROLE command can only be used to delete roles which are not assigned to any operator identification.

Format

DELETE-OPERATOR-ROLE
OPERATOR-ROLE = list-poss(20): <name 1..8> ,PUBSET = *HOME / <cat-id 1..4>

Operands

OPERATOR-ROLE = list-poss(20): <name 1..8>
Names of the operator roles to be deleted. In total, the names of 20 operator roles may be specified. These roles must not be assigned to any user ID (= operator identification) in the user catalog.

PUBSET =
Specifies the pubset from whose user catalog the role is to be deleted.

PUBSET = *HOME
The operator role is to be deleted from the user catalog of the home pubset.

PUBSET = <cat-id 1..4>
Exact specification of the pubset from whose catalog the operator role is to be deleted.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
2	0	SRM6001	Command executed with a warning
	32	SRM6020	System error during command processing
	64	SRM6040	Semantic error
	130	SRM6030	Command cannot temporarily be executed

DELETE-PAGING-FILE

Delete paging file from disk

Description status:	BS2000 OSD/BC V10.0A
Functional area:	System control and optimization
Domain:	SYSTEM-MANAGEMENT SYSTEM-TUNING
Privileges:	OPERATING TSOS
Routing code:	R

Function

This command is reserved for systems support staff. It deletes a paging file from a hard disk. The command will only work if the pubset the disk belongs to has been imported. When the paging file is deleted, the space it was occupying is overwritten with binary zeros.

Format

DELETE-**PAG**ING-**FILE**
VOLUME = list-poss(256): <vsn 1..6>

Operands

VOLUME = list-poss(256): <vsn 1..6>
Identifies by volume serial number the disk(s) from which one or more paging files are to be deleted.
Up to 256 disks can be selected.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed without error
	32	EMM2800	Internal error Guaranteed messages: EMM2818, EMM2828
	128	EMM2807	Operation cannot be performed because of shortage of resources Guaranteed messages: EMM2819, EMM2829

Note

Command processing is aborted in the event of an error with a return code of EMM2800 or EMM2807 occurring while a list is being specified.

Notes

- The command may take a number of minutes to process. For that reason it is not executed in the calling task but is delegated to a server task generated for the purpose. The message indicating that the job has been completed is delivered to the calling task asynchronously. If the calling task terminates before the completion message is delivered, the message is sent to the operator terminal (console).
- If more than one disk device is specified when the command is issued, a separate job is started for each device.

Example

Deleting the paging file from a disk with the volume serial number 2OSW.0:

```
/DELETE-PAGING-FILE VOLUME=2OSW.0
```

DELETE-SNAPSET

Delete Snapset

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Pubset and MRSCAT management
Domain:	MULTI-CATALOG-AND-PUBSET-MGMT SNAPSET
Privileges:	TSOS HSMS-ADMINISTRATION

Function

The DELETE-SNAPSET command deletes one or all existing Snapsets for an imported pubset. When a Snapset is deleted, the individual volumes of the Snapset are cleared and removed from the Snapset catalog. When all existing Snapsets are deleted, the context required for Snapset operation is also removed: the Snapset catalog is deleted and the Snapset limit is set to zero.

Format

DELETE-SNAPSET
PUBSET = <cat-id 1..4> ,**SNAPSET** = ***EARLIEST** / <name 1..1 with-low> / <integer -52..-1> / ***ALL**

Operands

PUBSET = <cat-id 1..4>
Catalog ID of the pubset for which the specified Snapset is to be deleted.

SNAPSET = *EARLIEST / <name 1..1 with-low> / <integer -52..-1> / *ALL
Specifies which Snapsets are to be deleted.

> ℹ When an ETERNUS DX is used, only the oldest Snapset can be deleted. This must be borne in mind when a Snapset which is to be deleted is specified explicitly using the Snapset ID or the relative age.

SNAPSET = *EARLIEST
The oldest Snapset is deleted.

SNAPSET = <name 1..1 with-low>
Specifies the Snapset explicitly by means of the Snapset ID. The maximum of 52 pubsets are distinguished by means of Snapset IDs specified which comprise letters from the 26 lowercase letters a to z and the 26 uppercase letters A to Z.

SNAPSET = <integer -52..-1>
The Snapset to be deleted is specified by means of its relative age. The value -1 specifies the latest Snapset here.

SNAPSET = *ALL
All Snapsets are deleted and the context for Snapset operation of this pubset is removed.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without errors
	32	CMD0216	No authorization
	64	DMS1351	Internal error
	64	DMS1386	Error in the memory request
	64	DMS1389	Error in MSCF communication
	64	DMS138B	No MRSCAT entry
	64	DMS138C	Pubset not accessible
	64	DMS13D5	The Snapset specified does not exist on the pubset
	64	DMS148F	GCF not loaded
1	64	DMS13D7	Internal error in Snapset management: Return code of GCF
2	64	DMS13D7	Internal error in Snapset management: Return code in the case of mount/demount
3	64	DMS13D7	Internal error in Snapset management: Return code when reading or writing the SVL
4	64	DMS13D7	Internal error in Snapset management: Return code when setting/resetting the reconfiguration lock
6	64	DMS13D7	Internal error in Snapset management: Return code of SHC-OSD
7	64	DMS13D7	Internal error in Snapset management: Return code of CCOPY
	64	DMS13DF	Subsystem SHC-OSD not available

DELETE-STORAGE-CLASS

Delete storage class definition

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Pubset and MRSCAT management
Domain:	MULTI-CATALOG-AND-PUBSET-MGMT
Privileges:	TSOS

Functions

The DELETE-STORAGE-CLASS command enables systems support to delete storage classes from the storage class catalog. If more than one storage class is specified, the SELECT operand can be used to restrict the set of storage classes being deleted to those with specific attributes. The DIALOG-CONTROL operand defines the conditions under which a dialog is conducted to verify that objects are to be deleted.

The SM pubset must have been imported to the local system (in exclusive or shared mode).

Before deleting a storage class, note that references to that storage class may still exist. The following types of reference are possible:

The storage class is already assigned to existing files. You can find out which files these are with the SHOW-FILE-ATTRIBUTES command and the STORAGE-CLASS selection criterion. Before deleting the storage class you should assign these files to another class or cancel their storage class assignment. Otherwise the files might be given an inappropriate storage location if migrated with HSMS and then recalled.

The storage class is assigned to user IDs as their default storage class (see the MODIFY-USER-PUBSET-ATTRIBUTES command). You can find out which user IDs these are with the SHOW-USER-ATTRIBUTES command. Before deleting the storage class you should assign these user IDs a different default storage class or cancel the default storage class assignment. Otherwise the users affected will have to specify a storage class or the file attributes explicitly when creating files in the pubset in question.

Format

```
DELETE-STORAGE-CLASS

 STORAGE-CLASS-NAME = *ALL / <composed-name 1..8 with-wild>
,PUBSET = <cat-id 1..4>
,SELECT = *ALL / *PARAMETERS(...)
    *PARAMETERS(...)
    | FILE-ATTRIBUTES = *ANY / *PARAMETERS(...)
    |     *PARAMETERS(...)
    |     | IO-ATTRIBUTES = *ANY / *PARAMETERS(...)
    |     |     *PARAMETERS(...)
    |     |     | PERFORMANCE = *ANY / list-poss(3): *STD / *HIGH / *VERY-HIGH
    |     |     |,USAGE = *ANY / list-poss(3): *READ-WRITE / *WRITE / *READ
    |     |,DISK-WRITE = *ANY / list-poss(3): *STD / *IMMEDIATE / *BY-CLOSE
    |     |,AVAILABILITY = *ANY / list-poss(2): *STD / *HIGH
    |     |,FILE-PREFORMAT = *ANY / list-poss(4): *BY-PUBSET-DEFAULT / *K / *NK2 / *NK4
    |     |,WORK-FILE = *ANY / list-poss(2): *NO / *YES
    |,VOLUME-SET-LIST = *ANY / *NONE / <composed-name 1..8>
,DIALOG-CONTROL = *STD / *NO / *ERROR / *CLASS-CHANGE / *MORE-THAN-ONE-CLASS
```

Operands

STORAGE-CLASS-NAME = *ALL / <composed-name 1..8 with-wild>
Specifies which storage classes to delete.

STORAGE-CLASS-NAME = *ALL
The storage classes to delete are all those which are assigned to the pubset specified in the PUBSET operand and which also meet the selection criteria defined in the SELECT operand.

STORAGE-CLASS-NAME = <composed-name 1..8 with-wild>
Name of the storage class that is to be deleted. Wildcards can be used to select more than one storage class.
The storage classes to delete are all those specified here which are assigned to the pubset specified in the PUBSET operand and which also meet the selection criteria defined in the SELECT operand.

PUBSET = <cat-id 1..4>
Catalog ID of the associated SM pubset. The SM pubset must have been imported to the local system (in exclusive or shared mode).

SELECT = <u>*ALL</u> / *PARAMETERS(...)
Specifies whether the set of storage classes specified in the STORAGE-CLASS-NAME operand is to be restricted by selection criteria.

SELECT = <u>*ALL</u>
There is no restriction on the set of storage classes specified in the STORAGE-CLASS-NAME operand.

SELECT = *PARAMETERS(...)
Defines selection criteria to restrict the set of storage classes specified in the STORAGE-CLASS-NAME operand. Only those storage classes which possess all the attributes specified next are to be deleted.
The default value, *ANY, means that the attribute in question is not a selection criterion.

FILE-ATTRIBUTES = <u>*ANY</u> / *PARAMETERS(...)
Specifies whether the selection of storage classes to delete is to be based on the file attributes defined in them.

FILE-ATTRIBUTES = <u>*ANY</u>
The file attributes are not a selection criterion.

FILE-ATTRIBUTES = *PARAMETERS(...)
The storage classes which are to be deleted are selected on the basis of the file attributes specified next.

IO-ATTRIBUTES = <u>*ANY</u> / *PARAMETERS(...)
Specifies whether the performance attributes are to be a selection criterion.

IO-ATTRIBUTES = *PARAMETERS(...)
The storage classes which are to be deleted must have definitions containing the performance attributes specified next.

PERFORMANCE = <u>*ANY</u> / list-poss(3): *STD / *HIGH / *VERY-HIGH
Deletes all storage classes which possess one of the specified performance attributes.

PERFORMANCE = <u>*ANY</u>
The performance attribute is not a selection criterion.

PERFORMANCE = *STD
Deletes only those storage classes which have their performance attribute set to *STD (no special performance requirement).

PERFORMANCE = *HIGH
Deletes only those storage classes which have their performance attribute set to *HIGH (high performance requirement).

PERFORMANCE = *VERY-HIGH
Deletes only those storage classes which have their performance attribute set to *VERY-HIGH (highest performance requirement).

USAGE = <u>*ANY</u> / list-poss(3): *READ-WRITE / *WRITE / *READ
Deletes all storage classes which have the performance attribute set for one of the specified I/O operations.

USAGE = <u>*ANY</u>
The I/O operation mode is not a selection criterion.

USAGE = *READ-WRITE
Deletes only those storage classes which have their performance attribute set for read/write operations.

USAGE = *WRITE
Deletes only those storage classes which have their performance attribute set for write operations.

USAGE = *READ
Deletes only those storage classes which have their performance attribute set for read operations.

DISK-WRITE = <u>*ANY</u> / list-poss(3): *STD / *IMMEDIATE / *BY-CLOSE
Specifies whether the attribute governing the time when data consistency is required after write operations is to be a selection criterion.

DISK-WRITE = <u>*ANY</u>
The point at which data consistency is required is not a selection criterion.

DISK-WRITE =*STD
Deletes only those storage classes which have their consistency point defined as DISK-WRITE=*STD.

DISK-WRITE = *IMMEDIATE
Deletes only those storage classes which have their consistency point defined as immediately after completion of write operations.

DISK-WRITE = *BY-CLOSE
Deletes only those storage classes which have their consistency point defined as only after CLOSE processing.

AVAILABILITY = <u>*ANY</u> / list-poss(2): *STD / *HIGH
Specifies whether the availability attribute is to be a selection criterion.

AVAILABILITY = <u>*ANY</u>
The required availability is not to be a selection criterion.

AVAILABILITY = *STD
Deletes only those storage classes for which no special availability requirements are defined.

AVAILABILITY = *HIGH
Deletes only those storage classes for which high availability requirements are defined.

FILE-PREFORMAT = *ANY / list-poss(4): *BY-PUBSET-DEFAULT / *K / *NK2 / *NK4
Specifies whether the attribute for preferred file format is to be a selection criterion.

FILE-PREFORMAT = *ANY
The preferred file format is not to be a selection criterion.

FILE-PREFORMAT = *BY-PUBSET-DEFAULT
Deletes only those storage classes for which the preferred file format is defined as the pubset-global default.

FILE-PREFORMAT = *K
Deletes only those storage classes in which files are set to be created as K files.

FILE-PREFORMAT = *NK2
Deletes only those storage classes in which files are set to be created as NK2 files.

FILE-PREFORMAT = *NK4
Deletes only those storage classes in which files are set to be created as NK4 files.

WORK-FILE = *ANY / list-poss(2): *NO / *YES
Specifies whether the work file attribute is to be a selection criterion.

WORK-FILE = *ANY
The work file attribute is not to be a selection criterion.

WORK-FILE = *NO
Deletes only those storage classes in which files are created as normal files.

WORK-FILE = *YES
Deletes only those storage classes in which files are created as work files.

VOLUME-SET-LIST = *ANY / *NONE / <composed-name 1..8>
Specifies whether the storage classes are to be selected on the basis of volume set list assignment.

VOLUME-SET-LIST = *ANY
Volume set list assignment is not to be a selection criterion.

VOLUME-SET-LIST = *NONE
Deletes only those storage classes to which no volume set list is assigned.

VOLUME-SET-LIST = <composed-name 1..8>
Deletes only those storage classes to which the specified volume set list is assigned.

DIALOG-CONTROL = *STD / *NO / *ERROR / *CLASS-CHANGE / *MORE-THAN-ONE-CLASS
Specifies whether and under what conditions a verification dialog is to be conducted with the user during the deletion process.
The verification dialog mechanism is available only in interactive mode, though that does include procedures.In batch mode all values have the same effect as *NO.

Systems support can intervene with the following inputs:

Y	the specified storage class or storage class set will then be deleted.
N	the specified storage class or storage class set will not be deleted.
T	processing of the command will be terminated.
S	the specified storage class or storage class set will be displayed. The verification dialog will then be repeated.
?	the possible responses will be listed, with an explanation of each.

DIALOG-CONTROL = *STD
The default *STD setting is equivalent to *MORE-THAN-ONE-FILE in an interactive dialog (when SYSCMD is connected to the terminal) and to *NO in procedures and in batch mode.

DIALOG-CONTROL = *NO
Systems support cannot intervene in DELETE-STORAGE-CLASS processing; all the specified files will be deleted (without a verification dialog).

DIALOG-CONTROL = *ERROR
If deletion of the selected storage classes proceeds without error, all the storage classes will be deleted immediately, as when *NO is specified (i.e. no verification dialog). However, if a correctable error occurs during deletion, then a verification dialog takes place for each storage class which it was not possible to delete.
In batch mode this value has the same effect as *NO.

DIALOG-CONTROL = *CLASS-CHANGE
A verification dialog is conducted for each storage class selected. In batch mode this value has the same effect as *NO.

DIALOG-CONTROL = *MORE-THAN-ONE-CLASS
If exactly one storage class is specified or identified by the selection criteria, this will be deleted immediately. If more than one storage class is to be deleted, a verification dialog is conducted for the set which is to be deleted. In batch mode this value has the same effect as *NO.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
	1	CMD0202	Syntax error in command
	32	CMD0221	Internal system error
	64	CMD0216	No authorization to issue command
	64	DMS1481	Error accessing storage class catalog
	64	DMS1483	Storage class catalog invalid
	64	DMS1485	Pubset not known
	64	DMS1486	Pubset is not a system-managed pubset
	64	DMS1487	Pubset not available
	64	DMS148A	Storage class not defined
	64	DMS1490	Storage class management not available for this pubset
	64	DMS149A	No storage classes meet the selection criteria
	64	DMS149C	No storage class exists for the specified pubset
	129	DMS148D	Not enough class 4/5 memory
	129	DMS148E	Error on MSCF connection to master
	129	DMS148F	GCF subsystem not ready

DELETE-SYSTEM-FILE

Delete system file

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing File processing
Domain:	FILE JOB PROCEDURE PROGRAM
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION
Routing code:	$ (with NBCONOPI=N) or E (with NBCONOPI=Y)

Function

The DELETE-SYSTEM-FILE command deletes the contents of a system file logically. If a file has been allocated to the system file to be deleted, the contents of that file will be logically deleted, but the allocation will remain in effect. This enables the user to delete an allocated file without removing the allocation to a system file. The DELETE-FILE command cannot be used on the allocated file, since the file is open.

If the logical system file SYSLST is specified, only the part that was created after the last COPY-SYSTEM-FILE command can be deleted, regardless of whether or not the COPY-SYSTEM-FILE command was executed successfully.

See section “System files” on page 1-73 and the description of the COPY-SYSTEM-FILE command.

Format

```
DELETE-SYSTEM-FILE                                                        Alias: DLSF

SYSTEM-FILE = *SYSLST / *SYSOUT / *OMF / *SYSLST-NUMBER(...) / *ALL-SYS(...)
  *SYSLST-NUMBER(...)
    | SYSLST-NUMBER = <integer 1..99> / <filename 1..2 with-wild(73)>
  *ALL-SYS(...)
    | SYS-IDENTIFIER = <filename 1..3 with-wild(76)>
,DIALOG-CONTROL = *STD / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE
,OUTPUT = *STD / *NO / *SYSOUT
,SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
```

Operands

SYSTEM-FILE = *SYSLST / *SYSOUT / *OMF / *SYSLST-NUMBER(...) / *ALL-SYS(...)
The name of the system file to be deleted.
The contents of the specified system file, or the (cataloged) SAM file allocated to it, will be logically deleted.
Any existing allocation will be retained. After its deletion, data will be written to the file starting at its beginning.

SYSTEM-FILE = *SYSLST
The contents of the specified system file, or the (cataloged) SAM file allocated to it, will be logically deleted.

SYSTEM-FILE = *SYSOUT
The contents of the specified system file, or the (cataloged) SAM file allocated to it, will be logically deleted.

SYSTEM-FILE = *OMF
Deletes the EAM object module file, which is produced and used by the language processor, for the job currently running. If this file is empty, the command will be ignored. This case does not trigger a spin-off mechanism or SDF-P error handling in procedures.

SYSTEM-FILE = *SYSLST-NUMBER(...)
The contents of the allocated SAM file will be logically deleted. The existing allocation will be retained; after its deletion, data will be written to the file starting at its beginning. The command will be rejected if the system file SYSLST-NUMBER is not allocated to a (cataloged) SAM file.

SYSLST-NUMBER = <integer 1..99> / <filename 1..2 with-wild(73)
The number of the SYSLST file which is to be deleted. This must always be given as a two-digit number. Several numbers may be specified by using a sequence of wildcards.

SYSTEM-FILE = *ALL-SYS(...)
Enables several system output files to be deleted simultaneously.

SYS-IDENTIFIER = <filename 1..3 with-wild(76)>
Deletes one or more system output files, which can be specified by a sequence of wildcards. The name of the system file is specified without its SYS prefix, e.g. LST for the SYSLST system file.

DIALOG-CONTROL = <u>*STD</u> / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE
Specifies whether and under what conditions a verification dialog is to be conducted with the user during the deletion process.
The verification dialog mechanism is available only in interactive mode, though that does include procedures. The only operand value that you can specify in batch mode is *STD or *NO.

The user can intervene with the following inputs:

– Y: the specified file or file set will then be deleted.
– N: the specified file or file set will not be deleted.
– T: processing of the command will be terminated.
– ?: the possible responses will be listed, with an explanation of each.

In addition, the following options can be specified, separated by commas:

– ,CHECK = NO
The DIALOG-CONTROL mode will be changed to ‘*NO’.
– ,CHECK = MULTIPLE
The DIALOG-CONTROL mode will be changed to ‘*MORE-THAN-ONE-FILE’.
– ,CHECK = SINGLE
The DIALOG-CONTROL mode will be changed to ‘*FILE-CHANGE’.
– ,CHECK = ERROR
The DIALOG-CONTROL mode will be changed to ‘*ERROR’.
– ,IGNORE = list-poss(2): ACCESS / EXDATE
Specifies which protection attributes are to be ignored during deletion.
– ,PASSWORD = list-poss(3): <c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647>
Enables password-protected files to be deleted (maximum of 3 passwords).

DIALOG-CONTROL = *STD
The default *STD setting is equivalent to *MORE-THAN-ONE-FILE in an interactive dialog (when SYSCMD is connected to the terminal) and to *NO in procedures and in batch mode.

DIALOG-CONTROL = *NO
The user cannot intervene in DELETE-SYSTEM-FILE processing; all the specified files will be deleted.

DIALOG-CONTROL = *ERROR
If deletion of the selected files proceeds without error, they will be deleted immediately, as when *NO is specified (i.e. no verification dialog). However, if a user-correctable error occurs, then a verification dialog takes place as for DIALOG-CONTROL=*FILE-CHANGE. In the event of an error, the user may acknowledge the error message, abort DELETE-SYSTEM-FILE processing or attempt to rectify the error. If he wishes, he can also change the DIALOG-CONTROL mode (see also the possible forms of intervention listed under the first DIALOG-CONTROL operand).

DIALOG-CONTROL = *FILE-CHANGE
For each system file to be deleted, the user has the intervention options described under the first DIALOG-CONTROL operand. For each file which is to be processed, the user can decide in a dialog whether it should be deleted or not (response: YES/NO). If in the dialog he specifies protection attributes under “IGNORE”, or one or more passwords under “PASSWORD”, these will be taken into account for any selected file and, if satisfied, the file will be deleted without further queries (“YES” must also be specified). The user can also abort DELETE-SYSTEM-FILE processing, or change the DIALOG-CONTROL mode.

DIALOG-CONTROL = *MORE-THAN-ONE-FILE
If exactly one system file is specified, this file will be deleted immediately. If SYSTEM-FILE is partially qualified, which means that more than one file is selected, or if “pathname” contains wildcards, the user can decide, each time the catalog ID changes, whether or not files from the new catalog are to be deleted (see the intervention options described for the first of the DIALOG-CONTROL operands). He must respond to the question issued by the system with “YES” or “NO”. DIALOG-CONTROL = *MORE-THAN-ONE-FILE is useful if wildcards are specified for “catid” in the FILE-NAME operand. In the dialog, DELETE-SYSTEM-FILE processing can be terminated, or the DIALOG-CONTROL mode can be changed.

OUTPUT = *STD / *NO / *SYSOUT
The user can specify whether a message (`DMS0800`) with the name of the deleted file is to be output to SYSOUT for each successful deletion. The default setting *STD is equivalent to OUTPUT=*NO.

OUTPUT = *NO
No messages are output to SYSOUT for successfully deleted files.

OUTPUT = *SYSOUT
For each file that is successfully deleted, a message with the name of that file is output to SYSOUT.

SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
In procedures the user can specify whether the spin-off mechanism or SDF-P error handling is to be triggered every time an error occurs (apart from syntax errors), or whether specific error conditions are to be ignored.

SUPPRESS-ERRORS = *NONE
All errors will trigger the spin-off mechanism or SDF-P error handling. The deletion of an empty system file is not treated as an error.

SUPPRESS-ERRORS = list-poss(3): <alphanum-name 7..7>
The user can define which errors are to be ignored by means of their DMS error codes (alphanum-name 7..7). If the specified error occurs, the spin-off mechanism will not be triggered. A maximum of 3 error codes may be specified.
DMS error code: 7 characters, of which the first three are always “DMS”; the last 4 characters identify the error; the digits 0...9 and letters A...F are permitted. When error codes are entered, no check is made to verify that valid error codes have been specified.

A detailed list of valid DMS error codes can be found on the manual server (URL: *http://manuals.ts.fujitsu.com*) by means of an HTML application and on the “BS2000 SoftBooks” DVD.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed
1	0	CMD0001	No action required
	1	CMD0202	Syntax or semantic error in command
	32	DMS0584	A state that does not allow the function to continue was reported during processing.
	64	DMS0572	SYSLST file not allocated to DMS file
	64	DMS0574	DMS error when deleting a system file Guaranteed message: DMS0574
	64	DMS0587	Use of the specified command has been restricted by systems support
	130	DMS0524	System address space exhausted
	130	DMS0594	Not enough virtual memory available

DELETE-TAPE-SET

Release volume serial number set (tape set)

Description status: BS2000 OSD/BC V10.0A

Functional area: File processing

Domain: FILE

Privileges: STD-PROCESSING

Routing code: $ (with NBCONOPI=N) or E (with NBCONOPI=Y)

Function

The DELETE-TAPE-SET command is used to release the set of volumes specified by means of the CREATE-TAPE-SET or EXTEND-TAPE-SET command.

Format

DELETE**-TAPE-SET**
TAPE-SET-NAME = <alphanum-name 1..4>

Operands

TAPE-SET-NAME = <alphanum-name 1..4>
Name of the tape set.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
	1	CMD0202	Syntax or semantic error in command
	32	DMS0584	During processing a status was reported that renders continuation of the function impossible.
	32	CMD05C7	Unexpected internal error in DMS
	64	DMS0586	It is not possible to access or reserve a volume at present
	64	DMS0587	Use of the specified command has been restricted by the system administrator
	64	DMS06FF	BCAM connection severed
	130	DMS0524	System address space exhausted

DELETE-VARIABLE

Delete variable

Description status:	SDF-P-BASYS V2.5E
Functional area:	Procedures
Domain:	PROCEDURE
Privileges:	STD-PROCESSING OPERATING HARDWARE-MAINTENANCE SECURITY-ADMINISTRATION SAT-FILE-MANAGEMENT SAT-FILE-EVALUATION

Function

DELETE-VARIABLE deletes the declaration of an S variable within the current scope, i.e. including the declarations of imported task variables.

The name of the S variable can no longer be used, and its value is deleted.

Either simple or complex variables can be deleted, but not individual elements of complex variables.

The following variable declarations cannot be deleted using DELETE-VARIABLE:

– procedure parameters
– elements of complex variables
– system variables (e.g. SYSWITCH)
– container JVs
– non-permanent container variables
– structure layouts

Note

In such cases, error message SDP1098 is only output if no wildcards are specified in the variable name.

Format

DELETE-**VAR**IABLE
VARIABLE-**NAME** = <structured-name 1..20 with-wild(40)> / list-poss(2000): <structured-name 1..20>

Operands

VARIABLE-NAME =
Name of the S variable to be deleted.

VARIABLE-NAME = <structured-name 1..20 with-wild(40)>
All the S variables which match this search pattern are deleted.

VARIABLE-NAME = list-poss(2000):<structured-name 1..20>
List of S variables to be deleted.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
1	0	CMD0001	Warning; nothing executed
	1	CMD0202	Syntax error
	3	CMD2203	Incorrect syntax file
	32	CMD0221	System error (internal error)
	64	CMD0216	Do not have required privilege
	64	SDP0091	Semantic error
	130	SDP0099	No further address space available

DELETE-VOLUME-SET-LIST

Delete volume set list

Description status: BS2000 OSD/BC V10.0A

Functional area: Pubset and MRSCAT management

Domain: MULTI-CATALOG-AND-PUBSET-MGMT

Privileges: TSOS

Function

The DELETE-VOLUME-SET-LIST command enables systems support to delete volume set lists from the volume set list catalog. If more than one volume set list is specified, the SELECT operand can be used to restrict the set of volume set lists being deleted to those containing specific volume sets. The DIALOG-CONTROL operand defines the conditions under which a dialog is conducted to verify that objects are to be deleted.

The SM pubset must have been imported to the local system (in exclusive or shared mode).

Format

DELETE-VOLUME-SET-LIST
VOLUME-SET-LIST-NAME = *ALL / <composed-name 1..8 with-wild>
,PUBSET = <cat-id 1..4>
,SELECT = *ALL / *BY-VOLUME-SET(...)
*BY-VOLUME-SET(...)
│ ENTRY = list-poss(255): <cat-id 1..4>
,DIALOG-CONTROL = *STD / *NO / *ERROR / *LIST-CHANGE / *MORE-THAN-ONE-LIST

Operands

VOLUME-SET-LIST-NAME = *ALL / <composed-name 1..8 with-wild>

Specifies which volume set lists to delete.

VOLUME-SET-LIST-NAME = *ALL

The volume set lists to delete are all those which are assigned to the pubset specified in the PUBSET operand and which also meet the selection criteria defined in the SELECT operand.

VOLUME-SET-LIST-NAME = <composed-name 1..8 with-wild>
Name of the volume set list that is to be deleted. Wildcards can be used to select more than one volume set list.
The volume set lists to delete are all those specified here which are assigned to the pubset specified in the PUBSET operand and which also meet the selection criteria defined in the SELECT operand.

PUBSET = <cat-id 1..4>
Catalog ID of the associated SM pubset. The SM pubset must have been imported to the local system (in exclusive or shared mode).

SELECT = *ALL / *BY-VOLUME-SET(...)
Specifies whether the set of volume set lists specified in the VOLUME-SET-LIST-NAME operand is to be restricted by selection criteria.

SELECT = *BY-VOLUME-SET(...)
The selection is to be restricted to volume set lists which contain any of the volume sets specified next.

> **ENTRY = list-poss(255): <cat-id 1..4>**
> Volume set ID.
> Up to 255 volume sets can be listed.

DIALOG-CONTROL = *STD / *NO / *ERROR / *LIST-CHANGE / *MORE-THAN-ONE-LIST
Specifies whether and under what conditions a verification dialog is to be conducted with the user during the deletion process.
The verification dialog mechanism is available only in interactive mode, though that does include procedures. In batch mode all values have the same effect as *NO .

Systems support can intervene with the following inputs:

Y	the specified volume set list or set of volume set lists will then be deleted.
N	the specified volume set list or set of volume set lists will not be deleted.
T	processing of the command will be terminated.
S	the specified volume set list or set of volume set lists will be displayed. The verification dialog will then be repeated.
?	the possible responses will be listed, with an explanation of each.

DIALOG-CONTROL = *STD
The default *STD setting is equivalent to *MORE-THAN-ONE-FILE in an interactive dialog (when SYSCMD is connected to the terminal) and to *NO in procedures and in batch mode.

DIALOG-CONTROL = *NO
Systems support cannot intervene in DELETE-VOLUME-SET-LIST processing; all the specified volume set lists will be deleted (without a verification dialog).

DIALOG-CONTROL = *ERROR
If deletion of the selected volume set lists proceeds without error, all the volume set lists will be deleted immediately, as when *NO is specified (i.e. no verification dialog). However, if a correctable error occurs during deletion, then a verification dialog takes place for each volume set list which it was not possible to delete.
In batch mode this value has the same effect as *NO.

DIALOG-CONTROL = *LIST-CHANGE
A verification dialog is conducted for each volume set list selected.
In batch mode this value has the same effect as *NO.

DIALOG-CONTROL = *MORE-THAN-ONE-LIST
If exactly one volume set list is specified or identified by the selection criteria, this will be deleted immediately. If more than one volume set list is to be deleted, a verification dialog is conducted for the set which is to be deleted.
In batch mode this value has the same effect as *NO.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
	1	CMD0202	Syntax error in command
	32	CMD0221	Internal system error
	64	CMD0216	No authorization to issue command
	64	DMS1482	Error accessing volume set list catalog
	64	DMS1484	Volume set list catalog invalid
	64	DMS1485	Pubset not known
	64	DMS1486	Pubset is not a system-managed pubset
	64	DMS1487	Pubset not available
	64	DMS148B	Volume set list not defined
	64	DMS1490	Storage class management not available for this pubset
	64	DMS149B	No volume set lists meet the selection criteria
	64	DMS149C	No volume set list exists for the specified pubset
	129	DMS148D	Not enough class 4/5 memory
	129	DMS148E	Error on MSCF connection to master
	129	DMS148F	GCF subsystem not ready

DETACH-DEVICE

Detach hardware units

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Device management
Domain:	DEVICE
Privileges:	OPERATING
Routing code:	G

Function

This command enables the operator to detach one or more hardware units, i.e. prohibit the operating system from using these units.

Effect of the DETACH-DEVICE command

1. Each of the hardware units specified assumes (if permitted) the state “detached explicitly”. In this case, the hardware units cannot be used by the system for input/output.
2. All relevant outward connections assume the state “removed implicitly”. The connections cannot be used.
3. Each outer unit none of whose internal connections is in the “included” state is placed in the state “detached implicitly”. The unit cannot be used.
4. If the units to be detached are hardware units with corresponding SVP actions, these actions are initiated. The units are placed in the configuration state “detached”, regardless of whether or not these actions are successful.
5. If a device which is capable of forming a path group is detached then the path group for this device is eliminated. If a channel or controller is detached then the path groups of all the connected devices are eliminated.
6. If a PAV device (**P**arallel **A**ccess **V**olumes, see the “Introduction to System Administration” [14]) is detached then the following applies:

   If a base device is detached, the associated alias devices switch to the “not ready” status and are implicitly detached. Alias devices can also be explicitly detached by means of the DETACH-DEVICE command.

Format

```
DETACH-DEVICE

UNIT = *CPU(...) / *EXTRA-CPU(...) / *CHANNEL(...) / *CONTROLLER(...) /
          *CHANNEL-RANGE(...) / *DEVICE-RANGE(...) / *PUBSET-DEVICES(...) /
          list-poss(255): <alphanum-name 2..2> / <x-text 4..4>

   *CPU(...)
     |  CPU-IDENTIFIER = list-poss(16): <x-text 2..2>

   *EXTRA-CPU(...)
     |  CPU-IDENTIFIER = *ALL / *ANY / <x-text 2..2>

   *CHANNEL(...)
     |  CHANNEL-PATH-ID = list-poss(16): <x-text 2..2>
     | ,SCOPE = *OWN-SYSTEM-ONLY / *VM2000-GLOBAL

   *CONTROLLER(...)
     |  CONTROLLER-UNIT = list-poss(16):<alphanum-name 2..2> / <x-text 4..4>
     | ,SCOPE = *OWN-SYSTEM-ONLY / *VM2000-GLOBAL

   *CHANNEL-RANGE(...)
     |  FROM = <x-text 2..2>
     | ,TO = <x-text 2..2>
     | ,SCOPE = *OWN-SYSTEM-ONLY / *VM2000-GLOBAL

   *DEVICE-RANGE(...)
     |  FROM = <alphanum-name 2..2> / <x-text 4..4>
     | ,TO = <alphanum-name 2..2> / <x-text 4..4>

   *PUBSET-DEVICES(...)
     |  PUBSET = list-poss(255): <cat-id 1..4> / *BY-PUBRES-DEVICE(...)
     |     *BY-PUBRES-DEVICE(...)
     |       |  UNIT = list-poss(255): <alphanum-name 2..2> / <x-text 4..4>

,FORCE = *STD / *YES / *NO(...) / *UNCONDITIONAL-OFFLINE

   *NO(...)
     |  WAIT = *NO / *STD / <integer 1..32767>(...)
     |     <integer 1..32767>(...)
     |       |  DIM = *STD / *MIN / *SEC
```

Operands

UNIT =
Specifies the hardware units to be detached from the system.

UNIT = *CPU(...)
Specifies the CPU to be detached.

CPU-IDENTIFIER = list-poss(16): <x-text 2..2>
Specifies the CPU identifier.

UNIT = *EXTRA-CPU(...)
Specifies the extra CPU that is to be detached.

CPU-IDENTIFIER = *ALL / *ANY / <x-text 2..2>
Specifies the identifier of the extra CPU. *ANY is used to detach any given extra CPU while *ALL detaches all available extra CPUs.

UNIT = *CHANNEL(...)
Specifies the channel to be detached.

CHANNEL-PATH-ID = list-poss(16): <x-text 2..2>
Specifies the channel path ID of the channel. In the case of channels, the device code (MN) is the same as the hexadecimal channel path ID.

SCOPE =
Specifies how the command is to be executed under VM2000.

SCOPE = <u>*OWN-SYSTEM-ONLY</u>
The command is only executed in the local system.

SCOPE = *VM2000-GLOBAL
If entered at the Monitor System (S server), the command is executed at all guest systems. If entered at another guest system or at the Monitor System (SQ server), the command is rejected with the message NKR0178.
The command is rejected if one of the guest systems is protected against global detachment (system parameter VMGIORAL=NO) and the channel that is to be detached does not yet possess the status DETACHED (unless FORCE= *UNCONDITIONAL-OFFLINE has been specified).

UNIT = *CONTROLLER(...)
Specifies the controller to be detached.

CONTROLLER-UNIT = list-poss(16): <alphanum-name 2..2> / <x-text 4..4>
Specifies the mnemonic device code (MN) of the controller to be detached.

SCOPE =
Specifies how the command is to be executed under VM2000.

SCOPE = *OWN-SYSTEM-ONLY
The command is only executed in the local system.

SCOPE = *VM2000-GLOBAL
If entered at the Monitor System (S server), the command is executed at all guest systems. If entered at another guest system or at the Monitor System (SQ server), the command is rejected with the message NKR0178.
The command is rejected if one of the guest systems is protected against global detachment (system parameter VMGIORAL=NO) and the controller that is to be detached does not yet possess the status DETACHED.

UNIT = *CHANNEL-RANGE(...)
Specifies a set of channels to be detached.

Note
In the case of channels, the device code (MN) is the same as the hexadecimal channel path ID.

FROM = <x-text 2..2>
Specifies the channel path ID of the first channel in the set of channels to be detached.

TO = <x-text 2..2>
Specifies the channel path ID of the first (FROM) and last (TO) channel in the set of channels to be detached. The following rules apply: $chn_1\text{-}id$ < $chn_2\text{-}id$ and $chn_2\text{-}id$ - $chn_1\text{-}id$ ≤ 64, i.e. up to 64 channels can be detached at the same time.

SCOPE =
Specifies how the command is to be executed under VM2000.

SCOPE = *OWN-SYSTEM-ONLY
The command is only executed in the local system.

SCOPE = *VM2000-GLOBAL
If entered at the Monitor System (S server), the command is executed at all guest systems. If entered at another guest system or at the Monitor System (SQ server), the command is rejected with the message NKR0178.
The command is rejected if one of the guest systems is protected against global detachment (system parameter VMGIORAL=NO) and the channels that are to be detached do not yet possess the status DETACHED (unless FORCE= *UNCONDITIONAL-OFFLINE has been specified).

UNIT = *DEVICE-RANGE(...)
Specifies a number of up to 256 devices that are to be detached. The device type codes specified in the range do not have to be contiguous. If no device is generated for one of the device type codes, then processing continues with the next device type code that follows.

FROM = <alphanum-name 2..2> / <x-text 4..4>
Specifies the device type code of the first device in the range of devices to be detached.

TO = <alphanum-name 2..2> / <x-text 4..4>
Specifies the device type code of the last device in the range of devices to be detached.

UNIT = PUBSET-DEVICES(...)
Specifies that a pubset's disks are to be detached. The pubset must have been imported at least once. The mnemonic device codes (MNs) of the associated disks are administered in the system disk's SVL. This entry is made on IMPORT-PUBSET, EXPORT-PUBSET or MODIFY-PUBSET-PROCESSING.
As many disks as possible are always detached. If pubset disks are mirrored then the following should be noted:

– In the case of mirroring with DRV, both disks are detached.
– In the case of mirroring in external disk storage systems (see the "SHC-OSD" User Guide [37]) then only the standard disks (source or normal unit) are detached. If the mirror disks are to be detached, then the mirror disk (target or additional mirror unit) of the system disk (pubres) must be specified in the PUBSET operand.

PUBSET = list-poss(255): <cat-id 1..4> / *BY-PUBRES-DEVICE(...)
Designates the pubset whose disks are to be detached. The pubset can be specified via the catalog ID or the device code of its system disk.

PUBSET = <cat-id 1..4>
Specifies the pubset's catalog ID. A corresponding entry must exist in the MRS catalog. If no such entry exists then the disks can only be detached by specifying the system disk (see PUBSET=*BY-PUBRES-DEVICE).

PUBSET = *BY-PUBRES-DEVICE(...)
Specifies the pubset's system disk (pubres).

UNIT = list-poss(255): <alphanum-name 2..2> / <x-text 4..4>
Mnemonic device code (MN) of the pubres.

UNIT = list-poss(16): <alphanum-name 2..2> / <x-text 4..4>
Specifies the mnemonic device code (MN) if a controller or a device is to be detached.

FORCE =
Specifies the execution mode.

FORCE = *STD
The reconfiguration job is to be executed only if the unit is not being used. There is a maximum wait time of 15 minutes for release of the unit. During the wait time the unit is in the DETACH-PENDING state. If it is released within 15 minutes, the unit switches to the DETACHED state, otherwise it returns to the ATTACHED state.

FORCE = *YES
The reconfiguration job is to be executed immediately. This setting is not permitted for detaching CPU (see notes).

FORCE = *NO(...)
The reconfiguration job is only to be executed immediately if the unit is not being used. Otherwise the time specified in the WAIT operand must elapse before the unit is released. The device then switches to the DETACH-PENDING state. If the unit is released within the specified period, it switches to the DETACHED state; if not, it switches to the ATTACHED state.

WAIT =
Specifies the maximum wait time for execution of the reconfiguration job in execution mode FORCE=*NO.

WAIT = *NO
No maximum wait time is specified for execution of the reconfiguration job.

WAIT = *STD
15 minutes is set as the maximum wait time for execution of the reconfiguration job.

WAIT = <integer 1..32767>(...)
Specifies the maximum wait time.

DIM =
Governs whether the specified value for the wait time is to be interpreted as minutes or seconds.

DIM = *STD
Wait time as for DIM=*MIN.

DIM = *MIN
Specifies the maximum wait time in minutes.
Possible values: 1 ≤ <integer> ≤ 546

DIM = *SEC
Specifies the maximum wait time in seconds.
Possible values: 1 ≤ <integer> ≤ 32767

FORCE = *UNCONDITIONAL-OFFLINE
The reconfiguration job should always be performed immediately. Unlike FORCE=*YES, the hardware action associated with the job should always be executed immediately. This setting is only permitted for the detachment of channels (CHANNEL or CHANNEL-RANGE).
The protection of any of the guest systems against global detachment (system parameter VMGIORAL=NO) is ignored if this operand is specified.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
1	64	ETMRK..	Error in command execution
2	64	ETMRK..	Command processed partially without error
4	64	NKR0...	Hardware unit already detached
12	64	NKR0...	Internal check negative
16	64	NKR0...	Caller error
20	64	NKR0...	Software error

If there is an error in command termination, the maincode contains the message code of the message output during command processing.
The command return codes with the ETMRK.. maincodes occur only if the hardware unit to be reconfigured is a CPU.

Notes

- If the reconfiguration job cannot be executed with FORCE=*NO within the specified maximum wait time, it is rejected with the following message:

  ```
  NKR0037  mn MAY CURRENTLY NOT BE DETACHED
  NKR0049  mn DETACHMENT REJECTED
  ```

  In this case the operator should take one of the following actions:
  - Request more detailed information with the SHOW command, terminate all tasks which have reserved the device, or assign other devices.
  - Repeat the reconfiguration job in execution mode FORCE=*YES.
- Regardless of the execution mode set, a reconfiguration job is not executed if the unit affected is absolutely essential for the system. The system requires the following units:
  - the only or the last CPU ready for operation
  - the only or the last operator terminal
  - the only or the last controller for the public disks
  - a disk drive for a public disk.

- The “detach pending” state can be terminated by means of the appropriate command ATTACH-DEVICE or DETACH-DEVICE ..., FORCE=*YES.
- A CPU may only be detached with FORCE=*YES.
- The following steps are performed when detaching hardware units in the execution mode FORCE=*NO(...):

  a) if the command is permitted, message `NKR0092` is issued,

  b) if the unit is being used neither by the system nor by user jobs, the command is executed immediately,

  c) if the unit concerned is being used, the command is executed when the unit is no longer occupied. If execution is not possible within the period specified by WAIT, the following messages are issued:

  ```
  NKR0037  mn MAY CURRENTLY NOT BE DETACHED
  NKR0049  mn DETACHMENT REJECTED
  ```

  d) the command /DET UNIT=mn,FORCE=*NO(...) can be canceled by /ATT UNIT=mn
- For all devices, connections, etc., 2 alphanumeric characters are valid as ‘mn’.
  In the case of disk/tape devices and controllers, ‘mn’s comprising 4 hexadecimal characters (which must be specified in alphanumeric notation without 'X') are also valid. Restrictions see the “System Installation” manual [46].

Examples

Detach channels 41, 51 and 23 as soon as possible; wait state: 5 minutes

```
/DETACH *CH((41,51,23)),FORCE=*NO(WAIT=5(DIM=*MIN))
```

Immediate unconditional detachment of channels 30

```
/DETACH *CH((30)),FORCE=*UNCONDITIONAL-OFFLINE
```

Detaching the CPU 0

```
/DETACH-DEVICE UNIT=*CPU(CPU-IDENTIFIER=00)
/DET *CPU(00)
```

DETACH-GS-UNIT

Terminate use of GS unit

Description status:	GSMAN V19.0A
Functional area:	Caching media control Global storage administration
Domain:	DEVICE
Privileges:	TSOS

Function

The basic configuration of global storage (GS) is defined on the SVP. It includes the number of GS units available and how they are attached. All changes to the basic configuration must be made on the SVP, and they do not take effect until the system is restarted.
The systems of the S series (e.g. S150) in addition offer GS unit attachment and detachment functions which can be used to handle hardware faults and for maintenance purposes. The operating system makes these functions available to systems support personnel through the ATTACH- and DETACH-GS-UNIT reconfiguration commands.
In the event of serious hardware faults, a GS unit is implicitly detached by the operating system.

The applicability of the reconfiguration commands depends on the GS configuration (one or two GS units), on how they are currently being used and on the mode of GS operation in a HIPLEX MSCF network (see also the “HIPLEX MSCF” manual [25] or the section headed “The global storage (GS) medium” in the “Introduction to System Administration” [14]):

– In a network with shared GS, the nodes use GS as a shared resource (global GS operation).
– In a network without shared GS, individual nodes can use their GS locally (local GS operation).

In global GS operation in an XCS network the command applies to all the systems on the network. The GSMAN subsystem is then only available after XCS has been started.

Independently of the operating mode, it is also possible to disconnect the last GS unit following termination of all GS usage. This is only possible if all the systems in the XCS network are running BS2000/OSD ≥ V5.0. If any system in the XCS network is running BS2000/OSD < V5.0 then the command is rejected with the message EGC2003.

The following applies in all cases:

– The DETACH-GS-UNIT command deactivates the specified GS unit. This disconnects the GS units from all GS servers in the GS complex.

- A GS unit which is detached online while the second GS unit is in the ATTACHED state will from then on contain obsolete data. Since write jobs for dual partitions may be in progress during the detachment process, the data consistency of the detached GS unit cannot be guaranteed.
- The DETACH-GS-UNIT command will as a rule not be accepted if there are still partitions in use on the specified GS unit.
  An exception to this rule applies to dual partitions with the second GS unit still in the ATTACHED state. In this case, along with reduced data security, use of the dual partitions can if necessary be functionally restricted (e.g. DAB buffers) when switching to mono mode.

  Note

  > If a GS unit of a DAB dual partition is detached, then caching is either continued at the remaining GS unit or is restricted depending on the setting of the ACTION-AT-DUAL-GS-ERROR parameter in the DAB subsystem initialization file (see the section on error recovery in the "DAB" User Guide [5]). If the GS unit is re-attached then full scale caching is resumed.

- At system startup, the attached GS units are started up, and differences in the data they contain are eliminated. If there is only one GS unit attached, it will be started up even if it contains "obsolete" data.

The SHOW-GS-STATUS command allows systems support personnel to check which GS units are in operation.

Format

DETACH-GS-UNIT
GS-UNIT = <integer 1..2>

Operands

GS-UNIT = <integer 1..2>
Specifies the number of the GS unit which is to be detached.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
1	0	EGC2002	GS unit is already detached
	1	CMD0202	Syntax error
	32	CMD0221	Internal error
	64	CMD0216	No authorization to invoke the command
	64	EGC0005	Command aborted by user
	64	EGC0112	No GS available
	64	EGC2003	Last GS unit cannot be detached
	64	EGC2004	GS unit does not exist
	64	EGC2012	Partition in use
	64	EGC2013	Communication error
	64	EGC2014	SVP error
	64	EGC2016	System error
	128	EGC0010	GSMAN subsystem is not ready
	128	EGC0110	Command temporarily not executable

DISCONNECT-CMD-SERVER

Delete/modify entry in operator command table

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Operator function control
Domain:	not allocated
Privileges:	OPERATING
Routing code:	@

Function

The DISCONNECT-CMD-SERVER command deletes the link between an operator command and a command server running as a $CONSOLE application.

The DISCONNECT-CMD-SERVER command is only permitted for $CONSOLE applications and is only ever effective for the command-issuing $CONSOLE application.

Format

DISCONNECT**-CMD-SERVER**
CMD-NAME = <structured-name 1..30>

Operands

CMD-NAME = <structured-name 1..30>
Specifies the command name for which an entry in the command table is to be deleted. The command name may be up to 30 characters long and must comply with the naming convention for command names.
The entry with the highest priority, i.e. the entry assigned to the command-issuing $CONSOLE application, is deleted. If the command was entered in the list of operator commands using the CONNECT-CMD-SERVER command, the command name and all aliases are deleted from the list of operator commands when the last link is deleted.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
1	0	NBR1114	Link involved does not exist
	1	CMD0202	Syntax error

DISCONNECT-GS-SERVER

Terminate connection to the GS units of a complex

Description status:	GSMAN V19.0A
Functional area:	Global storage administration
Domain:	DEVICE
Privileges:	TSOS

Function

During reconfiguration, systems support can use the DISCONNECT-GS-SERVER command to terminate the connection between a GS server and the GS units in a GS complex. The command disconnects the local GS server from all the GS units in the GS complex.
The command is only accepted if the GS is not used by the local server. In the case of parallel HIPLEX, the command is only accepted if the XCS network does not consist of only one node at the time the command is issued.

In a VM2000 guest system which has assigned real GS and shared VM, the DISCONNECT-GS-SERVER command results in the hardware disconnection of the GS server. If the command is to be executed, all the guest systems at this GS server that have assigned real GS must not be using the GS.

The command calls any long-running background functions of the involved SVP. Other BS2000 functions that require the SVP (e.g. IOCCOPY or CPU and channel reconfiguration commands) should not be called at the same time. If possible, the BS2000 load should be reduced.

For further details on administering global storage, see the “Introduction to System Administration” [14].

Systems support can use the SHOW-GS-COMPLEX-CONFIGURATION command to obtain information on the current configuration of the GS complex.

Format

DISCONNECT-GS-SERVER

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
	1	CMD0202	Syntax error
	32	CMD0221	Internal error
	64	EGC0112	No GS available
	64	EGC2013	Communication error
	64	EGC2014	SVP error
	64	EGC2016	System error
	64	EGC2210	GS server not in GS complex
	64	EGC2212	GS server is already disconnected
	64	EGC2214	Other active GS servers in XCS
	64	EGC2215	XCS not ready
	64	EGC2216	Other active servers in XCS
	64	EGC2215	System error
	128	EGC0010	GSMAN subsystem not ready

EDIT-FILE-ATTRIBUTES

Start guided dialog for MODIFY-FILE-ATTRIBUTES

Description status: BS2000 OSD/BC V10.0A

Functional area: File processing

Domain: FILE

Privileges: STD-PROCESSING

Function

This command starts the guided dialog mechanism for the MODIFY-FILE-ATTRIBUTES command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified file.

Format

EDIT-FILE-ATTRIBUTES	Alias: **EDFA**
FILE-**NAME** = <filename 1..54>	

Operands

FILE-NAME = <filename 1..54>
Name of the file for which guided dialog is required. The operands are assigned the values specifically applicable to the named file.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error

In addition, all the return codes of the MODIFY-FILE-ATTRIBUTES and SHOW-FILE-ATTRIBUTES commands can returned.
If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-FILE-GENERATION-SUPPORT

Start guided dialog for MODIFY-FILE-GENERATION-SUPPORT

Description status: BS2000 OSD/BC V10.0A

Functional area: File processing

Domain: FILE-GENERATION-GROUP

Privileges: STD-PROCESSING

Function

This command starts the guided dialog mechanism for the MODIFY-FILE-GENERATION-SUPPORT command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified file generation.

Format

EDIT-FILE-GENERATION-SUPPORT
GENERATION-**NAME** = <filename 1..54 without-vers>

Operands

GENERATION-NAME = <filename 1..54 without-vers>
Name of the file generation for which guided dialog is required. The operands are assigned the values specifically applicable to the named file generation.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error

In addition, all the return codes of the MODIFY-FILE-GENERATION-SUPPORT and SHOW-FILE-ATTRIBUTES commands can returned.
If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-FILE-GROUP-ATTRIBUTES

Start guided dialog for MODIFY-FILE-GROUP-ATTRIBUTES

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE-GENERATION-GROUP
Privileges:	STD-PROCESSING

Function

This command starts the guided dialog mechanism for the MODIFY-FILE-GROUP-ATTRIBUTES command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified file generation group.

Format

EDIT-FILE-GROUP-ATTRIBUTES
GROUP**-NAME** = <filename 1..47 without-gen-vers>

Operands

GROUP-NAME = <filename 1..47 without-gen-vers>
Name of the file generation group for which guided dialog is required. The operands are assigned the values specifically applicable to the named file generation group.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error

In addition, all the return codes of the MODIFY-FILE-GROUP-ATTRIBUTES and SHOW-FILE-ATTRIBUTES commands can returned.
If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-FILE-LINK

Start guided dialog for ADD-FILE-LINK

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING

Function

This command starts the guided dialog mechanism for the ADD-FILE-LINK command. Where technically possible and helpful, the predefined default operand values are each replaced by values from the TFT entry corresponding to the specified link name.
This command does not cause the existing TFT entry to be modified; it creates a new entry and implicitly releases the “old” TFT entry (see the ADD-FILE-LINK command).
The command is rejected if there is no TFT entry for the specified link name.

Format

EDIT-FILE-LINK	Alias: **EDFL**
LINK-NAME = <filename 1..8 without-gen>	

Operands

LINK-NAME = <filename 1..8 without-gen>
Link name for guided dialog is required.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error

In addition, all the return codes of the ADD-FILE-LINK and SHOW-FILE-LINK commands can returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-JOB

Start guided dialog for MODIFY-JOB

Description status: BS2000 OSD/BC V10.0A

Functional area: Job processing

Domain: JOB

Privileges:
STD-PROCESSING
OPERATING
SAT-FILE-EVALUATION
SAT-FILE-MANAGEMENT
SECURITY-ADMINISTRATION

Function

This command starts the guided dialog for the MODIFY-JOB command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified job.

Format

```
EDIT-JOB                                                        Alias: EDJ

JOB-IDENTIFICATION = *TSN(...) / *MONJV(...)

  *TSN(...)

   ⏐  TSN = <alphanum-name 1..4>

  *MONJV(...)

   ⏐  MONJV = <filename 1..54 without-gen-vers>
```

Operands

JOB-IDENTIFICATION =
Type of job allocation. Jobs can be identified by their TSN or monitoring JV.

JOB-IDENTIFICATION = *TSN(...)

TSN = <alphanum-name 1..4>
Task serial number of the required job.

JOB-IDENTIFICATION = *MONJV(...)

MONJV = <filename 1..54 without-gen-vers>
Name of the JV which monitors the required job.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-JOB and SHOW-JOB-STATUS commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-JOB-OPTIONS

Start guided dialog for MODIFY-JOB-OPTIONS

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing
Domain:	JOB
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE OPERATING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION TSOS

Function

This command starts the guided dialog for the MODIFY-JOB-OPTIONS command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified job.

Format

```
EDIT-JOB-OPTIONS                                                    Alias: EDJO

JOB-IDENTIFICATION = *OWN / *TSN(...) / *MONJV(...)

   *TSN(...)
     |  TSN = <alphanum-name 1..4>

   *MONJV(...)
     |  MONJV = <filename 1..54 without-gen-vers>
```

Operands

JOB-IDENTIFICATION =

This operand is available only to privileged users.

Specifies the job for which job monitoring is to be modified. The job can be identified by the TSN or by the job variable which monitors it.

JOB-IDENTIFICATION = *OWN
Job monitoring is to be modified for the user’s own task. The settings for INFORMATION-LEVEL, OPERATOR-INTERACTION, SYSLST-LIMIT and LOGGING can only be modified for the user’s own task.

JOB-IDENTIFICATION = *TSN(...)

TSN = <alphanum-name 1..4>
Task serial number of the required job.

JOB-IDENTIFICATION = *MONJV(...)

MONJV = <filename 1..54 without-gen-vers>
Name of the JV which monitors the required job.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-JOB-OPTIONS, SHOW-JOB-OPTIONS and SHOW-JOB-STATUS commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-JV

Start guided dialog for MODIFY-JV

Description status:	JV V15.1A
Functional area:	Job variables
Domain:	JOB-VARIABLES
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE OPERATING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

This function is only available to the user if the chargeable software product JV has been loaded as a subsystem.

Function

This command starts the guided dialog for the MODIFY-JV command. Where technically possible (a JV can contain up to 256 characters, but MODIFY-JV can only set a maximum of 254), the current content of the JV is predefined as the new content and can thus be edited.

Format

```
EDIT-JV                                                                         Alias: EDJV

 JV-NAME = <filename 1..54 without-gen-vers>
,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL
,PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647>
```

Operands

JV-NAME = <filename 1..54 without-gen-vers>
Name of the JV for which guided dialog is required. The SET-VALUE operand is assigned the current JV content.

OUTPUT-FORMAT =
Determines the output format.

OUTPUT-FORMAT = *CHARACTER
Output in character format.

OUTPUT-FORMAT = *HEXADECIMAL
Output in hexadecimal format.

PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647>
Read or write password of the JV. The PASSWORD operand has the following special characteristics:

- The input field is automatically blanked out in the guided dialog.
- The password entered is not logged.

PASSWORD = *NONE
The JV has no password or the password was already specified in the ADD-PASSWORD command.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-JV and SHOW-JV -ATTRIBUTES commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-JV-ATTRIBUTES

Start guided dialog for MODIFY-JV-ATTRIBUTES

Description status:	JV V15.1A
Functional area:	Job variables
Domain:	JOB-VARIABLES
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

This function is only available to the user if the chargeable software product JV has been loaded as a subsystem.

Function

This command starts the guided dialog for the MODIFY-JV-ATTRIBUTES command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified JV.

Format

EDIT-JV-ATTRIBUTES	Alias: **EDJVA**
JV-NAME = <filename 1..54 without-gen-vers>	

Operands

JV-NAME = <filename 1..54 without-gen-vers>
Name of the JV for which guided dialog is required. The operands are assigned the values specifically applicable to the named JV.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-JV-ATTRIBUTES and SHOW-JV-ATTRIBUTES commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-MASTER-CATALOG-ENTRY

Start guided dialog for MODIFY-MASTER-CATALOG-ENTRY

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Pubset and MRSCAT management
Domain:	MULTI-CATALOG-AND-PUBSET-MGMT
Privileges:	TSOS

Function

This command starts the guided dialog for the MODIFY-MASTER-CATALOG-ENTRY command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified pubset.

Format

EDIT-MASTER-CATALOG-ENTRY
PUBSET = <cat-id 1..4>

Operands

PUBSET = <cat-id 1..4>
Identifies the pubset for which guided dialog is required. The operands are assigned the values specifically applicable to the named pubset.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-MASTER-CATALOG-ENTRY and SHOW-MASTER-CATALOG-ENTRY commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-POSIX-USER-ATTRIBUTES

Start guided dialog for MODIFY-POSIX-USER-ATTRIBUTES

Description status:	SRPMNUC V19.0A
Functional area:	User management POSIX administration and application
Domain:	USER-ADMINISTRATION
Privileges:	STD-PROCESSING POSIX-ADMINISTRATION USER-ADMINISTRATION

Function

This command starts the guided dialog for the MODIFY-POSIX-USER-ATTRIBUTES command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified user.

Format

EDIT-POSIX-USER-ATTRIBUTES
USER-IDENTIFICATION = <name 1..8>
,**PUB**SET = *HOME / <cat-id 1..4>

Operands

USER-IDENTIFICATION = <name 1..8>
BS2000 user ID whose POSIX user attributes are to be modified.

PUBSET = *HOME / <cat-id 1..4>
Pubset in whose user catalog the POSIX user attributes are to be modified.

PUBSET = *HOME
The change is made on the home pubset.

PUBSET = <cat-id 1..4>
The change is made on the specified pubset.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-POSIX-USER-ATTRIBUTES and SHOW-POSIX-USER-ATTRIBUTES commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P command return codes are supplied.

EDIT-POSIX-USER-DEFAULTS

Start guided dialog for MODIFY-POSIX-USER-DEFAULTS

Description status:	SRPMNUC V19.0A
Functional area:	User management POSIX administration and application
Domain:	USER-ADMINISTRATION
Privileges:	STD-PROCESSING POSIX-ADMINISTRATION USER-ADMINISTRATION

Function

This command starts the guided dialog for the MODIFY-POSIX-USER-DEFAULTS command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified pubsest.

Format

EDIT-POSIX-USER-DEFAULTS
PUBSET = ***HOME** / <cat-id 1..4>

Operands

PUBSET = *HOME / <cat-id 1..4>
Pubset in whose user catalog the POSIX default user attributes are to be modified.

PUBSET = *HOME
The change is made on the home pubset.

PUBSET = <cat-id 1..4>
The change is made on the specified pubset.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-POSIX-USER-DEFAULTS and SHOW-POSIX-USER-DEFAULTS commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-PUBSET-CACHE-ATTRIBUTES

Start guided dialog for MODIFY-PUBSET-CACHE-ATTRIBUTES

Description status: BS2000 OSD/BC V10.0A

Functional area: Caching media control
Pubset and MRSCAT management

Domain: MULTI-CATALOG-AND-PUBSET-MGMT

Privileges: TSOS

Function

This command starts the guided dialog for the MODIFY-PUBSET-CACHE-ATTRIBUTES command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified pubset.

Format

```
EDIT-PUBSET-CACHE-ATTRIBUTES

 PUBSET = <cat-id 1..4>
,PUBSET-TYPE = *SINGLE-FEATURE / *SYSTEM-MANAGED(...)
   *SYSTEM-MANAGED(...)
     |   VOLUME-SET = <cat-id 1..4>
```

Operands

PUBSET = <cat-id 1..4>
Identifies the pubset for which guided dialog is required. The operands are assigned the values specifically applicable to the named pubset.

PUBSET-TYPE = *SINGLE-FEATURE / *SYSTEM-MANAGED(...)
Specifies the type of pubset for which a new entry is to be created in the MRSCAT.

PUBSET-TYPE = *SINGLE-FEATURE
The pubset is an SF pubset.

PUBSET-TYPE = *SYSTEM-MANAGED(...)
The pubset is an SM pubset. The PFA cache configuration is defined or modified for the volume set specified next.

VOLUME-SET = <cat-id 1..4>
Identifies the volume set.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error
	64	DMS140C	Pubset type conflict Wrong pubset type specified

In addition, all the return codes of the MODIFY-PUBSET-CACHE-ATTRIBUTES and SHOW-PUBSET-CACHE-ATTRIBUTES commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-PUBSET-DEFINITION-FILE

Start guided dialog for MODIFY-PUBSET-DEFINITION-FILE

Description status: BS2000 OSD/BC V10.0A

Functional area: Pubset and MRSCAT management

Domain: MULTI-CATALOG-AND-PUBSET-MGMT

Privileges: TSOS

Function

This command starts the guided dialog for the MODIFY-PUBSET-DEFINITION-FILE command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified pubset.

Format

EDIT-PUBSET**-DEF**INITION**-FILE**
PUBSET = <cat-id 1..4> ,**VOL**UME**-SET** = <cat-id 1..4>

Operands

PUBSET = <cat-id 1..4>
Identifies the pubset for which guided dialog is required. The operands are assigned the values specifically applicable to the named pubset.

VOLUME-SET = <cat-id 1..4>
Identifies the volume set.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error
	64	DMS1486	Pubset is not a system-managed pubset

In addition, all the return codes of the MODIFY-PUBSET-DEFINITION-FILE and SHOW-PUBSET-DEFINITION-FILE commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P command return codes are returned.

EDIT-PUBSET-SPACE-DEFAULTS

Start guided dialog for MODIFY-PUBSET-SPACE-DEFAULTS

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Pubset and MRSCAT management System control and optimization
Domain:	MULTI-CATALOG-AND-PUBSET-MGMT
Privileges:	TSOS

Function

This command starts the guided dialog for the MODIFY-PUBSET-SPACE-DEFAULTS command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified pubset .

Format

EDIT-PUBSET**-SPACE-DEF**AULTS
PUBSET = <cat-id 1..4>
,**SCOPE** = ***TEMP**ORARY / ***PERMANENT** / ***NEXT-PUB**SET**-SESSION**

Operands

PUBSET = <cat-id 1..4>
Identifies the pubset for which guided dialog is required. The operands are assigned the values specifically applicable to the named pubset.

SCOPE = *TEMPORARY / *PERMANENT / *NEXT-PUBSET-SESSION
Defines how long the setting applies and when it comes into effect.

SCOPE = *TEMPORARY
The setting takes immediate effect and applies only for the duration of the current pubset session, i.e. until the pubset is taken out of service.

SCOPE = *PERMANENT
The setting takes immediate effect and applies until the next modification.

SCOPE = *NEXT-PUBSET-SESSION
The setting applies until the next permanent modification. However, the setting does not take effect until the start of the next pubset session.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-PUBSET-SPACE-DEFAULTS and SHOW-PUBSET-SPACE-DEFAULTS commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-SPACE-SATURATION-LEVELS

Start guided dialog for MODIFY-SPACE-SATURATION-LEVELS

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Pubset and MRSCAT management System control and optimization
Domain:	MULTI-CATALOG-AND-PUBSET-MGMT
Privileges:	TSOS

Function

This command starts the guided dialog for the MODIFY-SPACE-SATURATION-LEVELS command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified pubset or volume set.

Format

```
EDIT-SPACE-SATURATION-LEVELS

 PUBSET = <cat-id 1..4>
,PUBSET-TYPE = *SINGLE-FEATURE / SYSTEM-MANAGED(...)
   SYSTEM-MANAGED(...)
     |  VOLUME-SET = <cat-id 1..4>
,SCOPE = *TEMPORARY / *PERMANENT / *NEXT-PUBSET-SESSION
```

Operands

PUBSET = <cat-id 1..4>
Identifies the pubset for which guided dialog is required. The operands are assigned the values specifically applicable to the named pubset.

PUBSET-TYPE = <u>*SINGLE-FEATURE</u> / *SYSTEM-MANAGED(...)
Specifies whether the pubset is an SF or SM pubset.

PUBSET-TYPE = <u>*SINGLE-FEATURE</u>
The pubset is an SF pubset. The thresholds are defined on a pubset-global basis.

PUBSET-TYPE = *SYSTEM-MANAGED(...)
The pubset is an SM pubset. The thresholds are defined for the volume set specified next.

VOLUME-SET = <cat-id 1..4>
Identifies the volume set.

SCOPE = *TEMPORARY / *PERMANENT / *NEXT-PUBSET-SESSION
Defines how long the setting applies and when it comes into effect.

SCOPE = *TEMPORARY
The setting takes immediate effect and applies only for the duration of the current pubset session, i.e. until the pubset is taken out of service.

SCOPE = *PERMANENT
The setting takes immediate effect and applies until the next modification.

SCOPE = *NEXT-PUBSET-SESSION
The setting applies until the next permanent modification. However, the setting does not take effect until the start of the next pubset session.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error
	64	DMS140C	Pubset type conflict: Wrong pubset type specified

In addition, all the return codes of the MODIFY-SPACE-SATURATION-LEVELS and SHOW-SPACE-SATURATION-LEVELS commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-STORAGE-CLASS

Start guided dialog for MODIFY-STORAGE-CLASS

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Pubset and MRSCAT management
Domain:	MULTI-CATALOG-AND-PUBSET-MGMT
Privileges:	TSOS

Function

This command starts the guided dialog for the MODIFY-STORAGE-CLASS command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the storage class.

Format

EDIT-STORAGE-**CLASS**
STORAGE-**CLASS-NAME** = <composed-name 1..8> **,PUB**SET = <cat-id 1..4>

Operands

STORAGE-CLASS-NAME = <composed-name 1..8>
Name of the storage class for which guided dialog is required. The operands are assigned the values specifically applicable to the named storage class.

PUBSET = <cat-id 1..4>
Identifies the pubset for which the storage class was created. The pubset must have been imported to the local system (in exclusive or shared mode).

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-STORAGE-CLASS and SHOW-STORAGE-CLASS commands can returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-USER-ATTRIBUTES

Start guided dialog for MODIFY-USER-ATTRIBUTES

Description status:	SRPMNUC V19.0A
Functional area:	User management
Domain:	USER-ADMINISTRATION
Privileges:	STD-PROCESSING USER-ADMINISTRATION

Function

This command starts the guided dialog for the MODIFY-USER-ATTRIBUTES command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified user.

Format

EDIT-USER-ATTRIBUTES
USER-IDENTIFICATION = <name 1..8> ,**PUB**SET = ***HOME** / <cat-id 1..4>

Operands

USER-IDENTIFICATION = <name 1..8>
Name of the user ID whose entry is to be modified.

PUBSET = *HOME / <cat-id 1..4>
(SM) pubset in whose user catalog the pubset-specific user attributes are to be modified.

PUBSET = *HOME
The change is made on the home pubset.

PUBSET = <cat-id 1..4>
The change is made on the specified pubset.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-USER-ATTRIBUTES and SHOW-USER-ATTRIBUTESS commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P return codes are supplied.

EDIT-USER-PUBSET-ATTRIBUTES

Start guided dialog for MODIFY-USER-PUBSET-ATTRIBUTES

Description status:	SRPMNUC V19.0A
Functional area:	User management Pubset and MRSCAT management
Domain:	USER-ADMINISTRATION
Privileges:	STD-PROCESSING USER-ADMINISTRATION

Function

This command starts the guided dialog for the MODIFY-USER-PUBSET-ATTRIBUTES command. Where technically possible and helpful, the predefined default operand values are each replaced by values currently applicable to the specified user.

Format

EDIT-USER-PUBSET-ATTRIBUTES
USER-IDENTIFICATION = <name 1..8> ,PUBSET = *HOME / <cat-id 1..4>

Operands

USER-IDENTIFICATION = <name 1..8>
Name of the user ID whose pubset-specific user attributes are to be modified.

PUBSET = *HOME / <cat-id 1..4>
(SM) pubset in whose user catalog the pubset-specific user attributes are to be modified.

PUBSET = *HOME
The change is made on the home pubset

PUBSET = <cat-id 1..4>
The change is made on the specified pubset.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error

In addition, all the return codes of the MODIFY-USER-PUBSET-ATTRIBUTES and SHOW-USER-ATTRIBUTES commands can be returned. If errors occur while S variables are being processed, the corresponding SDF-P command return codes are returned.

ELSE

Initiate ELSE branch

Description status:	SDF-P-BASYS V2.5E
Functional area:	Procedures
Domain:	PROCEDURE
Privileges:	STD-PROCESSING OPERATING HARDWARE-MAINTENANCE SECURITY-ADMINISTRATION SAT-FILE-MANAGEMENT SAT-FILE-EVALUATION

Function

ELSE is an SDF-P control flow command and serves to initiate the ELSE branch in IF blocks. The ELSE branch (the commands between ELSE and END-IF) is processed if the test fails to meet any of the conditions set by the IF command or any ELSE-IF commands that are included (ELSE-IF command are chargeable).

In an IF-BLOCK-ERROR block or a (chargeable) IF-CMD-ERROR block, the ELSE branch is taken is no error occurs.

Restrictions

Users with SECURITY-ADMINISTRATION, SAT-FILE-EVALUATION or SAT-FILE-MANAGEMENT privilege can use the command in procedures only.

Format

ELSE

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
	1	CMD0202	Syntax error
	1	SDP0118	Command in incorrect context
	1	SDP0223	Incorrect environment
	3	CMD2203	Incorrect syntax file
	32	CMD0221	System error (internal error)
	64	SDP0091	Semantic error
	130	SDP0099	No further address space available

Example

See IF command.

ENCRYPT-FILE

Encrypt an unencrypted file

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING TSOS SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT

Use of this command makes sense only when one of the chargeable software products openCRYPT-SERV or openCRYPT-SOFT is available for performing encryption.

Function

The ENCRYPT-FILE command converts an unencrypted file into an encrypted file. The encryption method used for conversion is set using the system parameter FILECRYP and is stored in the catalog entry (see the *ENCRYPTION* output field in the SHOW-FILE-ATTRIBUTES command).

Within a file generation group either all generations are unencrypted or all generations except tape generations are encrypted. Within a file generation group all encrypted generations have the same crypto password and the same encryption method.

When a file is converted into an encrypted file the read and execute passwords are implicitly deleted, but the write password remains unchanged. With files that are already encrypted the assignment of read and execute passwords is ignored.

Encrypted files are protected against unauthorized decryption:
The encrypted file contents can only be accessed if the user knows the crypto password specified for encryption. The relevant crypto password must be entered in the crypto password table (see the ADD-CRYPTO-PASSWORD command) both for explicit decryption (DECRYPT-FILE command) and for implicit decryption while a file is being accessed or processed.

File encryption does not, however, include greater protection against deletion, overwriting or destruction of the file contents. File encryption can thus not replace data protection and security.

Format

```
ENCRYPT-FILE

 FILE-NAME = <filename 1..54 without-gen>
,CRYPTO-PASSWORD = <c-string 1..8> / <x-string 1..16> / *SECRET / *FROM-FILE(...)
    *FROM-FILE(...)
      |  FILE-NAME = <filename 1..54 without-gen>
,CONFIRM-PASSWORD = *NOT-SPECIFIED / <c-string 1..8> / <x-string 1..16> / *SECRET
```

Operands

FILE-NAME = <filename 1..54 without-gen>
Name of the file to be converted. It must satisfy the following requirements:

- It must already have a catalog entry.
- The pubset on which it is cataloged must be accessible locally.
- It may not yet be encrypted.
- It may not reside on a private disk.
- No tape type may be entered.

CRYPTO-PASSWORD = <c-string 1..8> / <x-string 1..16> / *SECRET / *FROM-FILE(...)
Crypto password for protecting against unauthorized decryption. To permit access to the unencrypted file content this password must be entered in the job's crypto password table. The operand has the following special characteristics:

- The password entered is not logged.
- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

CRYPTO-PASSWORD = <c-string 1..8> / <x-string 1..16>
Direct specification of the crypto password. If the system parameter FREFCRYP contains a user ID, direct specification of a crypto password is permitted only for files of this user ID. Only a reference file may be specified for files of other user IDs (CRYPTO-PASSWORD=*FROM-FILE).

CRYPTO-PASSWORD = *FROM-FILE(...)
The crypto password is to be taken over from a reference file. If the password has not been entered in the job's crypto password table (with ADD-CRYPTO-PASSWORD), it must be confirmed in the CONFIRM-PASSWORD operand.

FILE-NAME = <filename 1..54 without-gen>
Reference file from which the crypto password is taken over.
The pubset on which the reference file is cataloged must be locally accessible.

CONFIRM-PASSWORD = *NOT-SPECIFIED / <c-string 1..8> / <x-string 1..16> / *SECRET
Confirmation of the password entry to guard agains typos.
The CONFIRM-PASSWORD operand has the following special features:

- The value entered is not logged.
- In guided dialog the entry field is automatically blanked out.
- If *SECRET or ^ is specified, in unguided dialog and in foreground procedures SDF provides a non-displaying entry field for concealed entry of the password.

The default is *NOT-SPECIFIED, i.e. the password need not be acknowledged. This operand value is, however, only allowed if the crypto password is taken over from a reference file and this crypto password is entered in the job's crypto password table (with ADD-CRYPTO-PASSWORD or temporarily with ENCRYPT-FILE).

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed
2	0	DMS05B6	Time conversion UTC to LT errored
2	0	DMS05F5	Defective blocks were not copied
	1	CMD0202	Syntactical or semantic error in the command
	32	DMS0584	During processing a status was reported which prevents the function from being continued.
	32	DMS05C7	Unexpected internal error in DMS
	64	CMD0216	Privileges error
	64	DMS0512	Requested catalog not found
	64	DMS051B	Requested user ID not in pubset Guaranteed message: DMS051B
	64	DMS051C	User does not have access right for pubset Guaranteed message: DMS051C
	64	DMS0535	Specified file not shareable
	64	DMS057C	Processing not possible because of HSMS error
	64	DMS057E	File migrated, HSMS not available
	64	DMS0585	An error was detected during catalog processing or during multiprocessor processing.
	64	DMS0588	Disk storage could not be assigned

(Part 1 of 2)

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	64	DMS05FC	Specified user ID not in the home pubset
	64	DMS0609	Access to system file not possible
	64	DMS060D	Invalid file name for reference file
	64	DMS0663	File encryption not permitted
	64	DMS0667	File cannot be used as reference file
	64	DMS0669	Protection attribute modified implicitly
	64	DMS066A	Crypto password cannot be used
	64	DMS066D	Crypto password entry restricted
	64	DMS0681	DMS error while executing job
	64	DMS0684	File does not exist
	64	DMS0691	Crypto password list has reached maximum size
	64	DMS0692	Maximum number of crypto passwords per task has been reached
	64	DMS06B5	File open or catalog entry not updated after system error
	130	DMS0524	System address space exhausted
	130	DMS0582	The file is currently locked or in use and cannot be processed
	130	DMS0585	An error was detected during catalog processing or during multiprocessor processing.
	130	DMS0588	Disk storage could not be assigned
	130	DMS0594	Insufficient virtual memory available

(Part 2 of 2)

END-CJC-ACTION

Mark end of CJC command sequence

Description status:	JV V15.1A
Functional area:	Job variables
Domain:	JOB-VARIABLES
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

This function is available to the user only if the chargeable software product JV has been loaded as a subsystem.

Function

The END-CJC-ACTION command is the last command in a CJC command sequence. As soon as it is issued, the ADD-CJC-ACTION command becomes effective, i.e. the specifications contained in it are evaluated.
(For structure and execution of the CJC command sequences, and also notes, see the ADD-CJC-ACTION command description.)

Format

END-CJC-ACTION

Return codes

(SC2)	SC1	Maincode	Meaning
0	1	CMD0202	No CJC command sequence was initiated

Note

Command return codes referring to a CJC command sequence are not returned until after completion of the CJC command sequence with the END-CJC-ACTION command. Possible values are indicated in the description of the ADD-CJC-ACTION command.

END-IF

Mark end of IF command block

Description status:	SDF-P-BASYS V2.5E
Functional area:	Procedures
Domain:	PROCEDURE
Privileges:	STD-PROCESSING OPERATING HARDWARE-MAINTENANCE SECURITY-ADMINISTRATION SAT-FILE-MANAGEMENT SAT-FILE-EVALUATION

Function

END-IF is an SDF-P control flow command and serves to terminate a command block initiated by means of an IF, IF-BLOCK-ERROR or IF-CMD-ERROR command. Procedure execution then continues with the command that follows END-IF.

Restrictions

Users with SECURITY-ADMINISTRATION, SAT-FILE-EVALUATION or SAT-FILE-MANAGEMENT privilege can use the command in procedures only.

Format

END-IF
BLOCK = ***LAST** / <structured-name 1..255>

Operands

BLOCK = *LAST / <structured-name 1..255>
This terminates the command block initiated by means of an IF, IF-BLOCK-ERROR or IF-CMD-ERROR command.
The default value is *LAST, i.e. the last command block to have been initiated is now terminated.
When a block name is specified, SDF-P checks whether this name was assigned to the last command block to have been initiated. If the specified name and the label do not tally, SDF-P rejects the END-IF command with an error message. Specifying a block name is therefore useful for consistency checking and for documentation.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
	1	CMD0202	Syntax error
	1	SDP0118	Command in incorrect context
	1	SDP0223	Incorrect environment
	3	CMD2203	Incorrect syntax file
	32	CMD0221	System error (internal error)
	64	SDP0091	Semantic error
	130	SDP0099	No further address space available

END-PARAMETER-DECLARATION

Mark end of parameter declarations

Description status:	SDF-P-BASYS V2.5E
Functional area:	Procedures
Domain:	PROCEDURE
Privileges:	STD-PROCESSING OPERATING HARDWARE-MAINTENANCE SECURITY-ADMINISTRATION SAT-FILE-MANAGEMENT SAT-FILE-EVALUATION

Function

END-PARAMETER-DECLARATION is an SDF-P control flow command and serves to terminate the parameter declarations initiated by means of the BEGIN-PARAMETER-DECLARATION command. It is the last command in the procedure head.

Format

END-PARAMETER-**DECL**ARATION

Return codes

The END-PARAMETER-DECLARATION command can only be used as the last command in the procedure head of an S procedure. SDF-P detects errors in the procedure head during pre-analysis and terminates the procedure call. The command return codes can only occur if the command is used outside the procedure head.

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
	1	CMD0202	Syntax error
	1	SDP0118	Command in incorrect context
	1	SDP0139	Maximum number of backward jumps reached
	3	CMD2203	Incorrect syntax file
	32	CMD0221	System error (internal error)

(Part 1 of 2)

(SC2)	SC1	Maincode	Meaning
	64	SDP0091	Semantic error
	130	SDP0099	No further address space available

(Part 2 of 2)

END-PROCEDURE

Mark end of non-S procedure file

Description status:	SYSFILE V19.0A
Functional area:	Procedures
Domain:	PROCEDURE
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The END-PROCEDURE command must appear as the last record in a non-S procedure file. It terminates execution of the procedure and causes processing to return to the point of interruption in the calling procedure - if there was one; otherwise it causes a return to primary command input (the terminal in interactive mode, ENTER file in batch mode). The system files, including TASKLIB, are given the allocations in force at the time of interruption. A file that was assigned by means of ASSIGN-SYSLST in the procedure is closed with END-PROCEDURE.

The END-PROCEDURE command may be used in procedure files only and therefore does not appear in the menu.

Format

END-PROCEDURE

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
2	0	SSM2039	Error on closing output system file; the SYSOUT message contains the DMS error code as an insert
	64	SSM1013	No procedure has been called

Notes

- Commands in procedure files:

  If an error in the command sequence activates the spin-off mechanism, processing branches to the next of the following commands:
  END-PROCEDURE, LOGOFF, EXIT-JOB, SET-JOB-STEP, CANCEL-PROCEDURE or EXIT-PROCEDURE.

- The effect of an END-PROCEDURE command in ESCAPE mode is described under the HOLD-PROCEDURE command (figure 7 on page 3-406).

- If a procedure is called in ESCAPE mode, the following message is issued when control is returned:
  TASK IS IN ESCAPE-MODE AT LEVEL NUMBER i

  where i = number of the procedure level to which processing has returned.
  When primary command input is reached (level 0), this message is not issued.

Examples

See the BEGIN-PROCEDURE, CANCEL-PROCEDURE, HOLD-PROCEDURE and RESUME-PROCEDURE commands.

ENDP-RESUME

Terminate non-S procedure run and resume loaded program

Description status:	SYSFILE V19.0A
Functional area:	Procedures
Domain:	PROCEDURE SDF
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The ENDP-RESUME command terminates execution of a non-S procedure and resumes a program loaded during the procedure.
In S procedures, the EXIT-PROCEDURE command with RESUME-PROGRAM=*YES must be used for this function.

Method of operation

ENDP-RESUME is used when a program call is to be implemented as a command, e.g. a call to the PROG.EDIT problem program:

/START-EXE-PROG FROM-FILE=PROG.EDIT is to be implemented as the command /X-EDIT:

– In a user syntax file a command must, for example, be defined under the name X-EDIT and implemented as a procedure (SDF-A statement ADD-COMMAND; for further details see the “SDF-A” manual [33]). The path name of the procedure file is also defined in the process.

– The user syntax file must be activated (MODIFY-SDF-OPTIONS command). The user must be authorized to execute the procedure file. The procedure file may have the following contents:

```
/BEGIN-PROC
/LOAD-EXEPROG FROM-FILE=PROG.EDIT
/ENDP-RESUME
```

Format

ENDP-RESUME

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
2	0	SSM2039	Error on closing output system file; the SYSOUT message contains the DMS error code as an insert
	1	CMD0202	Syntax error
	3	CMD2203	Incorrect syntax file
	32	CMD0221	System error (internal error)
	64	SSM1013	No procedure has been called
	130	SDP0099	No further address space available

ENTER-JOB

Start command sequence (ENTER file) as batch job

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing
Domain:	JOB
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE OPERATING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION
Routing code:	P

Function

The ENTER-JOB command enables another batch job (ENTER job) to be initiated from an interactive or batch job. This new batch job is independent of the issuing job and is assigned its own task sequence number (TSN) by the system. The batch job is also called an ENTER job (see figure). The former term ENTER procedure, on the other hand, is regarded as obsolete because it is too reminiscent of the ENTER-PROCEDURE command. However, this command expects a different file format.

Restrictions

Users with SECURITY-ADMINISTRATION, SAT-FILE-EVALUATION or SAT-FILE-MANAGEMENT privilege can use the command in procedures only.

Privileged functions

An operator can start the ENTER file on the console with the job attributes which are specified in the SET-LOGON-PARAMETERS command of the ENTER file (see default values *STD) or modify or add job attributes. This special function is linked to the consoles and user IDs with the OPERATING privilege (see also the DEFAULT-FROM-FILE operand, page 3-242). If the command is issued from a user ID which is not equipped with the OPERATING privilege, job attributes from the SET-LOGON-PARAMETERS command of the ENTER file are ignored.

Method

The commands for the ENTER job must be stored in a file. The first command of such an ENTER file must always be SET-LOGON-PARAMETERS, and the last EXIT-JOB or LOGOFF. The name of the file must be specified in the ENTER-JOB command. When the command is accepted, a message containing the task sequence number (TSN) which assigned the system to the new ENTER job is issued as the job confirmation. The ENTER job does not have to start immediately as the operator or systems support can restrict the number and runtime of the batch jobs in the system. If this is the case, the ENTER job is entered in the job queue and remains there until it can be processed by the system.

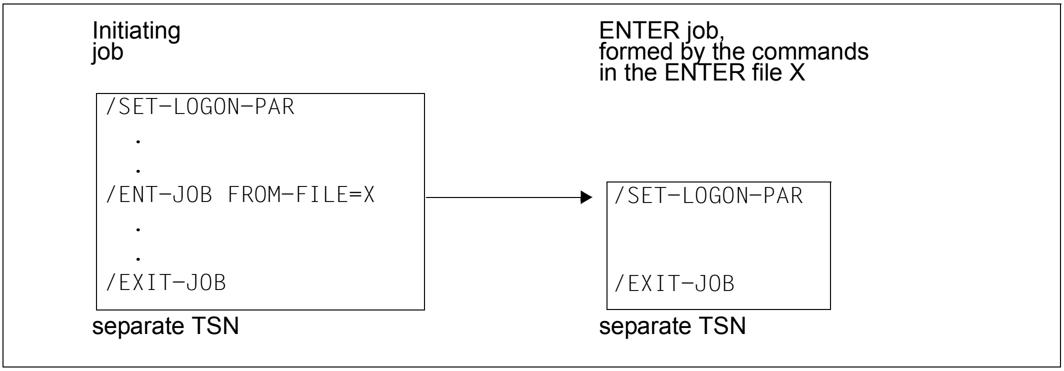


Figure 4: ENTER-JOB starts ENTER job

Format

```
ENTER-JOB                                                                        Alias: ENJ

FROM-FILE = *LIBRARY-ELEMENT(...) / <filename 1..54 without-gen>
  *LIBRARY-ELEMENT(...)
    |  LIBRARY = <filename 1..51 without-gen>
    | ,ELEMENT = <composed-name 1..38>
,PROCESSING-ADMISSION = *STD / *PARAMETERS(...) / *FROM-CALLER(...)
  *PARAMETERS(...)
    |  USER-IDENTIFICATION = *NONE / <name 1..8>
    | ,ACCOUNT = *NONE / <alphanum-name 1..8>
    | ,PASSWORD = *NONE / <c-string 1..8> / <c-string 9..32> / <x-string 1..16> / *SECRET
  *FROM-CALLER(...)
    |  ACCOUNT = *STD / <alphanum-name 1..8>
,DEFAULT-FROM-FILE = *STD / *NO / *YES
,FILE-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / *SECRET
    | ,CRYPTO-PASSWORD = *NONE / <c-string 1..8> / <x-string 1..16> / *SECRET
,DELETE = *NO / *YES
,HOST = *STD / <c-string 1..8> / <filename 1..54 without-gen> / *ANY / *CATALOG(...)
  *CATALOG(...)
    |  IDENTIFICATION = <c-string 1..4> / <filename 1..54 without-gen>
,JOB-CLASS = *STD / <name 1..8> / *BY-USER-ID
,JOB-NAME = *NO / <name 1..8>
,MONJV = *NONE / <filename 1..54 without-gen-vers>
,JV-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / *SECRET / <integer -2147483648..2147483647>
,JOB-PRIORITY = *STD / <integer 1..9> / *BY-JOB-CLASS
,RERUN-AFTER-CRASH = *STD / *NO / *YES
,FLUSH-AFTER-SHUTDOWN = *STD / *NO / *YES
```

(Part 1 of 3)

```
,SCHEDULING-TIME = *STD / *PARAMETERS(...) / *BY-CALENDAR(...)

   *PARAMETERS(...)

      START = *STD / *BY-JOB-CLASS / *SOON / *IMMEDIATELY / *AT-STREAM-STARTUP /
                 *WITHIN(...) / *AT(...) / *EARLIEST(...) / *LATEST(...)

        *WITHIN(...)

              HOURS = 0 / <integer 0..23 hours>
             ,MINUTES = 0 / <integer 0..59 minutes>

        *AT(...)

              DATE = *TODAY / <date>
             ,TIME = <time>

        *EARLIEST(...)

              DATE = *TODAY / <date>
             ,TIME = <time>

        *LATEST(...)

              DATE = *TODAY / <date>
             ,TIME = <time>

     ,REPEAT-JOB = *STD / *BY-JOB-CLASS / *NO / *DAILY / *WEEKLY / *AT-STREAM-STARTUP /
                      *PERIOD(...)

        *PERIOD(...)

              HOURS = 0 / <integer 0..23 hours>
             ,MINUTES = 0 / <integer 0..59 minutes>

   *BY-CALENDAR(...)

      CALENDAR-NAME = <filename 1..54 without-gen-vers>

     ,SYMBOLIC-DATE = <filename 1..20 without-cat-user-vers> /
                      <partial-filename 2..20 without-cat-user>

,LIMIT = *STD / <integer 1..32767> / *BY-DATE(...)

   *BY-DATE(...)

      DATE = <date>
     ,TIME = <time>

,RESOURCES = *PARAMETERS(...)

   *PARAMETERS(...)

      RUN-PRIORITY = *STD / <integer 30..255> / *BY-JOB-CLASS

     ,CPU-LIMIT = *STD / *NO / <integer 1..32767 seconds> / *BY-JOB-CLASS

     ,SYSLST-LIMIT = *STD / *NO / <integer 0..999999> / *BY-JOB-CLASS
```

(Part 2 of 3)

```
,LOGGING = *PARAMETERS (...)
    *PARAMETERS(...)
      |  LISTING = *STD / *NO / *YES
,JOB-PARAMETER = *STD / *NO / <c-string 1..127>
,PROTECTION = *NONE / *CANCEL
```

(Part 3 of 3)

Operands

FROM-FILE = *LIBRARY-ELEMENT(...) / <filename 1..54 without-gen>
Name of the ENTER file.

i If the file/library name is specified without a catalog/user ID and if it is not cataloged in the user ID, the system tries to access a file or library of the same name in the system default ID. (For information on this “secondary read” function see the “Introductory Guide to DMS” [13].)

If the job submitter is not the file owner (differing user IDs), the file must be accessible (see the USER-ACCESS=*ALL-USERS operand in the CREATE-FILE and MODIFY-FILE-ATTRIBUTES commands).
The job submitter must in any case have at least execution privileges if the file is protected by a basic ACL or GUARDS.
If the file has an execute password, the password must be specified in the FILE-PASSWORD operand.

FROM-FILE = *LIBRARY-ELEMENT(...)
The ENTER file is stored in a PLAM library.

LIBRARY = <filename 1..51 without-gen>
Name of the library containing the ENTER file as an element. The specification of a library list (see the “LMS” manual [21]) instead of a library file is not supported.

ELEMENT = <composed-name 1..38>
Name of the element.
The following applies for the sum of the lengths of the library and element names:
– Without catalog and user IDs the sum may amount to at most 39 characters.
– In the case of a multi-character catalog ID, the sum with the full path name of the library including the catalog and user IDs may amount to at most 52 characters.

PROCESSING-ADMISSION =
Specifies the user ID under which the batch job is to run (target user ID).

PROCESSING-ADMISSION = *STD
The batch job is to run under the user's own user ID (i.e. the one under which ENTER-JOB was specified).

The following applies if the batch job is started on the console by an operator:
The batch job runs under the user ID which is contained in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file. If no user ID is specified there, a distinction must be made between the following two cases:

– If the operator LOGON function is not enabled (compatible mode), the console has no user ID of its own and the batch job is rejected.
– If the operator LOGON function is enabled (incompatible mode), the batch job runs under the operator's user ID.

PROCESSING-ADMISSION = *PARAMETERS(...)
Parameters defining the LOGON authorization of the target user ID.

USER-IDENTIFICATION = *NONE / <name 1..8>
User ID under which the batch job should run.

ACCOUNT = *NONE / <alphanum-name 1..8>
Account number of the user ID.

PASSWORD = *NONE / <c-string 1..8> / <c-string 9..32> / <x-string 1..16> / *SECRET
Password for the user ID.
The long password mechanism is supported (<c-string 9..32>). See the MODIFY-USER-PROTECTION command for details of the long password mechanism.
The PASSWORD operand has the following special characteristics:

– The password entered is not logged.
– The input field is automatically blanked out in the guided dialog.
– In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

PROCESSING-ADMISSION = *FROM-CALLER(...)
This specification is only permissible for tasks with the OPERATING privilege.
The batch job is to run under the user ID of the calling task. However, the option of specifying a different account number is available.

ACCOUNT = *STD / <alphanum-name 1..8>
Account number for job execution. Default value =*STD, i.e. job execution is billed under the account number of the calling task.

DEFAULT-FROM-FILE = *STD / *NO / *YES
This operand is only available for tasks with the OPERATING privilege.
Specifies whether the job attributes from the SET-LOGON-PARAMETERS command of the ENTER file are to be taken over when the default value is specified implicitly or explicitly in the ENTER-JOB command (see also the section "Privileged functions", page 3-237).

DEFAULT-FROM-FILE = *STD
If the ENTER-JOB command is called from the console, attributes with which the default value is specified are taken over from the ENTER file (corresponds to DEFAULT-FROM-FILE=*YES). With all other calls the specification corresponds to DEFAULT-FROM-FILE=*NO.

DEFAULT-FROM-FILE = *NO
When the default value is specified, the job attributes are not taken over from the ENTER file.

DEFAULT-FROM-FILE = *YES
Job attributes which are specified in the SET-LOGON-PARAMETERS command of the ENTER file are taken over when the default value is specified implicitly or explicitly for the various attributes in the ENTER-JOB command.

FILE-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / *SECRET
Password with which the ENTER file is protected against writing or execution
(see also the DELETE operand). The password specified is ignored for password-protected PLAM elements.
The FILE-PASSWORD operand has the following special characteristics:

– The password entered is not logged.
– The input field is automatically blanked out in the guided dialog.
– In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

CRYPTO-PASSWORD = *NONE / <c-string 1..8> / <x-string 1..16> / *SECRET
This operand is available only for tasks with the OPERATING privilege. For an encrypted ENTER file a temporary copy is created which is decrypted with the help of the crypto password.
The operand has the following special characteristics:

– The password entered is not logged.
– The input field is automatically blanked out in the guided dialog.
– In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

DELETE = *NO / *YES
Specifies whether the ENTER file is to be deleted after the batch job has been processed. Deletion is only possible if the job submitter also has write access. If the file is protected against write accesses by a password, this password must be contained in the password table for the job (see ADD-PASSWORD command) or in the FILE-PASSWORD operand.

HOST =
Host on which the batch job is to run.
The host can be identified by the host name or the catalog ID (*CATALOG). It must be in the "active" state, otherwise the command is rejected. Operand values other than *STD are available only to users who have the HIPLEX MSCF (multiprocessor systems) software product.

HOST = *STD

The ENTER job is running on the local host.

HOST = <c-string 1..8>

Host name of the host the ENTER job is to run on.

HOST = <filename 1..54 without-gen>

Name of a job variable which has the host's name as its content.

HOST = *ANY

Allowed only on an XCS network. For details see the “HIPLEX MSCF” manual [25].

HOST = *CATALOG(...)

IDENTIFICATION = <c-string 1..4>

Catalog ID of the host on which the batch job is to run.
The catalog must be accessible, otherwise the command is rejected.

IDENTIFICATION = <filename 1..54 without-gen>

Name of a job variable which has the host's catalog ID as its content.

JOB-CLASS = *STD / <name 1..8> / *BY-USER

Job class in which the batch job is to run. The job class must be permitted for batch jobs. The user can ascertain the job classes he/she is allowed to use from his/her user entry for the home pubset (which can be queried using the SHOW-USER-ATTRIBUTES command). This also displays the default job class that is preset with *STD. Users can obtain information about the characteristics of job classes (job class definition) by means of the SHOW-JOB-CLASS command.

JOB-CLASS = *STD

Default job class of the target user ID.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

JOB-CLASS = <name 1..8>

The ENTER job runs on the local computer.

JOB-CLASS = *BY-USER-ID

This specification is only possible with the OPERATING privilege..

The batch job is to run in the default job class of the target user ID.

JOB-NAME = *NO / <name 1..8>
Name for the ENTER job. The ENTER job can be addressed using this name (e.g. with SHOW-JOB-STATUS). The name is also printed on the header page of the printer listing. For the sake of compatibility a sequential search is performed to attempt to take over one of the following values as the name of the ENTER job:

1. *Not when started from the console:* A job name specified here which is not equal to *NO
2. *Not when started from the console:* A marker located before the ENTER-JOB command (/.<name> ENTER-JOB ...)
3. The job name from the SET-LOGON-PARAMETERS command in the ENTER file
4. A marker located before the SET-LOGON-PARAMETERS command in the ENTER file
5. *Not when started from the console:* The name of the command-submitting job

If none of these values is available, the ENTER job is assigned no name, which is shown in the outputs with *NONE, (NONE) or blanks.

MONJV = *NONE / <filename 1..54 without-gen-vers>
Specifies whether the batch job is to be monitored by a JV.

MONJV = *NONE
The batch job is not monitored.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
Tasks with the OPERATING privilege can use the DEFAULT-FROM-FILE operand to set this default mechanism.

MONJV = <filename 1..54 without-gen-vers>
Name of the JV which is to monitor the batch job. A JV can only be defined if the chargeable subsystem JV is available (see also the "Job Variables" manual [20]).
Job monitoring is only started if the batch job is accepted by the system’s job management (`JOB ACCEPTED`).
The job originator must have write authorization because he instructs the system to write to the JV. If the JV is not accessible at the time of command processing, an error message is output to SYSOUT and the command is rejected. If the specified JV does not yet exist, it is - if the requisite authorization exists - created and made available for all users (ACCESS=*WRITE and USER-ACCESS=*ALL-USERS). The user can address his/her batch job using the specified JV.

Catalog ID of the computer on which the batch job is to run.

$S	Job on queue
$R	Job running
$T	Job terminated
$A	Job aborted
$M	Job exported with MOVE-JOBS

JV-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647> / *SECRET

Password for the JV.
The operand is only evaluated when job monitoring has been defined (see the MONJV operand).
The operand JV-PASSWORD is defined as “secret”:

– The password entered is not logged.
– The input field is automatically blanked out in the guided dialog.
– In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

JOB-PRIORITY = *STD / <integer 1..9> / *BY-JOB-CLASS

Job priority to be given to the batch job.
The lower the value, the higher the priority. The values can be queried with the SHOW-USER-ATTRIBUTES and SHOW-JOB-CLASS commands.

JOB-PRIORITY = *STD

The standard priority specified for the job class applies.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

JOB-PRIORITY = *BY-JOB-CLASS

This specification is only possible with the OPERATING privilege..
The batch job is to be assigned the default priority specified for the job class.

RERUN-AFTER-CRASH = *STD / *NO / *YES

Specifies whether the batch job is to be restarted during the next system session if processing has been aborted as the result of a system error or termination of the system session.
The operand is not evaluated if job repetition is enabled in the REPEAT-JOB operand.

RERUN-AFTER-CRASH = *STD
RERUN-AFTER-CRASH=*NO applies.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

FLUSH-AFTER-SHUTDOWN = *STD / *NO / *YES
Specifies whether the batch job is to be removed from the job queue if it has not been processed by the end of the session.
The operand is not evaluated if job repetition is enabled in the REPEAT-JOB operand.
The specification FLUSH-AFTER-SHUTDOWN=*YES with the warning JMS0056 is ignored for calendar jobs.

FLUSH-AFTER-SHUTDOWN = *STD
FLUSH-AFTER-SHUTDOWN =*NO applies.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

SCHEDULING-TIME = *STD / *PARAMETERS(...) / *BY-CALENDAR(...)
Defines how scheduling times are specified for the batch job.

SCHEDULING-TIME = *STD
The default settings for START and REPEAT-JOB scheduling time specifications for the selected job class apply (see the operands of the SCHEDULING-TIME= *PARAMETERS(...) structure).
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

SCHEDULING-TIME = *PARAMETERS(...)
Defines a scheduling time (start time) for the batch job. It is also possible to define job repeats (repeat job).

START =
Starting time for the batch job. Values other than *STD are appropriate only if permitted in accordance with the job class definition (see the SHOW-JOB-CLASS command). Only the value *IMMEDIATELY can also be permitted as an alternative by means of the user properties (see the SHOW-USER-ATTRIBUTES command).

START = <u>*STD</u>
The default value for the chosen job class applies.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

START = *BY-JOB-CLASS
This specification is only possible with the OPERATING privilege.
The batch job is to be assigned the start time defined for the job class.

START = *SOON
The job is to be started as soon as possible, in accordance with its priority.

START = *IMMEDIATELY
The job is to be started immediately.

START = *AT-STREAM-STARTUP
The job is to be started after the next startup of the job scheduler.

START = *WITHIN(...)
The job is to be started within the specified time period.

HOURS = <u>0</u> / <integer 0..23 *hours*>
Number of hours.

MINUTES = <u>0</u> / <integer 0..59 *minutes*>
Number of minutes.

START = *AT(...)
The job is to be started exactly at the time specified in the following.

DATE = *TODAY / <date>
Date. This can be specified in the form [yy]yy-mm-dd. Only the last two digits of the year are evaluated, which means that the century is ignored in four-digit year specifications. 20 is automatically prefixed to two-digit year specifications < 80, 19 to two-digit year specifications ≥ 80.

TIME = <time>
Time of day in the format hh:mm, where hh = hours and mm = minutes. Seconds are not interpreted.

START = *EARLIEST(...)
The job is to be started no earlier than the time specified.

DATE = *TODAY / <date>
Date. This can be specified in the form [yy]yy-mm-dd. Only the last two digits of the year are evaluated, which means that the century is ignored in four-digit year specifications.
20 is automatically prefixed to two-digit year specifications < 80, 19 to two-digit year specifications ≥ 80.

TIME = <time>
Time of day in the format hh:mm, where hh = hours and mm = minutes. Seconds are not interpreted.

START = *LATEST(...)
The job is to be started no later than the time specified.

DATE = *TODAY / <date>
Date. This can be specified in the form [yy]yy-mm-dd. Only the last two digits of the year are evaluated, which means that the century is ignored in four-digit year specifications.
20 is automatically prefixed to two-digit year specifications < 80, 19 to two-digit year specifications ≥ 80.

TIME = <time>
Time of day in the format hh:mm, where hh = hours and mm = minutes. Seconds are not interpreted.

REPEAT-JOB =
Time interval at which the batch job is to be repeated. Values other than *STD are appropriate only if permitted in accordance with the job class definition (see the SHOW-JOB-CLASS command). The time interval for the repetitions depends on the specification in the START operand; see the note in this regard, “Combinations of the START and REPEAT-JOB operands”. For the repetitions, the following applies:

- The i-th repetition (i ≥ 1) of a job is not started until the (i1)th repetition has ended.
- Cancellation of the currently executing job (i) has no effect on the start of (i+1); (i ≥ 0).
- Cancellation of the entire job: Both the currently executing job (i) and the subsequent job (i+1) must be canceled, (i ≥ 0); (CANCEL-JOB command, or make job (i) the last job in the series using the command MODIFY-JOB ...,REPEAT-JOB=*NO).

REPEAT-JOB = *STD
The default value for the chosen job class applies.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

REPEAT-JOB = *BY-JOB-CLASS
This specification is only possible with the OPERATING privilege.
The batch job is to be assigned the default value defined for the job class.

REPEAT-JOB = *NO
The batch job is not repeated.

REPEAT-JOB = *DAILY
Daily repetition at the time specified with START.

REPEAT-JOB = *WEEKLY
Weekly repetition at the time specified with START.

REPEAT-JOB = *AT-STREAM-STARTUP
Repetition following each startup of the job scheduler.

REPEAT-JOB = *PERIOD(...)
Repetition after the specified time interval.

HOURS = 0 / <integer 0..23 *hours*>
Number of hours.

MINUTES = 0 / <integer 0..59 *minutes*>
Number of minutes.

SCHEDULING-TIME = *BY-CALENDAR(...)
The batch job scheduling time and any repeat jobs are specified in the form of a symbolic date defined in a calendar file (calendar job). The entries in a calendar file can be listed with the SHOW-CALENDAR command. Creation of calendar files with the CALENDAR utility is described in the “Calendar” manual [4].

CALENDAR-NAME = <filename 1..20 without-cat-user-gen-vers>
Name of the calendar file.

SYMBOLIC-DATE = <filename 1..20 without-cat-user-vers> / <partial-filename 2..20 without-cat-user>
Symbolic date which defines the scheduling time and any repetition cycles within the calendar file. The symbolic date may also be given in partially qualified mode. In this way, several scheduling times can be defined for one calendar day with the appropriate definition of SYSDATs.

Example: Definition of SYMDATs in the calendar file:

- WORK.DAY.1 (every other day at 06:00 hrs)
- WORK.DAY.2 (every other day at 6:00 PM hrs)
- WORK.WEEK.1 (every Friday at 21:00 hrs)

A calendar job considering all three scheduling points is started with `SYMBOLIC-DATE=WORK.`.

LIMIT = *STD / <integer 1..32767> / *BY-DATE(...)
Governs how long a calendar job remains in existence. This limit applies in addition to the limits set by the calendar.

LIMIT = *STD
The duration of the calendar job depends entirely on the symbolic date entry in the calendar.

LIMIT = <integer 1..32767>
This specification is only permitted for calendar jobs.
Maximum number of repetitions of the calendar job
On termination, a check is performed to determine whether the run counter has reached or exceeded the maximum value. If this is the case, the entire calendar job is terminated. Otherwise the run counter is incremented by 1.

LIMIT = *BY-DATE(...)
This specification is only permitted for calendar jobs.
Entries in the calendar file are only taken into account up to the specified limit. No further repeat job is generated for calendar entries after the limit; the calendar job terminates. The limit refers solely to the schedule entries in the file, not to the real runtime of the jobs. Repeat jobs with a "permissible" start date are not subject to any further restrictions and are, for example, also started after the specified date if this was not possible earlier because of delays in the job scheduler.
The date specification consists of the day and the time:

DATE = <date>
Date. This can be specified in the form [yy]yy-mm-dd. Only the last two digits of the year are evaluated, which means that the century is ignored in four-digit year specifications. 20 is automatically prefixed to two-digit year specifications < 80, 19 to two-digit year specifications ≥ 80.

TIME = <time>
Time of day.

RESOURCES = PARAMETERS(...)
Values for run priority, CPU time and maximum number of SYSLST records.

RUN-PRIORITY = *STD / <integer 30..255> / *BY-JOB-CLASS
Run priority the batch job is to be given. The lower the value, the higher the priority. The maximum permissible priority value is the lesser of the two values (i.e. the more favorable of the values) from the user catalog and the job class definition.
If no maximum value is defined for the job class, the following rules apply:

– If the value specified explicitly is numerically lower than the value in the user entry, the message JMS0045 is issued. The batch job is assigned the higher of the two values (i.e. the less favorable value) for the run priority from the user entry and the default run priority for the job class.
– If no value is specified or if *STD or *BY-JOB-CLASS is explicitly specified, the batch job is assigned the standard run priority of the job class.

The values can be queried with the SHOW-USER-ATTRIBUTES and SHOW-JOB-CLASS commands.

RUN-PRIORITY = *STD
The standard run priority specified for the job class applies.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

RUN-PRIORITY = *BY-JOB-CLASS
This specification is only possible with the OPERATING privilege. The batch job is to be assigned the default value defined for the job class.

CPU-LIMIT = *STD / *NO / <integer 1..32767 *seconds*> / *BY-JOB-CLASS
Maximum CPU time, in seconds, that the batch job may consume. The maximum time permitted depends on the job class specified. See also section “Time limits in BS2000” on page 1-103.

CPU-LIMIT = *STD
The default value for the chosen job class applies.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
Tasks with the OPERATING privilege can set this default mechanism in the DEFAULT-FROM-FILE operand.

CPU-LIMIT = *NO
The ENTER job is to run with no time limit (NTL).
This operand value is permitted only if the requisite authorization exists in the user entry or the job class definition.

CPU-LIMIT = *BY-JOB-CLASS
This specification is only possible with the OPERATING privilege.
The batch job is to be assigned the default value defined for the job class.

SYSLST-LIMIT = *STD / *NO / <integer 0..999999> / *BY-JOB-CLASS
Designates the maximum number of records output by the job to the system files SYSLST, SYSLST01, SYSLST02, ..., SYSLST99. Data records in the system file SYSOUT that are simultaneously written to SYSLST are not counted.
This value must not be above the limit set in the job class definition. This limit may be queried using the SHOW-JOB-CLASS command.
If the specified number is exceeded:
- in batch mode, the job is terminated abnormally
- in interactive mode, the user may specify whether the job is to be continued or terminated. If continued, output is limited by “number” again.

SYSLST-LIMIT = *STD
The default value for the chosen job class applies.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

SYSLST-LIMIT = *NO
Sets no limit on the number of records output.

SYSLST-LIMIT = *BY-JOB-CLASS
This specification is only possible with the OPERATING privilege.
The batch job is to be assigned the default value defined for the job class.

LOGGING = *PARAMETERS(...)
Controls the logging of the job progress.

LISTING = *STD / *NO / *YES
Specifies whether the job run is also to be logged on SYSLST.

LISTING = *STD
LISTING=*NO applies.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

JOB-PARAMETER =
Specifies additional attributes for the selected job class - assuming that systems support has defined some and made them known.

JOB-PARAMETER = *STD
JOB-PARAMETER=*NO applies.
When the batch job is started by the operator on the console, the specification in the operand of the same name in the SET-LOGON-PARAMETERS command of the ENTER file is evaluated.
If there is no value specified there, the standard value specified for the job class applies.

JOB-PARAMETER = *NO
No additional attributes.

JOB-PARAMETER = <c-string 1..127>
Arbitrary sequence of characters defined by the system administrator to identify additional job class attributes.

PROTECTION = *NONE / *CANCEL
Specifies whether the batch job is to be protected against accidental termination with the CANCEL-JOB command.

PROTECTION = *NONE
The batch job is not protected against accidental termination.

PROTECTION = *CANCEL
The batch job is protected against unintentional cancelation. In interactive jobs and on consoles which wish to terminate this batch job with the CANCEL-JOB command, the system additionally requests confirmation. Accidental termination of the batch job due to incorrect specification of the job number should thus be prevented.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
2	0	CMD0002	Command executed with a warning e.g. DELETE=*YES ignored for repeat job or when a library member is specified
	1	CMD0202	Syntax error in command
	32	CMD0221	system error
	64	JMS0630	Semantic or privilege error, e.g. host, catalog ID, job class unknown; MONJV not accessible, maximum number of jobs reached
	64	JMS0640	ENTER file faulty or not accessible, e.g. missing SET-LOGON-PARAMETERS, not a SAM or ISAM file, file empty, no access right
	64	JMS0670	Error in a remote job
	130	JMS0620	No further storage space or TSN available, or specified MONJV already monitoring a job
	130	JMS0650	MSCF not available or no connection to the specified host or the specified catalog is currently not available

Notes

– Combinations of the START and REPEAT-JOB operands:

	REPEAT-JOB		
START	**AT-STREAM-STARTUP**	**DAILY or WEEKLY**	**PERIOD**
IMMEDIATELY or SOON	a)	c)	c)
AT or EARLIEST	a)	d)	f)
LATEST or WITHIN	a)	c)	g)
AT-STREAM-STARTUP	b)	e)	e)

Table 49: Combination of the START and REPEAT-JOB operands in the ENTER-JOB command

a) The first and all subsequent starts of the job take place as specified.

b) The first start of the job is made with START=*AT-STREAM-STARTUP. All further starts take place after the startup of the job scheduler with START=*SOON.

c) The base time for the repetition cycle is the time the job is accepted.

d) The specified point in time (START=...., TIME=....) is the base time for the repetition cycle.

e) The first start of the job follows startup of the job scheduler. This point in time is the base time for the repetition cycle. Further starts take place with START=*SOON.

f) The specified point in time (START=...., TIME=....) is the base time for the repetition cycle. The second and all further starts take place with START=*SOON.

g) The base time for the repetition cycle is the time the job is accepted. All further starts take place with START=*SOON.

– The following applies to the *WITHIN, *AT and *LATEST entries in the START operand: After the specified point in time or period of time, jobs that have not been started are treated in the same way as jobs started with START=*SOON and highest job priority.

Example

A job with START=*LATEST(...) could not be started by the desired time because the job scheduler was not active. It will then be started (within the same session) as soon as possible after the next startup of the job scheduler.

- Determining the scheduling time of a calendar job:
  - In the first version of the calendar job, the symbolic date (SYMDAT) specified in the SYMBOLIC-DATE operand is passed on to the CALENDAR component in the evaluation of the job attributes. The CALENDAR component returns the next point in time (date and time), with regard to the current point in time, specified by the SYMDATs defined in the calendar file.
    In the case of partially qualified SYMDATs, a scheduling time is returned for each SYMDAT beginning with the character string, and the calendar job is started at each of these scheduling times.
  - The scheduling times of the following versions are determined according to the same procedure while the previous jobs are processed.

  As a consequence, any modifications made to the calendar file only take effect on calendar job versions whose scheduling time is determined after the update of the calendar file.
  In particular, the number of calendar job versions started by means of a partially qualified SYMDAT can be extended (by defining new SYMDATs) or reduced (by deleting SYMDATs).

- Creation of a temporary copy of the ENTER file (S.IN file):
  A temporary copy of the ENTER file is created in the following cases under a file name with the prefix S.IN (supplemented by partial name for TSN, date and time or library, member and time):
  - if the file is located on private disk.
  - if the file is cataloged under a different user ID.
  - if the file is a temporary file.
  - if the file is a library member.
  - if the file is encrypted.

  Special characteristics of the S.IN file:

  - The S.IN file is protected by a file lock for the duration of the batch job.
    Note the following: the file lock is set when the pubset on which the files are located is imported. The file locks only consider files to which batch jobs from the current job pool (on the home pubset) refer. If the files are located on a shared pubset, the file locks are coordinated from the master computer.

  - An S.IN file is always unencrypted. It must therefore be possible to decrypt the ENTER file when the command is processed, i.e. the crypto password must be specified in the CRYPTO-PASSWORD operand.

- ENTER files can be protected by means of read, write and execute passwords. The execute password or a higher-ranking password must be specified in the FILE-PASSWORD operand before an ENTER-JOB command is issued. The write password must be specified when the file is to be deleted after it has been executed

(DELETE=*YES). The passwords are checked for correctness as soon as the ENTER-JOB command is processed. If a user subsequently changes the passwords, the successful check still applies and the file is executed.

- When the ENTER job is started on a remote system (HOST not equal to *STD), all the passwords used (CRYPTO-PASSWORD, FILE-PASSWORD and JV-PASSWORD) must be specified directly in the ENTER-JOB command. Passwords which were entered in the job submitting task's password list with the ADD-PASSWORD and ADD-CRYPTO-PASSWORD commands apply only when the job is started locally (HOST = *STD).

- ENTER files can be SAM or ISAM files and have a variable record length. 72 characters are interpreted per record. In the case of ISAM files the key field can occur in any position in the record as it will be hidden.

- Any LOGON operands specified in the ENTER file are ignored if the command is not entered on the console or with DEFAULT-FROM-FILE=*YES specified (see "Privileged functions" page 3-237).

- DELETE operand: Despite DELETE=*YES being entered, the file is not deleted when
  - the job submitter is not the (co-)owner of the file,
  - the file is a library member,
  - the job terminates abnormally,
  - the job is aborted.

  The last two cases do not apply for the following files:
  - files on private disk
  - temporary files
  - files which are cataloged under a different user ID from that under which the ENTER job is to execute

  In these cases the ENTER file is deleted immediately after the S.IN file has been created. This enables, for instance, the private disk to be removed before the ENTER job has executed.

- A batch job which is to run on a remote system can be accessed via a MONJV only when the catalog ID of the partner system’s home pubset is entered in the MRSCAT of the systems concerned.

- In the case of repeat jobs which were started with REPEAT-JOB=*PERIOD(...), compliance with the time interval has priority over compliance with the start time. Consequently the start time shifts in the event of time changes.

- In nonprivileged job classes, up to 32767 waiting jobs are permissible. Any ENTER jobs in excess of this number are rejected.

ENTER-PROCEDURE

Start command sequence (procedure file) as batch job

Description status: BS2000 OSD/BC V10.0A

Functional area: Job processing Procedures

Domain: PROCEDURE

Privileges: STD-PROCESSING
HARDWARE-MAINTENANCE

Function

Using the ENTER-PROCEDURE command, the user can start a procedure as a batch job. In contrast to the ENTER-JOB command, the user does not have to create a separate ENTER file. The procedure parameters are consequently variable at every asynchronous procedure execution (background procedure). ENTER files can only be started with the ENTER-JOB command.

Method

1. The procedure file is created as a copy under the name `S.PROC.tsn.date.time`, where *date* has the format yyyy-mm-dd and *time* has the format hh.mm.ss.
2. An ENTER file with the name `S.E.tsn.date.time` and the following contents is created:

   ```
   /SET-LOGON-PARAMETERS
     .
     .
     .
   /CALL-PROCEDURE FROM-FILE=S.PROC.tsn.date.time, -
   /                PROCEDURE-PARAMETERS=(parameter)
     .
     .
     .
   /EXIT-JOB SYSTEM-OUTPUT=option
   ```

   The value of *parameter* corresponds to the entry in the PROCEDURE-PARAMETERS operand. The value of *option* is set in accordance with the entry in the SYSTEM-OUTPUT operand. After procedure execution, the copy of the procedure file is deleted.
3. The ENTER file is started with ENTER-JOB. Entries for the operands PROCESSING-ADMISSION, JOB-CLASS, JOB-NAME, MONJV, JV-PASSWORD, JOB-PRIORITY, RERUN-AFTER-CRASH, FLUSH-AFTER-SHUTDOWN, SCHEDULING-TIME, START, REPEAT-JOB, LIMIT, RESOURCES, LISTING and JOB-PARAMETER are transferred to the ENTER-JOB command.

Format

```
ENTER-PROCEDURE                                                                        Alias: ENP

FROM-FILE = <filename 1..54 without-gen> / *LIBRARY-ELEMENT(...)

   *LIBRARY-ELEMENT(...)

        LIBRARY = <filename 1..51 without-gen>
       ,ELEMENT = <composed-name 1..38>

,PROCEDURE-PARAMETERS = *NO / <text 0..1800 with-low>

,PROCESSING-ADMISSION = *SAME / *PARAMETERS(...)

   *PARAMETERS(...)

        USER-IDENTIFICATION = *NONE / <name 1..8>
       ,ACCOUNT = *NONE / <alphanum-name 1..8>
       ,PASSWORD = *NONE / <c-string 1..8> / <c-string 9..32> / <x-string 1..16> / *SECRET

,PROCEDURE-PASSWORD = *NONE / <x-string 1..8> / <c-string 1..4> /
                                   <integer -2147483648..2147483647> / *SECRET

,CRYPTO-PASSWORD = *NONE / <c-string 1..8> / <x-string 1..16> / *SECRET

,HOST = *STD / <c-string 1..8> / *ANY

,JOB-CLASS = *STD / <name 1..8>

,JOB-NAME = *NO / <name 1..8>

,MONJV = *NONE / <filename 1..54 without-gen-vers>

,JV-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / *SECRET /
                           <integer -2147483648..2147483647>

,JOB-PRIORITY = *STD / <integer 1..9>

,RERUN-AFTER-CRASH = *NO / *YES

,FLUSH-AFTER-SHUTDOWN = *NO / *YES

,SCHEDULING-TIME = *STD / *PARAMETERS(...) / *BY-CALENDAR(...)

   *PARAMETERS(...)

        START = *STD / *SOON / *IMMEDIATELY / *AT-STREAM-STARTUP / *WITHIN(...) / *AT(...) /
                      *EARLIEST(...) / *LATEST(...)

          *WITHIN(...)
                HOURS = 0 / <integer 0..23 hours>
               ,MINUTES = 0 / <integer 0..59 minutes>

          *AT(...)
                DATE = *TODAY / <date>
               ,TIME = <time>
```

(Part 1 of 2)

```
          *EARLIEST(...)
               DATE = *TODAY / <date>
              ,TIME = <time>

          *LATEST(...)
               DATE = *TODAY / <date>
              ,TIME = <time>

         ,REPEAT-JOB = *STD / *NO / *DAILY / *WEEKLY / *AT-STREAM-STARTUP / *PERIOD(...)

          *PERIOD(...)
               HOURS = 0 / <integer 0..23 hours>
              ,MINUTES = 0 / <integer 0..59 minutes>

     *BY-CALENDAR(...)
          CALENDAR-NAME = <filename 1..54 without-gen-vers>
         ,SYMBOLIC-DATE = <filename 1..20 without-cat-user-vers> /
                          <partial-filename 2..20 without-cat-user>

,LIMIT = *STD / <integer 1..32767> / *BY-DATE(...)
     *BY-DATE(...)
          DATE = <date>
         ,TIME = <time>

,RESOURCES = *PARAMETERS (...)
     *PARAMETERS(...)
          RUN-PRIORITY = *STD / <integer 30..255>
         ,CPU-LIMIT = *STD / *NO / <integer 1..32767 seconds>
         ,SYSLST-LIMIT = *STD / *NO / <integer 0..999999>

,LOGGING = *STD / *YES / *NO
,LISTING = *NO / *YES
,JOB-PARAMETER = *NO / <c-string 1..127>
,SYSTEM-OUTPUT = *STD / *STDOUT / *PRINT / *MAIL / *DELETE
,ASSIGN-SYSTEM-FILES = *STD / *PARAMETERS(...)
     *PARAMETERS(...)
          SYSLST = *STD / *PRIMARY / *DUMMY / <filename 1..54>
         ,SYSOUT = *STD / *PRIMARY / *DUMMY / <filename 1..54>

,PROTECTION = *NONE / *CANCEL
```

(Part 2 of 2)

Operands

FROM-FILE = *LIBRARY-ELEMENT(...) / <filename 1..54 without-gen>
Name of the file or PLAM library element which contains the procedure.

> i If the file/library name is specified without a catalog/user ID and if it is not cataloged in the user ID, the system tries to access a file or library of the same name in the system default ID. (For information on this “secondary read” function see the “Introductory Guide to DMS” [13].)

The procedure must not begin with the SET-LOGON-PARAMETERS or LOGON command, i.e. it must not be an ENTER file.
If the job submitter is not the file owner (differing user IDs), the file must be accessible (see the operand PROTECTION=PARAMETERS in the CREATE-FILE and MODIFY-FILE-ATTRIBUTES commands).
The job submitter must in any case have at least execution privileges if the file is protected by a basic ACL or GUARDS.
If the file has an execute password, the password must be specified in the PROCEDURE-PASSWORD operand.

FROM-FILE = *LIBRARY-ELEMENT(...)
The procedure is stored in a PLAM library.

LIBRARY = <filename 1..51 without-gen>
Name of the library containing the procedure as an element. The specification of a library list (see the “LMS” manual [21]) instead of a library file is not supported.

ELEMENT = <composed-name 1..38>
Name of the element.
The following applies for the sum of the lengths of the library and member names:

– Without catalog and user IDs the sum may amount to at most 39 characters.
– In the case of a multi-character catalog ID, the sum with the full path name of the library including the catalog and user IDs may amount to at most 52 characters.

PROCEDURE-PARAMETERS = *NO / <text 0..1800 with-low>
Parameter values which are to be set instead of the appropriate symbolic parameters. Parameter values must be enclosed in parentheses. Input is carried out as described in the CALL-PROCEDURE command.

PROCESSING-ADMISSION =
Specifies the user ID under which the batch job is to run.

PROCESSING-ADMISSION = *SAME
The batch job should run under the current user ID (i.e. the one under which ENTER-PROCEDURE was specified).

PROCESSING-ADMISSION = *PARAMETERS(...)
Parameters defining the LOGON authorization of the destination user ID.

USER-IDENTIFICATION = *NONE / <name 1..8>
User ID under which the batch job should run.

ACCOUNT = *NONE / <alphanum-name 1..8>
Account number of the user ID.

PASSWORD = *NONE / <c-string 1..8> / <c-string 9..32> / <x-string 1..16> / *SECRET
Password for the user ID.
The long password mechanism is supported (<c-string 9..32>). A hash algorithm converts the long password to the internal 8-byte representation. See the MODIFY-USER-PROTECTION command for details of the long password mechanism.
The PASSWORD operand has the following special characteristics:

- The password entered is not logged.
- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

PROCEDURE-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647> / *SECRET
Password protecting the procedure file from being executed. The operand has the following special characteristics:

- The password entered is not logged.
- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

CRYPTO-PASSWORD = *NONE / <c-string 1..8> / <x-string 1..16> / *SECRET
Password used when encrypting the procedure file. The copy of the procedure file (S.PROC file) is decrypted with the aid of the crypto password.
The operand has the following special characteristics:

- The password entered is not logged.
- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

HOST =
Specifies the host the job is to run on. Operand values other than *STD are available only to users who have the HIPLEX MSCF (multiprocessor systems) software product.

HOST = *STD
The job is to run on the local host.

HOST = <c-string 1..8>
Host name of the host the ENTER job is to run on.

HOST = *ANY
Allowed only on an XCS network. For details see the “HIPLEX MSCF” manual [25].

JOB-CLASS = *STD / <name 1..8>
Job class into which the job is to be placed. The SHOW-USER-ATTRIBUTES or SHOW-JOB-CLASS command can be used to query authorization for the various job classes.

JOB-NAME = *NO / <name 1..8>
Name for the ENTER job. The ENTER job can be addressed using this name (e.g. with SHOW-JOB-STATUS). The name is also printed on the header page of the printer listing.

If *NO is specified, the ENTER job is assigned the name of the job issuing the command, for reasons of compatibility. Only if this job has no name as well, the ENTER job is assigned no name, which is shown in the outputs with *NONE, (NONE) or blanks.

MONJV = *NONE / <filename 1..54 without-gen-vers>
Specifies whether the batch job is to be monitored by a JV.

MONJV = *NONE
The batch job is not monitored.

MONJV = <filename 1..54 without-gen-vers>
Only possible if the chargeable JV subsystem is loaded (see also the "Job Variables" manual [20]).
Name of the job variable (JV) that is to monitor the batch job. The user can address the batch job via this JV. If the specified JV does not yet exist, it is - if the requisite authorization exists - created and made available for all users (ACCESS=*WRITE and USER-ACCESS=*ALL-USERS).
The system sets the JV to appropriate values while the batch job is being processed:

$S	Job on queue
$R	Job running
$T	Job terminated
$A	Job aborted
$M	Job exported with MOVE-JOBS

JV-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647> / *SECRET

Password for the JV.

The operand has the following special characteristics:

- The password entered is not logged.
- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

JOB-PRIORITY = *STD / <integer 1..9>

Job priority to be given to the batch job.

The lower the value, the higher the priority. The values can be queried with the SHOW-USER-ATTRIBUTES and SHOW-JOB-CLASS commands.

JOB-PRIORITY = *STD

The standard priority specified for the job class applies.

RERUN-AFTER-CRASH = *NO / *YES

Specifies whether the batch job is to be restarted during the next system session if processing has been aborted as the result of a system error or termination of the system session.

Note

The operand is not evaluated if job repetition is enabled in the REPEAT operand.

FLUSH-AFTER-SHUTDOWN = *NO / *YES

Specifies whether the batch job is to be removed from the job queue if it has not been processed by the end of the session.

Note

The operand is not evaluated if job repetition is enabled in the REPEAT operand. The specification FLUSH-AFTER-SHUTDOWN=*YES with the warning JMS0056 is ignored for calendar jobs.

SCHEDULING-TIME = *STD / *PARAMETERS(...) / *BY-CALENDAR(...)

Defines how scheduling times are specified for the batch job.

SCHEDULING-TIME = *STD

The default settings for START and REPEAT-JOB scheduling time specifications for the selected job class apply (see the operands of SCHEDULING-TIME=*PARAMETERS(...)).

SCHEDULING-TIME = *PARAMETERS(...)
Defines a scheduling time (start time) for the batch job. It is also possible to define job repeats (repeat job).

START =
Starting time for the batch job. Values other than *STD are ignored unless they are permitted by virtue of the job class definition (see the SHOW-JOB-CLASS command).

START = <u>*STD</u>
The default value for the chosen job class applies.

START = *SOON
The job is to be started as soon as possible, in accordance with its priority.

START = *IMMEDIATELY
The job is to be started immediately.

START = *AT-STREAM-STARTUP
The job is to be started after the next startup of the job scheduler.

START = *WITHIN(...)
The job is to be started within the specified time period.

HOURS = <u>0</u> / <integer 0..23 *hours*>
Number of hours.

MINUTES = <u>0</u> / <integer 0..59 *minutes*>
Number of minutes.

START = *AT(...)
The job is to be started exactly at the time specified in the following.

DATE = *TODAY / <date>
Date. This can be specified in the form [yy]yy-mm-dd. Only the last two digits of the year are evaluated, which means that the century is ignored in four-digit year specifications. 20 is automatically prefixed to two-digit year specifications < 80, 19 to two-digit year specifications ≥ 80.

TIME = <time>
Time of day in the format hh:mm, where hh = hours and mm = minutes. Seconds are not interpreted.

START = *EARLIEST(...)
The job is to be started no earlier than the time specified.

DATE = *TODAY / <date>
Date. This can be specified in the form [yy]yy-mm-dd. Only the last two digits of the year are evaluated, which means that the century is ignored in four-digit year specifications. 20 is automatically prefixed to two-digit year specifications < 80, 19 to two-digit year specifications ≥ 80.

TIME = <time>
Time of day in the format hh:mm, where hh = hours and mm = minutes. Seconds are not interpreted.

START = *LATEST(...)
The job is to be started no later than the time specified.

DATE = *TODAY / <date>
Date. This can be specified in the form [yy]yy-mm-dd. Only the last two digits of the year are evaluated, which means that the century is ignored in four-digit year specifications. 20 is automatically prefixed to two-digit year specifications < 80, 19 to two-digit year specifications ≥ 80.

TIME = <time>
Time of day in the format hh:mm, where hh = hours and mm = minutes. Seconds are not interpreted.

REPEAT-JOB =
Time interval at which the batch job is to be repeated. Values other than *STD are ignored unless they are permitted by virtue of the job class definition (see the SHOW-JOB-CLASS command). The time interval for the repetitions depends on the specification in the START operand; see the note in this regard, “Combinations of the START and REPEAT-JOB operands”. For the repetitions, the following applies:

- The i-th repetition (i ≥ 1) of a job is not started until the (i-1)th repetition has ended.
- Cancellation of the currently executing job (i) has no effect on the start of (i+1); (i ≥ 0).
- Cancellation of the entire job: Both the currently executing job (i) and the subsequent job (i+1) must be canceled, (i ≥ 0); (CANCEL-JOB command, or make job (i) the last job in the series using the command MODIFY-JOB ...,REPEAT-JOB=*NO).

REPEAT-JOB = <u>*STD</u>
The default value for the chosen job class applies.

REPEAT-JOB = *NO
The batch job is not repeated.

REPEAT-JOB = *DAILY
Daily repetition at the time specified with START.

REPEAT-JOB = *WEEKLY
Weekly repetition at the time specified with START.

REPEAT-JOB = *AT-STREAM-STARTUP
Repetition following each startup of the job scheduler.

REPEAT-JOB = *PERIOD(...)
Repetition after the specified time interval.

HOURS = <u>0</u> / <integer 0..23 *hours*>
Number of hours.

MINUTES = <u>0</u> / <integer 0..59 *minutes*>
Number of minutes.

SCHEDULING-TIME = *BY-CALENDAR(...)
The batch job scheduling time and any repeat jobs are specified in the form of a symbolic date defined in a calendar file (calendar job). The entries in a calendar file can be listed with the SHOW-CALENDAR command. Creation of calendar files with the CALENDAR utility is described in the “Calendar” manual [4].

CALENDAR-NAME = <filename 1..54 without-cat-user-gen-vers>
Name of the calendar file.

SYMBOLIC-DATE = <filename 1..20 without-cat-user-vers> /
<partial-filename 2..20 without-cat-user>
Symbolic date which defines the scheduling time and any repetition cycles within the calendar file. The symbolic date may also be given in partially qualified mode. In this way, several scheduling times can be defined for one calendar day with the appropriate definition of SYSDATs.

Example: Definition of SYMDATs in the calendar file:

- WORK.DAY.1 (every other day at 06:00 hrs)
- WORK.DAY.2 (every other day at 6:00 PM hrs)
- WORK.WEEK.1 (every Friday at 21:00 hrs)

A calendar job considering all three scheduling points is started with `SYMBOLIC-DATE=WORK.`.

LIMIT = *STD / <integer 1..32767> / *BY-DATE(...)
Governs how long a calendar job remains in existence. This limit applies in addition to the limits set by the calendar.

LIMIT = *STD
The duration of the calendar job depends entirely on the symbolic date entry in the calendar.

LIMIT = <integer 1..32767>
This specification is only permitted for calendar jobs.
Maximum number of repetitions of the calendar job
On termination, a check is performed to determine whether the run counter has reached or exceeded the maximum value. If this is the case, the entire calendar job is terminated. Otherwise the run counter is incremented by 1.

LIMIT = *BY-DATE(...)

This specification is only permitted for calendar jobs. Entries in the calendar file are only taken into account up to the specified limit. No further repeat job is generated for calendar entries after the limit; the calendar job terminates.
The limit refers solely to the schedule entries in the file, not to the real runtime of the jobs. Repeat jobs with a "permissible" start date are not subject to any further restrictions and are, for example, also started after the specified date if this was not possible earlier because of delays in the job scheduler.
The date specification consists of the day and the time:

DATE = <date>

Date. This can be specified in the form [yy]yy-mm-dd. Only the last two digits of the year are evaluated, which means that the century is ignored in four-digit year specifications. 20 is automatically prefixed to two-digit year specifications < 80, 19 to two-digit year specifications ≥ 80.

TIME = <time>

Time of day.

RESOURCES = *PARAMETERS(...)

Values for run priority, CPU time and maximum number of SYSLST records.

RUN-PRIORITY = *STD / <integer 30..255>

Run priority the batch job is to be given. The lower the value, the higher the priority. The maximum permissible priority value is the lesser of the two values (i.e. the more favorable of the values) from the user catalog and the job class definition.
If no maximum value is defined for the job class, the following rules apply:

- If the value specified explicitly is numerically lower than the value in the user entry, the message JMS0045 is issued. The batch job is assigned the higher of the two values (i.e. the less favorable value) for the run priority from the user entry and the default run priority for the job class.
- If no value is specified or if *STD or *BY-JOB-CLASS is explicitly specified, the batch job is assigned the standard run priority of the job class.

The values can be queried with the SHOW-USER-ATTRIBUTES and SHOW-JOB-CLASS commands.

RUN-PRIORITY = *STD

The standard priority specified for the job class applies.

CPU-LIMIT = *STD / *NO / <integer 1..32767 *seconds*>
Maximum CPU time, in seconds, that the batch job may consume. The maximum time permitted depends on the job class specified. See also section “Time limits in BS2000” on page 1-103.

CPU-LIMIT = *STD
The default value for the chosen job class applies.

CPU-LIMIT = *NO
The ENTER job is to run with no time limit (NTL). This operand value is permitted only if the requisite authorization exists in the user entry or the job class definition.

SYSLST-LIMIT = *STD / *NO / <integer 0..999999>
Designates the maximum number of records output by the job to the system files SYSLST, SYSLST01, SYSLST02, ..., SYSLST99. Data records in the system file SYSOUT that are simultaneously written to SYSLST are not counted.
This value must not be above the limit set in the job class definition. This limit may be queried using the SHOW-JOB-CLASS command.

SYSLST-LIMIT = *STD
The default value for the chosen job class applies. If the specified number is exceeded:

– in batch mode, the job is terminated abnormally
– in interactive mode, the user may specify whether the job is to be continued or terminated. If continued, output is limited by “number” again.

SYSLST-LIMIT = *NO
Sets no limit on the number of records output.

LISTING = *NO / *YES
Specifies whether the job run is also to be logged on SYSLST.

LOGGING =
Controls logging of job execution. For non-S procedures the LOGGING operand is ignored because logging is defined in the BEGIN-PROCEDURE in that instance.

LOGGING = *STD
Logging only takes place if the procedure file is not read-protected.

LOGGING = *YES
Job execution is logged. Execution of a file can only be logged if the correct password has been entered in the password table of the job originator by means of the ADD-PASSWORD command or is specified in the PROCEDURE-PASSWORD operand.

LOGGING = *NO
Job execution is not logged.

JOB-PARAMETER =
Specifies additional attributes for the selected job class - assuming that systems support has defined some and made them known.

JOB-PARAMETER = *NO
No additional attributes.

JOB-PARAMETER = <c-string 1..127>
Arbitrary sequence of characters defined by the system administrator to identify additional job class attributes.

SYSTEM-OUTPUT =
Controls output of the system files SYSLST and SYSOUT on job termination (see also the EXIT-JOB command, SYSTEM-OUTPUT operand).

SYSTEM-OUTPUT = *STD
Outputs the system files SYSLST and SYSOUT to printer or in an email (see *STDOUT) in the event of errored execution.

SYSTEM-OUTPUT = *STDOUT
Depending on the setting of the system parameter SSMOUT, the system files SYSLST and SYSOUT are output to printer (see *PRINT) or sent by email (see *MAIL). In the case of cross-system command processing, the system parameter of the target system is evaluated.

SYSTEM-OUTPUT = *PRINT
Outputs the system files SYSLST and SYSOUT to printer.

SYSTEM-OUTPUT = *MAIL
The system files SYSLST and SYSOUT are sent by email. The receiver address(es) is/are taken over from the user entry of the user ID of the terminated job. How the receiver address is selected from an address list in accordance with the job name is described under the MAIL-FILE command.
If transfer by email is not possible (e.g. no email address in the user ID), the system files are output to printer.

SYSTEM-OUTPUT = *DELETE
Output of the system files SYSLST and SYSOUT is suppressed.

ASSIGN-SYSTEM-FILES =
Specifies which allocation should apply to the system files SYSLST and SYSOUT at the start of the batch job. The operand supports the assignment of cataloged files to the two system files during an asynchronous procedure run. Consequently the procedure file can, for instance, continue output to SYSOUT in interactive mode while directing output to a cataloged file in batch mode. The assignment to dummy files (*DUMMY) is also supported (see also FILE-NAME=*DUMMY in the ADD-FILE-LINK command).

ASSIGN-SYSTEM-FILES = *STD
The primary allocation for SYSLST and SYSOUT is preset, i.e. output on printer at the end of the task (dependent on SYSTEM-OUTPUT), if there has been no change in allocation within the procedure.

ASSIGN-SYSTEM-FILES = *PARAMETERS(...)
Specifies which allocation should apply to SYSLST and SYSOUT at the start of the batch job.

> **SYSLST = *STD / *PRIMARY / *DUMMY / <filename 1..54>**
> Output destination to which the system file SYSLST is to be assigned. The default value is *STD, i.e. the existing assignment is unchanged.
>
> **SYSOUT = *STD / *PRIMARY / *DUMMY / <filename 1..54>**
> Output destination to which the system file SYSOUT is to be assigned. The default value is *STD, i.e. the existing assignment is unchanged.

PROTECTION = *NONE / *CANCEL
Specifies whether the batch job is to be protected against accidental termination with the CANCEL-JOB command.

PROTECTION = *NONE
The batch job is not protected against accidental termination.

PROTECTION = *CANCEL
The batch job is not protected against accidental termination. In interactive jobs and on consoles which wish to terminate this batch job with the CANCEL-JOB command, the system additionally requests confirmation. Accidental termination of the batch job due to incorrect specification of the job number should thus be prevented.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
2	0	CMD0002	Command executed with warning, e.g. DELETE=*YES ignored for repeat job or when a library member is specified
	1	CMD0202	Syntax error in command
	32	CMD0221	system error
	64	JMS0630	Semantic or privilege error, e.g. host, catalog ID, job class unknown; MONJV not accessible, maximum number of jobs reached
	64	JMS0640	Error when accessing the procedure file, e.g. not a SAM or ISAM file, file empty, no access right
	64	JMS0670	Error in a remote job
	130	JMS0620	No further storage space or TSN available, or specified MONJV already monitoring a job
	130	JMS0650	MSCF non available or no connection to the specified host

Notes

– Combinations of the START and REPEAT-JOB operands:

	REPEAT-JOB		
START	AT-STREAM-STARTUP	DAILY or WEEKLY	PERIOD
IMMEDIATELY or SOON	a)	c)	c)
AT or EARLIEST	a)	d)	f)
LATEST or WITHIN	a)	c)	g)
AT-STREAM-STARTUP	b)	e)	e)

Table 50: START and REPEAT-JOB operand combinations in the ENTER-PROCEDURE command

a) The first and all subsequent starts of the job take place as specified.

b) The first start of the job is effected with START=*AT-STREAM-STARTUP. All further starts take place after the startup of the job scheduler with START=*SOON.

c) The base time for the repetition cycle is the time the job is accepted.

d) The specified point in time (START=...., TIME=....) is the base time for the repetition cycle.

e) The first start of the job follows startup of the job scheduler. This point in time is the base time for the repetition cycle. All further starts take place with START=*SOON.

f) The specified point in time (START=...., TIME=....) is the base time for the repetition cycle. The second and all further starts take place with START=*SOON.

g) The base time for the repetition cycle is the time the job is accepted. All further starts take place with START=*SOON.

- The following applies to the *WITHIN, *AT and *LATEST entries in the START operand: After the specified point in time or period of time, jobs that have not been started are treated in the same way as jobs started with START=*SOON and highest job priority.

  Example

  A job with START=*LATEST(...) could not be started by the desired time because the job scheduler was not active. It will then be started (within the same session) as soon as possible after the next startup of the job scheduler.

- Determining the scheduling time of a calendar job:

  - In the first version of the calendar job, the symbolic date (SYMDAT) specified in the SYMBOLIC-DATE operand is passed on to the CALENDAR component in the evaluation of the job attributes. The CALENDAR component returns the next point in time (date and time), with regard to the current point in time, specified by the SYMDATs defined in the calendar file.
    In the case of partially qualified SYMDATs, a scheduling time is returned for each SYMDAT beginning with the character string, and the calendar job is started at each of these scheduling times.
  - The scheduling times of the following versions are determined according to the same procedure while the previous jobs are processed.

  As a consequence, any modifications made to the calendar file only take effect on calendar job versions whose scheduling time is determined after the update of the calendar file.
  In particular, the number of calendar job versions started by means of a partially qualified SYMDAT can be extended (by defining new SYMDATs) or reduced (by deleting SYMDATs).

- A batch job which is to run on a remote system can be accessed via a MONJV only when the catalog ID of the partner system’s home pubset is entered in the MRSCAT of the systems concerned.

- Access to procedure files via RFA is not possible.

- So as to ensure availability of the procedure file at execution time (start of the batch job), a copy of the file with the prefix S.PROC. is always created.

- Essentially the operands are identical with those of the ENTER-JOB command. Only the following operands of the ENTER-JOB command are not supported:

FILE-PASSWORD	The generated ENTER file is assigned a random password with which the FILE-PASSWORD is supplied.
DELETE	DELETE=*YES always applies

- The ENTER file created is always started with DELETE=*YES. The ENTER file and the copy of the procedure file (S.PROC file) are not automatically deleted if the job is to be repeated (REPEAT operand).
- The copy of the procedure file (S.PROC file) is password-protected, but may be deleted by the owner without entry of the password. The user can thus delete S.PROC files that were not deleted on account of a system error. The user may not delete S.PROC files for repeat jobs. The same applies to password protection on the S.E file.
- The S.PROC and S.E files are set up with DESTROY-BY-DELETE=*YES depending on the system parameter DESTLEV.
  Procedure files can be protected by means of read, write and execute passwords. The execute password or a higher-ranking password must be specified in the PROCEDURE-PASSWORD operand.
  A read password must be specified if job execution is to be logged.
  The passwords are validated on processing the ENTER-PROCEDURE command. If the passwords are subsequently changed, the successful validation during ENTER-PROCEDURE processing remains in force and the procedure file can be executed. If the procedure to be executed is a PLAM library element, a password specified in the PROCEDURE-PASSWORD operand is interpreted as a password only for accessing the PLAM library.
- S.IN files are always unencrypted. It must therefore be possible to encrypt the procedure file when the command is processed, i.e. the crypto password must be specified in the CRYPTO-PASSWORD operand.
- When the batch job is started on a remote host (HOST not equal to *STD), all the passwords used (CRYPTO-PASSWORD, PROCEDURE-PASSWORD and JV-PASSWORD) must be specified directly in the ENTER-PROCEDURE command. Passwords which were entered in the job submitting task's password list with the ADD-PASSWORD and ADD-CRYPTO-PASSWORD commands apply only when the job is started locally (HOST = *STD).
- The S.PROC and the S.E files are protected by a file lock for the duration of the batch job.
  Note the following: the file locks are set when the pubset is imported on which the files are located. The file locks only consider files to which batch jobs from the current job pool (on the home pubset) refer. If the files are located on a shared pubset, the file locks are coordinated from the master computer.

– In nonprivileged job classes, up to 32767 waiting jobs are permissible. Any ENTER jobs in excess of this number are rejected.

EOF

Generate EOF condition for SYSDTA system input file

Description status:	SYSFILE V19.0A
Functional area:	Job processing File processing
Domain:	JOB PROCEDURE PROGRAM
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The EOF command generates an EOF condition for the system input file SYSDTA. In interactive mode, the EOF command in conjunction with the ESCAPE function (e.g. K2 key) terminates the reading in of data at the terminal (SYSDTA = SYSCMD).
In a user program, the EOF command terminates data input (SYSDTA = input file) if the data record that has just been read is equivalent to an EOF command and a corresponding instruction is used in the program. Control then passes to the routine for end-of-file processing.

Format

EOF

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
	32	EXC0041	System error
	130	CMD2282	Command not executed because no program is loaded

EXECUTE-POSIX-CMD

Call POSIX commands from BS2000

Description status:	POSIX-BC V10.0A
Functional area:	POSIX administration and application
Domain:	PROCEDURE
Privileges:	STD-PROCESSING

Function

The EXECUTE-POSIX-CMD command enables the user to call POSIX commands directly in BS2000 without first calling a POSIX shell explicitly to permit their execution. The user can specify a single command, a list of commands (command sequence) or the name of a BS2000 file which contains a script.
The user must satisfy the requirements for calling a POSIX shell (see the START-POSIX-SHELL command).

The commands specified are forwarded to a POSIX shell without a syntax check being performed and called using the dot command. The environment variable `EXECUTE_POSIX_CMD=“YES“` is set in the shell. Querying variables (e.g. in */etc/profile* or *.profile*) enables unwanted outputs to be suppressed when EXECUTE-POSIX-CMD is executed.

Processing of the EXECUTE-POSIX-CMD command ends only when all the commands which have been passed have been processed (i.e. all background processes which have been started are terminated). Processing can be aborted by pressing the K2 key twice.

The POSIX commands specified can optionally be logged in a BS2000 file (INPUT-LOG-FILE operand) and, if required, can be called again from this file.

By default the command’s outputs are directed to SYSOUT (OUTPUT operand), but they can also be written to a file.

Details of permissible inputs (scope of the POSIX commands supported) and general information about working with the POSIX shell are provided in the “POSIX (Commands)” manual [28].

Format

```
EXECUTE-POSIX-CMD                                                          Alias: ECXCMD

 CMD = <filename 1..54> / list-poss(15): <c-string 1..100 with-low>
,INPUT-LOG-FILE = *NONE / <filename 1..54 without-gen-vers>(...)
   <filename 1..54 without-gen-vers>(...)
     |  WRITE-MODE = *REPLACE / *EXTEND
,OUTPUT = *SYSOUT / <filename 1..54>
```

Operands

CMD=
Specifies the commands or scripts to be executed.

CMD = <filename 1..54>
Name of the BS2000 file from which the commands/command sequences are to be read.

CMD = list-poss (15): <c-string 1..100 with-low>
The command is specified explicitly. Multiple commands can be specified within a list in the form of a command sequence.

INPUT-LOG-FILE =
Specifies whether the inputs are to be logged.

INPUT-LOG-FILE = *NONE
The inputs are not logged.

INPUT-LOG-FILE = <filename 1..54 without-gen-vers>(...)
The inputs are logged in the specified BS2000 file.

> **WRITE-MODE = *REPLACE / *EXTEND**
> Specifies whether a file that has already been cataloged is to be overwritten or updated.

OUTPUT =
Specifies the place to which the command outputs are to be directed.

OUTPUT = *SYSOUT
Output is directed to the SYSOUT system file (i.e. in interactive mode, output on the monitor if nothing else is assigned).

OUTPUT = <filename 1..54>
The command outputs are to be written to the specified BS2000 file.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
x	64	CCM0999	The shell command, command sequence or script supplies an exit status with the value x (≠0) which can be obtained from the SC2.

Notes

- Commands/scripts which are executed using EXECUTE-POSIX-CMD cannot read from the standard input because the latter is closed before a command/script is executed. Consequently such commands/scripts receive EOF when they attempt to read from the standard input.
  The POSIX commands which may under some circumstances read from the standard input (in the case of queries) are: rm ,mv, bs2cp, mailx
- In the case of EXECUTE-POSIX-CMD, *stdout* and *stderr* are not connected to a terminal but to a pipe. Commands/scripts which require *stdout* and *stderr* to be connected to a terminal therefore work either not at all or incorrectly. These commands/scripts use the CRTE functions *isatty()* and *ttyname()* to determine whether or to which terminal *stdout* and *stderr* are connected.
  The POSIX commands which consequently might not work at all or only incorrectly are: tty, tabs, mesg, write, talk, more, patch, pax, nohup, ls, fg, bg
- When the shell command *exec* is executed with EXECUTE-POSIX-CMD, the current shell is unloaded and the mechanisms for forwarding outputs and/or the exit value of forked processes may possibly be disabled.
- The *fc* command is only effective for inputs outside scripts; it is therefore not suitable for use under /EXECUTE-POSIX-CMD.
- The shell commands executed using EXECUTE-POSIX-CMD are not logged in the usual command memory (*$HOME/.sh_history*) but in a separate short command memory (*HISTSIZE=100*) under */tmp/.ecxcmd_sh_history_<user-name>*.
  As a result, the *fc* command only accesses the history of the EXEC-POSIX-CMD calls.
- Under EXECUTE-POSIX-CMD, command substitutions by means of *’command’* or *$(command)* are always executed in a subshell. In the POSIX shell, on the other hand, a number of commands are available which are substituted within the shell itself. Consequently individual commands can behave differently from in the POSIX shell if the results are process-specific (e.g. *ftyp* and *bs2file* and accesses to variables or functions which have not been exported).

- In the command sequence the definition of alias names using the *alias* command has no effect because the command sequence is executed using the dot command. If alias names are to be defined and used, the command sequence must be copied to a (temporary) POSIX file. The file must be assigned Execute permission and executed (not using the dot command).
- The command sequence called using EXECUTE-POSIX-CMD is executed in a subshell which is generated internally by means of *fork*. The SYSFILE environment of the calling procedure is not available in this subshell. This can have an influence on BS2000 commands which are called using *bs2cmd* and on the POSIX commands *lp*, *lpstat* and *cancel*.

EXIT-JOB

Terminate current job

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing
Domain:	JOB
Privileges:	all privileges
Routing code:	@ (only with NBCONOPI=Y)

Function

The EXIT-JOB command terminates the current job. The virtual memory pages and devices occupied by the job are subsequently released and the system output files are readied for output.
If new file generations have been created during the job, the system outputs the names of the file generations concerned, their base values, and the names of the first and current file generations.

> Unlike the LOGOFF command, EXIT-JOB does not support output of the system files to tape.

Notes on Operator LOGON

If the “Operator LOGON” function is used (incompatible mode; system parameter NBCONOPI=Y), the EXIT-JOB command can also be issued on a physical operator terminal (console) or from an authorized user program with dynamic authorization names, but operands are not allowed in either case.

On a physical operator terminal the EXIT-JOB command revokes the authorization to enter commands; with an authorized user program it results in disconnection.
In both cases EXIT-JOB also implies the RELEASE-OPERATOR-ROLE command with OPERATOR-ROLE=*ALL, which means that all routing codes are relinquished and message subscription/suppression is canceled.

Operator activities on a physical operator terminal cannot be resumed until authentication has been carried out again using SET-LOGON-PARAMETERS under a user ID with the OPERATING privilege.

Format

```
EXIT-JOB

 MODE = *NORMAL / *ABNORMAL
,SYSTEM-OUTPUT = *ALL / *NONE / *STDOUT / *PARAMETERS(...)
    *PARAMETERS(...)
       | SYSLST-OUTPUT = *NONE / *PRINTER / *MAIL / *STDOUT
       | SYSOUT-OUTPUT = *NONE / *PRINTER / *MAIL / *STDOUT
,KEEP-CONNECTION = *NO / *YES
```

Operands

MODE =
Specifies whether the job is to be terminated normally or abnormally.

MODE = *NORMAL
The job is terminated normally. The status indicator of a monitoring job variable is set to '$T'.

MODE = *ABNORMAL
The job is terminated abnormally. The status indicator of a monitoring job variable is set to '$A'. Whether or not a dump is taken depends on the test options defined within the task (see MODIFY-TEST-OPTIONS command, DUMP operand).

SYSTEM-OUTPUT =
Controls output of the system files SYSLST and SYSOPT, and for batch jobs also output of the system file SYSOUT. Empty system files are not output.

SYSTEM-OUTPUT = *ALL
The system files SYSLST and SYSOUT are output to printer (see *PRINTER) or transferred by email (see *MAIL) depending on the settings of the system parameter SSMOUT.

SYSTEM-OUTPUT = *NONE
Output of the system files is suppressed.

SYSTEM-OUTPUT = *STDOUT
Depending on the setting of the system parameter SSMOUT, the system files SYSLST and SYSOUT are output to printer (see *PRINTER) or sent by email (see *MAIL).

SYSTEM-OUTPUT = *PARAMETERS(...)
Controls which of the two system files (SYSLST or SYSOUT) is output.

SYSLST-OUTPUT = *NONE
SYSLST is not output.

SYSLST-OUTPUT = *PRINTER
The system file SYSLST is to be output to printer. However, the output can be suppressed by systems support using the system parameters SSMLOGF1 and SSMLGOF2. The following applies:

– `SSMLGOF1=NO-SPOOL`
  In batch jobs output is suppressed.
  In interactive jobs the output is suppressed if SSMLGOF2=NO. Output takes place when SSMLGOF2=SPOOL and the SPOOL subsystem is available. In all other cases an inquiry is displayed on the terminal.

– `SSMLOGF1=REQ-SPOOL`
  Output is to printer. If SPOOL is not available, the job waits and the status in a job-monitoring MONJV remains $R. Only when SPOOL is available and printout can take place does the job terminate.

– `SSMLOGF1=STA-SPOOL`
  Printout takes place if SPOOL is available, otherwise it is suppressed.

SYSLST-OUTPUT = *MAIL
SYSLST is sent by email. The receiver address(es) is/are taken over from the user entry of the job’s user ID. How the receiver address is selected from an address list in accordance with the job name is described under the MAIL-FILE command.
If transfer by email is not possible (e.g. no email address in the user ID), output takes place in accordance with SYSLST-OUTPUT=*PRINTER.

SYSLST-OUTPUT = *STDOUT
Depending on the setting of the system parameter SSMOUT, the system file SYSLST is output to printer (see *PRINT) or sent by email (see *MAIL).

SYSOUT-OUTPUT = *NONE
SYSOUT is not output.

SYSOUT-OUTPUT = *PRINTER
The system file SYSOUT is to be output to printer. However, the output can be suppressed by systems support (see SYSLST-OUTPUT=*PRINTER).

SYSOUT-OUTPUT = *MAIL
SYSOUT is sent by email. The receiver address(es) is/are taken over from the user entry of the job’s user ID. How the receiver address is selected from an address list in accordance with the job name is described under the MAIL-FILE command.
If transfer by email is not possible (e.g. no email address in the user ID), output takes place to printer.

SYSOUT-OUTPUT = *STDOUT
Depending on the setting of the system parameter SSMOUT, the system file SYSOUT is output according to SYSOUT-OUTPUT=*PRINTER.

KEEP-CONNECTION =
Specifies whether the processor connection is to be retained, enabling a new job to be started immediately.
This operand is permitted only in interactive mode; it is ignored in batch mode.

KEEP-CONNECTION = *NO
Clears down the processor connection.

KEEP-CONNECTION = *YES
Retains the processor connection. A new job can be started immediately.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed [1]
	1	CMD0202	Syntax error
	32	CMD0221	System error; no message is sent to SYSOUT

[1] Cannot be evaluated since in this case the task is terminated!

Note

If an EXIT-JOB command is issued while a program is loaded, an ABEND-STXIT routine (if defined) will be activated. This does not happen if the EXIT-JOB command is called via the CMD macro (see the “Executive Macros” manual [22]).

Examples

```
/mod-test-opt dump=*std
/exit-job mode=*abnormal,keep-conn=*yes
%  IDA0N51 PROGRAM INTERRUPT AT LOCATION '00000000 (NONAME), (CDUMP), EC=50'
%  IDA0N45 DUMP DESIRED? REPLY (Y=USER-/AREADUMP TO DISK;
Y,<VOLUMETYPE>=USER-/AREADUMP TO TAPE; Y,SYSTEM=SYSTEMDUMP; N=NO)?n
%  EXC0419 /LOGOFF AT 1621 ON 12-01-26 FOR TSN '9VKP'
%  EXC0421 CPU TIME USED: 0.2166
%  JMS0150 INSTALLATION ' S190-40', BS2000 VERSION 'V180', HOST 'D016ZE07':
PLEASE ENTER '/SET-LOGON-PARAMETERS' OR '?'
```

*Example 2: Abnormal job termination and test option DUMP=*YES*

```
/mod-test-opt dump=*yes
/exit-job mode=*abnormal,keep-connection=*yes
%  IDA0N51 PROGRAM INTERRUPT AT LOCATION '00000000 (NONAME), (CDUMP), EC=50'
%  IDA0N53 DUMP BEING PROCESSED. PLEASE HOLD ON
%  IDA0N54 'USERDUMP' WRITTEN TO FILE '$USER1.DUMP.9VKP.00001'
%  IDA0N55 TITLE: 'TSN-9VKP UID-USER1  AC#-89001  USERDUMP  PC-00000000  EC-
50  VERS-150  DUMP-TIME  16:21:26 12-01-26'
%  EXC0419 /LOGOFF AT 1621 ON 12-01-26 FOR TSN '9VKP'
%  EXC0421 CPU TIME USED: 0.2166
%  JMS0150 INSTALLATION ' S190-40', BS2000 VERSION 'V180', HOST 'D016ZE07':
PLEASE ENTER '/SET-LOGON-PARAMETERS' OR '?'
```

EXIT-PROCEDURE

Terminate procedure run (return to caller)

Description status:	SYSFILE V19.0A
Functional area:	Procedures
Domain:	PROCEDURE
Privileges:	STD-PROCESSING OPERATING HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The EXIT-PROCEDURE command is an SDF-P control flow command.It terminates procedure execution and returns control to the caller, i.e. the procedure level from which the call was made. If EXIT-PROCEDURE terminates the last procedure, control returns to the primary command input (corresponding to procedure level 0):
in interactive mode, command input (SYSCMD) is assigned primarily to the terminal, in batch mode to the ENTER file.
The system files (including TASKLIB) receive the assignment in force at the interrupt point. A file that was assigned by means of ASSIGN-SYSLST in the procedure is closed by EXIT-PROCEDURE.

The ERROR operand can be used to pass a command return code to the caller. This code can be evaluated with SDF-P facilities (see IF-BLOCK-ERROR command).
The RESUME-PROGRAM operand can be used to specify that a loaded program may be continued after termination of the procedure. This function corresponds to the ENDP-RESUME command.

If an EOF condition is encountered (physical end of the procedure file is reached during command processing), the procedure is terminated as if an EXIT-PROCEDURE command had been issued. The caller receives a return code corresponding to the current error status on termination.
The effect of an EXIT-PROCEDURE command in ESCAPE mode is described under the HOLD-PROCEDURE command.

If errors occur in an *S procedure*, the EXIT-PROCEDURE command is *not* recognized.

Format

```
EXIT-PROCEDURE

ERROR = *NO (...) / *YES(...)

   *NO(...)
     |
     |    SUBCODE2 = 0 / <integer 0..255>
     |
     |   ,MAINCODE = CMD0001 / <alphanum-name 7..7>

   *YES(...)
     |
     |    SUBCODE1 = 64 / <integer 0..255>
     |
     |   ,SUBCODE2 = 0 / <integer 0..255>
     |
     |   ,MAINCODE = SDP0018 / <alphanum-name 7..7>

,RESUME-PROGRAM = *NO / *YES
```

Operands

ERROR =
This specifies which information on procedure execution the caller is to receive.

ERROR = *NO(...)
The caller receives the return code of class “NO-ERROR”. Additional information can be passed via the SUBCODE2 and MAINCODE operands:

> **SUBCODE2 = 0 / <integer 0..255>**
> This passes additional information in SUBCODE2. The default value is zero, i.e. there is no additional information.
>
> **MAINCODE = CMD0001 / <alphanum-name 7..7>**
> This passes a message code. The caller can request an explanation of the message by means of the HELP-MSG-INFORMATION command.
> The default value is CMD0001, i.e. procedure terminated without errors.

ERROR = *YES(...)
The caller receives a return code indicating an error. The operands SUBCODE1, SUBCODE2 and MAINCODE can be used to identify the error class and any additional information:

> **SUBCODE1 = 64 / <integer 0..255>**
> This indicates the error class of the relevant error.
>
> SUBCODE1=0 is also possible for reasons of compatibility. Because the convention (see “Return codes” on page 1-66) specifies that this value stands for “no error”, ERROR=*YES(SUBCODE1=0) is executed like ERROR=*NO.

SUBCODE2 = 0 / <integer 0..255>
This passes additional information in SUBCODE2. The default value is zero, i.e. there is no additional information.

MAINCODE = SDP0018 / <alphanum-name 7..7>
This passes a message code. The caller can request an explanation of the message by means of the HELP-MSG-INFORMATION command.

RESUME-PROGRAM = *NO / *YES
This specifies whether a loaded program is to be resumed when the procedure is terminated. If *YES is specified, the most recently loaded program can be resumed when the procedure is terminated.

Note

If a procedure is called in ESCAPE mode, the following message is issued when control is returned:

```
TASK IS IN ESCAPE-MODE AT LEVEL NUMBER i
```

where i = number of the procedure level to which processing has returned.

When primary command input is reached (level 0), this message is not issued.

Return codes

With the ERROR operand, the EXIT-PROCEDURE command can report any command return code to the caller. From the caller’s viewpoint this is the return code from the CALL-PROCEDURE or INCLUDE-PROCEDURE command. However, if execution of the EXIT-PROCEDURE command itself results in an error, control does not return to the caller but one of the following return codes is supplied and error handling within the procedure is activated.

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
2	0	SSM2039	Error on closing output system file; the SYSOUT message contains the DMS error code as an insert
	1	CMD0202	Syntax error
	3	CMD2203	Incorrect syntax file
	32	CMD0221	System error (internal error)
	64	SSM1013	No procedure has been called

Examples

See the CANCEL-PROCEDURE and HOLD-PROCEDURE commands.

EXPORT-FILE

Delete catalog entry of files (file export)

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION TSOS
Routing code:	$ (bei NBCONOPI=N) bzw. E (bei NBCONOPI=Y)

Function

The EXPORT-FILE command deletes the catalog entries for files which are contained on a private volume or Net-Storage volume (i.e. exports files). The storage space and data in the files is retained.

By default the command exports files to private disk. When files on a Net-Storage volume are to be exported, either the volume or the storage type *NET-STORAGE must be specified explicitly.

The IMPORT-FILE command allows their catalog entries to be recreated (i.e. the files to be imported).

Privileged functions

System support staff may export files belonging to any user ID. It may ignore the passwords of protected files and also branch to guided dialog whenever the user ID is changed.

By default, systems support (TSOS privilege) is a co-owner of all the files (and can therefore export files under any user ID). When SECOS is used, this co-ownership can be restricted.

Systems support can generate files under any user ID (TSOS privilege).In conjunction with the SECOS software product a user can allow other user IDs to act as co-owners. Co-owners of a user ID are then also allowed to export a file.

Overview of functions

	Function / Meaning	Level 1 operands	Level 2/3 operands
3-297	Identifier of the volume to be exported	VOLUME	
3-297	Name of the file or file generation to be exported	FILE	
	– file name	=*NAME	
	– file generation; all older generations are exported	=*GENERATIONS-BEFORE	
	– file generation; all younger generations are exported	=*GENERATIONS-AFTER	
3-298	Selection criteria for the files to be deleted	SELECT=*BY-ATTRIBUTES	
	Date specification		
3-298	– creation date		CREATION-DATE
3-300	– Expiration date (implicit retention period)		EXPIRATION-DATE
3-301	– date of last access		LAST-ACCESS-DATE
3-312	– Number of accesses to the file		ACCESS-COUNTER
3-303	– Type of volume		SUPPORT
	– – Net-Storage, Private disk, Tape		=*ANY
	– – Private disk		=*PRIVATE-DISK
	– – tape		=*TAPE
	Storage space		
3-303	– Storage type		STORAGE-TYPE
3-303	– – File type (on Net-Storage)		FILE-TYPE
3-304	– Reserved storage space		SIZE
3-305	– Number of extents		NUMBER-OF-EXTENTS
3-305	– Size of reserved storage space not yet used		NUMBER-OF-FREE-PAGES
3-313	– Number of the last page used (last page pointer)		HIGHEST-USED-PAGE
	File security/file protection		
3-306	– Access		ACCESS
3-307	– Shareability		USER-ACCESS
3-310	– Basic access control list		BASIC-ACL
3-311	– Highest activated access control		PROTECTION-ACTIVE

Table 51: Overview of the EXPORT-FILE command functions (Part 1 of 2)

	Function / Meaning	Level 1 operands	Level 2/3 operands
3-306	– Password protection		GUARDS
3-308	– BACKUP level		BACKUP-CLASS
3-313	– Type of file		TYPE-OF-FILES
	– Files only		=*FILE
	– PLAM libraries only		=*PLAM-LIBRARY
3-308	– File format		BLOCK-CONTROL-INFO
3-307	– Access method at creation		FILE-STRUCTURE
3-309	– Status of the file		STATUS
	– Closed		CLOSED-OUTPUT
	– Not closed properly and not yet reconstructed		REPAIR-NEEDED
3-309	– Ability to migrate from processing level (S0) to background level (S1 or S2) in conjunction with HSMS		MIGRATE
	Control parameters	DIALOG-CONTROL	
3-314	– User intervention not permitted (default setting for procedures and batch mode)	=*NO	
3-315	– Dialog when more than one file is referenced if the file name specified is not fully qualified (default setting for interactive mode)	=*MORE-THAN-ONE-FILE	
3-314	– Dialog when an error occurs	=*ERROR	
3-315	– Dialog when catalog ID changes	=*CATALOG-CHANGE	
3-314	– For each file selected, there will be an interactive dialog with the user to determine whether or not the current DELETE-FILE command should be processed.	=*FILE-CHANGE	
3-315	– Dialog when user ID changes	=*USER-ID-CHANGE	
3-315	Message for successfully exported files	OUTPUT	
	– Suppress	=*NO	
	– Output to SYSOUT	=*SYSOUT	
3-315	Ignore password protection	IGNORE-PROTECTION	
3-316	Ignore password protection provided by the specified passwords	PASSWORDS-TO-IGNORE	
3-316	Suppress error situations	SUPPRESS-ERRORS	

Table 51: Overview of the EXPORT-FILE command functions (Part 2 of 2)

Format

```
EXPORT-FILE

 VOLUME = *ANY / <alphanum-name 1..6>
,FILE = *NAME(...) / *GENERATIONS-BEFORE(...) / *GENERATIONS-AFTER(...)
   *NAME(...)
     |  FILE-NAME = *DUMMY / *ALL / <filename 1..54 with-wild(80)>
   *GENERATIONS-BEFORE(...)
     |  GENERATION-NAME = *DUMMY / <filename 1..54 with-wild(80)>
   *GENERATIONS-AFTER(...)
     |  GENERATION-NAME = *DUMMY / <filename 1..54 with-wild(80)>
,SELECT = *ALL / *BY-ATTRIBUTES(...)
   *BY-ATTRIBUTES(...)
     |  CREATION-DATE = *ANY / *NONE / *TODAY / *YESTERDAY / <integer -99999..991231> /
     |          <date> / [*INTERVAL](...)
     |    [*INTERVAL](...)
     |      |  FROM = *EARLIEST / *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
     |      | ,TO = *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
     | ,EXPIRATION-DATE = *ANY / *NONE / *TOMORROW / *TODAY / *YESTERDAY /
     |                    <integer -99999..991231> / <date> / [*INTERVAL](...)
     |    [*INTERVAL](...)
     |      |  FROM = *EARLIEST / *TOMORROW / *TODAY / *YESTERDAY /
     |      |             <integer -99999..991231> / <date>
     |      | ,TO = *LATEST / TODAY / *TOMORROW / *YESTERDAY /
     |      |         <integer -99999..991231> / <date>
     | ,LAST-ACCESS-DATE = *ANY / *NONE / *TODAY / *YESTERDAY / <integer -99999..991231> /
     |                     <date> / [*INTERVAL](...)
     |    [*INTERVAL](...)
     |      |  FROM = *EARLIEST / *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
     |      | ,TO = *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
```

(Part 1 of 3)

```
,SUPPORT = *ANY / list-poss(2): *PRIVATE-DISK / *TAPE

,STORAGE-TYPE = *ANY / *NET-STORAGE(...)

   *NET-STORAGE(...)
      FILE-TYPE = *ANY / *BS2000 / *NODE-FILE

,SIZE = *ANY / *FREESIZE / <integer 0..2147483647> / [*INTERVAL](...)

   [*INTERVAL](...)
      FROM = 0 / <integer 0..2147483647>
     ,TO = 2147483647 / <integer 0..2147483647>

,NUMBER-OF-EXTENTS = *ANY / <integer 0..65535> / [*INTERVAL](...)

   [*INTERVAL](...)
      FROM = 0 / <integer 0..65535>
     ,TO = 65535 / <integer 0..65535>

,NUMBER-OF-FREE-PAGES = *ANY / *SIZE / <integer 0..2147483647> / *ALL-ALLOCATED /
                        [*INTERVAL](...)

   [*INTERVAL](...)
      FROM = 0 / <integer 0..2147483647>
     ,TO = 2147483647 / <integer 0..2147483647>

,ACCESS = *ANY / *READ / *WRITE

,PASSWORD = *ANY / list-poss(4): *NONE / *READ-PASSWORD / *WRITE-PASSWORD /
                 *EXEC-PASSWORD

,USER-ACCESS = *ANY / list-poss(3): *OWNER-ONLY / *ALL-USERS / *SPECIAL

,FILE-STRUCTURE = *ANY / list-poss(5): *PAM / *SAM / *ISAM / *BTAM / *NONE

,BACKUP-CLASS = *ANY / list-poss(5): *A / *B / *C / *D / *E

,BLOCK-CONTROL-INFO = *ANY / list-poss(9): *NONE / *NO / *WITHIN-DATA-BLOCK /
                      *WITHIN-DATA-2K-BLOCK / *WITHIN-DATA-4K-BLOCK / *PAMKEY /
                      *NK / *NK2 / *NK4

,MIGRATE = *ANY / list-poss(3): *ALLOWED / *INHIBITED / *FORBIDDEN

,STATUS = *ANY / [*PARAMETERS](...)

   [*PARAMETERS](...)
      CLOSED-OUTPUT = *ANY / *YES / *NO
     ,REPAIR-NEEDED = *ANY / *YES
```

(Part 2 of 3)

```
        ,BASIC-ACL = *ANY / *NONE / *YES / [*PARAMETERS](...)

           [*PARAMETERS](...)
                OWNER = *ANY / *NO-ACCESS / [*PARAMETERS](...)

                   [*PARAMETERS](...)
                        READ = *ANY / *NO / *YES
                       ,WRITE = *ANY / *NO / *YES
                       ,EXEC = *ANY / *NO / *YES

               ,GROUP = *ANY / *NO-ACCESS / [*PARAMETERS](...)

                   [*PARAMETERS](...)
                        READ = *ANY / *NO / *YES
                       ,WRITE = *ANY / *NO / *YES
                       ,EXEC = *ANY / *NO / *YES

               ,OTHERS = *ANY / *NO-ACCESS / [*PARAMETERS](...)

                   [*PARAMETERS](...)
                        READ = *ANY / *NO / *YES
                       ,WRITE = *ANY / *NO / *YES
                       ,EXEC = *ANY / *NO / *YES

        ,PROTECTION-ACTIVE = *ANY / list-poss(2): *LEVEL-0 / *LEVEL-1

        ,ACCESS-COUNTER = *ANY / <integer 0..2147483647> / [*INTERVAL](...)

           [*INTERVAL](...)
                FROM = 0 / <integer 0..2147483647>

               ,TO = 2147483647 / <integer 0..2147483647>

        ,HIGHEST-USED-PAGE = *ANY / <integer 0..2147483647> / [*INTERVAL](...)

           [*INTERVAL](...)
                FROM = 0 / <integer 0..2147483647>

               ,TO = 2147483647 / <integer 0..2147483647>

        ,TYPE-OF-FILES = *ANY / *FILE / *PLAM-LIBRARY

,DIALOG-CONTROL = *STD / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE /
                                *CATALOG-CHANGE / *USER-ID-CHANGE

,OUTPUT = *STD / *NO / *SYSOUT

,IGNORE-PROTECTION = *NONE / list-poss(3): *WRITE-PASSWORD / *READ-PASSWORD /
                                 *EXEC-PASSWORD

,PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> /
                                    <integer -2147483648..2147483647>

,SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
```

(Part 3 of 3)

Operands

VOLUME = *ANY / <alphanum-name 1..6>
The VSN of the volume on which the file to be exported is stored. From the set of files selected by the FILE operand, only the files which are held on the specified volume will be exported. The VOLUME operand must not be user for file generation groups on tape.
If a file generation is to be exported from a private disk specified in the VOLUME operand, which holds a file generation group including the generation to be exported, but does not hold the group entry, gaps can result in the generation group. This gap will only be in the catalog entry; the file generation group remains complete on the disks.

VOLUME = *ANY
All the files identified by the FILE operand and contained on private volumes or Net-Storage volumes will be exported.

FILE = *NAME(...) / *GENERATIONS-BEFORE(...) / *GENERATIONS-AFTER
Specifies the files or file generation groups to be exported. *DUMMY designates a dummy file, or dummy generations which “always exist” and which satisfy all the selection criteria. Any other operands (except for DIALOG-CONTROL, OUTPUT and SUPPRESS-ERRORS) will be checked for their formal correctness, but otherwise ignored. The main use of *DUMMY is in testing procedures.

FILE = *NAME(...)
From within the volumes identified by VOLUME, the files named as follows will be selected:

> **FILE-NAME = *ALL / <filename 1..54 with-wild(80)> / *DUMMY**
> The name of the file to be exported. Only the user’s own user ID or a user ID for which the user is co-owner may be specified.

FILE = *GENERATIONS-BEFORE(...)
From within the volumes identified by VOLUME, all the file generations which are older than the one whose identification follows will be exported:

> **GENERATION-NAME = <filename 1..54 with-wild(80)> / *DUMMY**
> The name of a reference generation. All those file generations which have a number less than the reference generation will be exported.

FILE = *GENERATIONS-AFTER(...)
From within the volumes identified by VOLUME, all the file generations which are younger than the one whose identification follows will be exported:

> **GENERATION-NAME = <filename 1..54 with-wild(80)> / *DUMMY**
> The name of a reference generation. All those file generations which have a number greater than the reference generation will be exported.

SELECT = *ALL
All the files in the set specified by the FILE-NAME operand are selected.

SELECT = *BY-ATTRIBUTES(...)
Restricts the files in the set specified in the FILE-NAME operand to files which satisfy the criteria which follow. The default values, *ANY and ANY, each indicate that the file set is not to be restricted to any particular values for the attribute concerned.

CREATION-DATE = *ANY / *NONE / *TODAY / *YESTERDAY / <integer -99999..991231> / <date> / *INTERVAL(...)
The user can select the files to be exported by their creation date. See also the *CRE-DATE* and *CRE-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

CREATION-DATE = *ANY
The creation date is not to be used as a selection criterion.

CREATION-DATE = *NONE
Only those files which have the value NONE entered in the CREATION-DATE field in their catalog entry are exported, i.e. files which have never been opened.

CREATION-DATE = *TODAY
Only those files which have today's date entered in the CREATION-DATE field in their catalog entry are exported.

CREATION-DATE = *YESTERDAY
Only those files which have yesterday's date entered in the CREATION-DATE field in their catalog entry are exported.

CREATION-DATE = <integer -99999..991231>
Only those files which have the specified date entered in the CREATION-DATE field in their catalog entry are exported. Here, the user can specify the creation date in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value*: (6 digits, with preceding sign) the number of days from today's date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ ±0)

CREATION-DATE = <date>
Only those files which have the specified date entered in the CREATION-DATE field in their catalog entry are exported. The user can specify the creation date in the form [yy]yy-mm-dd.

CREATION-DATE = *INTERVAL(...)
Only those files which were created within the specified time period are exported. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified using <integer> values). It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). Whichever

of the operands is not specified will be replaced by the default value for use as the limit of the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = *EARLIEST / *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
Only those files which have been created since the specified date (i.e. CREATION-DATE ≥ specified date) are exported. If EARLIEST is specified, the system will use the earliest possible date.

FROM = *TODAY
Exports only those files which were created on the current date.

FROM = *YESTERDAY
Exports only those files which were created as of yesterday's date (i.e. files for which CREATION-DATE ≥ date of preceding day).

FROM = <integer -99999..991231>
Exports only those files which were created after the specified date (i.e. files for which CREATION-DATE ≥ specified date).

FROM = <date>
Exports only those files which were created after the specified date (i.e. files for which CREATION-DATE ≥ specified date).

TO = *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
Exports only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ specified date).

TO = *TODAY
Exports only those files which were created up to today's date (i.e. files for which CREATION-DATE ≤ current date).

TO = *YESTERDAY
Exports only those files which were created up to yesterday's date (i.e. files for which CREATION-DATE ≤ date of preceding day).

TO = <integer -99999..991231>
Exports only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ specified date).

TO = <date>
Exports only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ specified date).

EXPIRATION-DATE = *ANY / *NONE / *TOMORROW / *TODAY / *YESTERDAY / <integer -99999..991231> / <date> / *INTERVAL(...)
The user can select the files to be exported by their release date (expiration date). See also the *EXPIR-DATE* and *EXPIR-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

EXPIRATION-DATE = *ANY
The expiration date is not to be used as a selection criterion.

EXPIRATION-DATE = *NONE
Only those files which have no expiration date (i.e. the value NONE) entered in the EXPIRATION-DATE field in their catalog entry are exported.

EXPIRATION-DATE = *TOMORROW
Only those files which have tomorrow's date entered as the EXPIRATION-DATE in their catalog entry are exported.

EXPIRATION-DATE = *TODAY
Only those files which have today's date entered as the EXPIRATION-DATE in their catalog entry are exported.

EXPIRATION-DATE = *YESTERDAY
Only those files which have yesterday's date entered as the EXPIRATION-DATE in their catalog entry are exported.

EXPIRATION-DATE = <integer -99999..991231>
Only those files which have the specified date entered as the EXPIRATION-DATE in their catalog entry are exported. Here, the user can specify the expiration date in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value:* (6 digits with preceding sign), the number of days from today's date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ ±0 or TOMORROW ≙ +1)

EXPIRATION-DATE = <date>
Only those files which have the specified date entered as the EXPIRATION-DATE in their catalog entry are exported. The user can specify the expiration date in the form [yy]yy-mm-dd.

EXPIRATION-DATE = *INTERVAL(...)
Only those files for which the expiration date lies within the specified time period, i.e. files for which the retention period ends within the specified time period, will be exported. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified for EXPIRATION-DATE = <integer...>. It is also possible to specify limits using only the operand FROM (lower

limit) or TO (upper limit). Whichever of the operands is not specified will be replaced by the default value for use as the limit of the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = <u>*EARLIEST</u> / *TOMORROW / *TODAY / *YESTERDAY /
<integer -99999..991231> / <date>
If EARLIEST is specified, the system will use the earliest possible date. Only those files for which the retention period expires on or after the specified date (EXPIRATION-DATE ≥ specified date) are exported.

FROM = *TOMORROW
Exports only those files for which the EXPIRATION-DATE Ï the next day.

FROM = *TODAY
Exports only those files for which the EXPIRATION-DATE Ï the current date.

FROM = *YESTERDAY
Exports only those files for which the EXPIRATION-DATE Ï the specified date.

FROM = <integer -99999..991231>
Exports only those files for which the EXPIRATION-DATE Ï the specified date.

FROM = <date>
Exports only those files for which the EXPIRATION-DATE Ï the specified date.

TO = <u>*LATEST</u> / *TOMORROW / *TODAY / *YESTERDAY /
<integer -99999..991231> / <date>
Only files for which the retention period expires on or before the specified date (EXPIRATION-DATE ≤ specified date) will be exported.

TO = *TOMORROW
Exports only those files for which the EXPIRATION-DATE Î the next day.

TO = *TODAY
Exports only those files for which the EXPIRATION-DATE Î the current date.

TO = *YESTERDAY
Exports only those files for which the EXPIRATION-DATE Î the specified date.

TO = <integer -99999..991231>
Exports only those files for which the EXPIRATION-DATE Î the specified date.

TO = <date>
Exports only those files for which the EXPIRATION-DATE Î the specified date.

LAST-ACCESS-DATE = <u>*ANY</u> / *NONE / *TODAY / *YESTERDAY /
<integer -99999..991231> / <date> / *INTERVAL(...)
The user can select the files to be exported by the date when they were last accessed. See also the *ACC-DATE* and *ACC-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

LAST-ACCESS-DATE = *ANY
The date of last access is not to be used as a selection criterion.

LAST-ACCESS-DATE = *NONE
Exports only those files which have the value NONE entered in the LADATE field in their catalog entry, i.e. files which have never been opened.

LAST-ACCESS-DATE = *TODAY
Exports only those files for which today’s date has been entered as the LAST-ACCESS-DATE in the catalog entry.

LAST-ACCESS-DATE = *YESTERDAY
Exports only those files for which yesterday’s date has been entered as the LAST-ACCESS-DATE in the catalog entry.

LAST-ACCESS-DATE = <integer -99999..991231>
Exports only those files for which the specified date has been entered as the LAST-ACCESS-DATE in the catalog entry. Here, the user can specify the last access date in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value*: (6 digits, with preceding sign) the number of days from today’s date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ ±0)

LAST-ACCESS-DATE = <date>
Exports only those files for which the specified date has been entered as the LAST-ACCESS-DATE in the catalog entry. The user can specify the creation date in the form [yy]yy-mm-dd.

LAST-ACCESS-DATE = *INTERVAL(...)
Only files which were last accessed within the specified time period will be exported. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified using <integer> values). It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = *EARLIEST / *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
Only files which have been accessed since the specified date (i.e. for which LAST-ACCESS-DATE ≥ specified date) will be exported. If EARLIEST is specified, the system will use the earliest possible date.

FROM = *TODAY
Exports only those files which have been accessed up to the date of the current date.

FROM = *YESTERDAY
Exports only those files which have been accessed up to the date of the preceding day. Returns information on files for which the LAST-ACCESS-DATE ≥ the specified date.

FROM = <integer -99999..991231>
Exports only those files for which the LAST-ACCESS-DATE ≥ the specified date.

FROM = <date>
Exports only those files for which the LAST-ACCESS-DATE ≥ the specified date.

TO = *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
Only those files which were last accessed on or before the specified date (i.e. files for which LAST-ACCESS-DATE ≤ the specified date) are exported.

TO = *TODAY
Exports only those files which have been accessed up to the current date.

TO = *YESTERDAY
Exports only those files which have been accessed up to the date of the preceding day.

TO = <integer -99999..991231>
Exports only those files which were most recently accessed up to the specified date.

TO = <date>
Exports only those files which were most recently accessed up to the specified date.

SUPPORT = *ANY / list-poss(2): *PRIVATE-DISK / *TAPE
The user can select which files are to be exported by the EXPORT-FILE command by the type of their volumes.

SUPPORT = *ANY
The volume type is not to be used as a selection criterion.

SUPPORT = *PRIVATE-DISK
Only files on private disks are to be exported.

SUPPORT = *TAPE
Only files on tape or tape cartridge are to be exported.

STORAGE-TYPE = *ANY / *NET-STORAGE(...)
The user can select which files are to be exported with EXPORT-FILE by the type of the storage location of the file.

STORAGE-TYPE = *ANY
The storage type is not a selection criterion.

STORAGE-TYPE = *NET-STORAGE(...)
Only files on a Net-Storage volume are exported.

FILE-TYPE = *ANY / *BS2000 / *NODE-FILE
The user can select which files are to be exported with EXPORT-FILE by the file type.

FILE-TYPE = *ANY
The file type is not a selection criterion.

FILE-TYPE = *BS2000
Only BS2000 files on Net-Storage volumes are exported.

FILE-TYPE = *NODE-FILE
Only node files on Net-Storage volumes are exported.

SIZE = *ANY / *FREESIZE / <integer 0..2147483647> / *INTERVAL(...)
Only for files on disks and Net-Storage volumes: the user can select which files are to be exported by the EXPORT-FILE command on the basis of their size or the size of their reserved storage area (= number of PAM pages).
The value of the SIZE operand specifies the number of PAM pages: the upper and lower limits are both included in the range specified.

SIZE = *ANY
The file size is not to be used as a selection criterion.

SIZE = *FREESIZE
Only files for which storage space has been reserved but which do not yet occupy any of it (*HIGH-US-PA=0*) are to be exported.

SIZE = <integer 0..2147483647>
Only files for which the number of PAM pages reserved is exactly equal to the number specified here are to be exported.

SIZE = *INTERVAL(...)
Only files for which the number of pages lies within the specified range are to be exported. The upper and lower limits are both included in the range specified (see description of SIZE=<integer...>). It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit. Only files for which the number of PAM pages reserved is at least equal to the FROM value and at most equal to the TO value will be processed (FROM ≤ SIZE ≤ TO).

FROM = 0 / <integer 0..2147483647>
Only files for which the number of PAM pages reserved is at least equal to the specified number are to be exported (SIZE ≥ specified value).

TO = 2147483647 / <integer 0..2147483647>
Only files for which the number of PAM pages reserved is at most equal to the specified number are to be exported (SIZE ≤ specified value).

NUMBER-OF-EXTENTS = *ANY / <integer 0..65535> / *INTERVAL(...)
Only for disk files: the user can select the files to be processed by EXPORT-FILE by the number of extents into which the file is “split”. An extent is a contiguous area on a disk, occupied by one file; the output field *EXTENTS* shows how many extents the file has.

NUMBER-OF-EXTENTS = *ANY
The number of extents is not to be used as a selection criterion.

NUMBER-OF-EXTENTS = <integer 0..65535>
Only disk files with exactly the specified number of extents (*EXTENDS* = integer) are to be exported.

NUMBER-OF-EXTENTS = *INTERVAL(...)
Selects all the files whose number of extents lies within the specified range. The upper and lower limits are both included in the range specified. It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit. Only disk files which have at least as many EXTENTS as specified by the FROM value and at most as many as given by the TO value will be processed (FROM ≤ *EXTENTS* ≤ TO).

FROM = 0 / <integer 0..65535>
Only files for which the number of extents is at least equal to the specified number are to be exported (*EXTENTS* ≥ integer).

TO = 65535 / <integer 0..65535>
Only files for which the number of extents is at most equal to the specified number are to be exported (*EXTENTS* ≤ integer).

NUMBER-OF-FREE-PAGES = *ANY / *SIZE / <integer 0..2147483647> / *INTERVAL(...)
Only for files on disks and Net-Storage volumes: the user can select the files to be processed according to their number of free PAM pages, i.e. the number of pages reserved for a file but unused.

NUMBER-OF-FREE-PAGES = *ANY
The number of free PAM pages is not to be used as a selection criterion.

NUMBER-OF-FREE-PAGES = *SIZE
Only files which actually occupy no storage space are to be exported (i.e. no PAM page has been written yet).

NUMBER-OF-FREE-PAGES = <integer 0..2147483647>
Only files which have exactly the number of reserved but unused (=free) PAM pages specified by <integer...> are to be exported.

NUMBER-OF-FREE-PAGES = *INTERVAL(...)
Selects all files for which the number of unused PAM pages lies within the specified range. The upper and lower limits are both included in the range specified (see description of NUMBER-OF-FREE-PAGES = <integer...>). It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit.
Only disk files which have at least as many free PAM pages as specified by the FROM value and at most as many as given by the TO value will be exported (FROM ≤ FREE Î TO).

FROM = 0 / <integer 0..2147483647>
Only files for which the number free PAM pages is at least equal to the specified number are to be exported (FREE Ï integer).

TO = 2147483647 / <integer 0..2147483647>
Only files for which the number free PAM pages is at most equal to the specified number are to be exported (FREE Î integer).

ACCESS = *ANY / *READ / *WRITE
The specified access type will be used as a selection criterion (*ACCESS* output field).

ACCESS = *ANY
The ACCESS value is not to be used as a selection criterion.

ACCESS = *READ
Only files for which write access is prevented by ACCESS=READ, i.e. for which only read access is permitted, will be exported.

ACCESS = *WRITE
Only files for which write access is permitted will be exported.

PASSWORD = *ANY / list-poss(4): *NONE / *READ-PASSWORD / *WRITE-PASSWORD / *EXEC-PASSWORD
The user can select files with specific password types for processing by the EXPORT-FILE command.

PASSWORD = *ANY
Password protection is not to be used as a selection criterion.

PASSWORD = *NONE
Only files for which there is no password protection are to be exported.

PASSWORD = *READ-PASSWORD
Only files which are protected by a read password are to be exported.

PASSWORD = *WRITE-PASSWORD
Only files which are protected by a write password are to be exported.

PASSWORD = *EXEC-PASSWORD
Only files which are protected by an execute password are to be exported.

USER-ACCESS = *<u>ANY</u> / listposs(3): *OWNER-ONLY / *ALL-USERS / *SPECIAL
The user can select files using their access authorizations as a criterion.

USER-ACCESS = <u>*ANY</u>
The access authorization is not to be used as a selection criterion.

USER-ACCESS = *OWNER-ONLY
Only files which the owner alone may access are to be exported.

USER-ACCESS = *ALL-USERS
Only files which can be accessed by all users are to be exported.

USER-ACCESS = *SPECIAL
Only files which can be accessed by all user IDs including the maintenance IDs (i.e. user IDs with HARDWARE-MAINTENANCE privilege) are to be exported.

FILE-STRUCTURE = <u>*ANY</u> / list-poss(5): *PAM / *SAM / *ISAM / *BTAM / *NONE
The user can select files for processing according to their access method (*FILE-STRUC* output field).

FILE-STRUCTURE = <u>*ANY</u>
The access method is not to be used as a selection criterion.

FILE-STRUCTURE = *PAM
Only PAM files should be processed.

FILE-STRUCTURE = *SAM
Only SAM files are to be processed.

FILE-STRUCTURE = *ISAM
Only ISAM files are to be processed.

FILE-STRUCTURE = *BTAM
Only BTAM files are to be processed. BTAM files are tape files.

FILE-STRUCTURE = *NONE
Only files for which FILE-STRUC=NONE, i.e. files which have not yet been opened, are to be exported.

BACKUP-CLASS = <u>*ANY</u> / list-poss(5): *A / *B / *C / *D / *E
The user can select files for export by their BACKUP-CLASS level.

BACKUP-CLASS = <u>*ANY</u>
The BACKUP-CLASS level is not to be used as a selection criterion.

BACKUP-CLASS = *A
Only the files for which the value A is entered in the catalog as the BACKUP-CLASS (most frequent backup) are exported.

BACKUP-CLASS = *B
Only the files for which the value B is entered in the catalog as the BACKUP-CLASS are exported.

BACKUP-CLASS = *C
Only the files for which the value C is entered in the catalog as the BACKUP-CLASS are exported.

BACKUP-CLASS = *D
Only the files for which the value D is entered in the catalog as the BACKUP-CLASS are exported.

BACKUP-CLASS = *E
Only the files for which the value E is entered in the catalog as the BACKUP-CLASS are exported.

BLOCK-CONTROL-INFO = <u>ANY</u> / list-poss(4): NONE / NO / WITHIN-DATA-BLOCK / WITHIN-DATA-2K-BLOCK / WITHIN-DATA-4K-BLOCK / PAMKEY / NK / NK2 / NK4
The user can select files for export by their file format (*BLK-CONTR* output field).

BLOCK-CONTROL-INFO = *ANY
The BLOCK-CONTROL entry is not to be used as a selection criterion.

BLOCK-CONTROL-INFO = *NONE
Only files for which no BLK-CNTRL value has been defined, i.e. files which have not yet been opened, are to be exported.

BLOCK-CONTROL-INFO = *NO
Only files which contain no block control field are to be exported.

BLOCK-CONTROL-INFO = *WITHIN-DATA-BLOCK
Only files which were created with BLOCK-CONTROL-INFO=WITHIN-DATA-BLOCK (i.e. files for which block control information is held in a block control field at the start and within the data block) are to be exported.

BLOCK-CONTROL-INFO = *WITHIN-DATA-2K-BLOCK
Only files which were created with BLOCK-CONTROL-INFO=WITHIN-DATA-2K-BLOCK (i.e. files for which block control information is located at the start of each 2K block) are to be exported.

BLOCK-CONTROL-INFO = *WITHIN-DATA-4K-BLOCK
Only files which were created with BLOCK-CONTROL-INFO=WITHIN-DATA-4K-BLOCK (i.e. files for which block control information is located at the start of each 4K block) are to be exported.

BLOCK-CONTROL-INFO = *PAMKEY
Only files which use a separate PAM key for the block control field (i.e the block control information is held in a separate key field, outside the PAM block) are to be exported.

BLOCK-CONTROL-INFO = *NK
Only the NK files are to be exported, i.e. files which can also be stored on NK volumes (NK2 and NK4).

BLOCK-CONTROL-INFO = *NK2
Only files which can also be stored on NK2 volumes (but not NK4 volumes) are to be exported.

BLOCK-CONTROL-INFO = *NK4
Only files which can also be stored on NK4 volumes are to be exported.

MIGRATE = <u>*ANY</u> / list-poss(2): *ALLOWED / *INHIBITED / *FORBIDDEN
The user can specify which files are to be processed by the EXPORT-FILE command by the migration entry in the catalog (see the CREATE-FILE command, MIGRATE operand).

MIGRATE = <u>*ANY</u>
The specified files are to be exported, irrespective of the value in the MIGRATE operand in each of their catalog entries.

MIGRATE = *ALLOWED
Only files for which the catalog entry specifies the appropriate operand value, i.e. files which may be migrated to storage levels S1 and S2, are to be exported.

MIGRATE = *INHIBITED
Only files for which the catalog entry specifies the appropriate operand value, i.e. files which may not be migrated, are to be processed.

MIGRATE = *FORBIDDEN
Only files for which the catalog entry specifies the appropriate operand value are to be exported, i.e. files for which an intensified migration lock is declared. The files may not even be migrated for a brief period (e.g. for reorganization purposes).

STATUS = <u>*ANY</u> / *PARAMETERS(...)
The current file status is used as a selection criterion.

STATUS = <u>*ANY</u>
The file status is not to be used as a selection criterion.

STATUS = *PARAMETERS(...)
Only the files which have the specified status are selected for export. The following selection criteria are possible:

i The selection criteria within the *PARAMETERS(...) structure are logically ORed.

CLOSED-OUTPUT = *ANY / *YES / *NO
Specifies whether the “file closed” status is to be used as a selection criterion.

CLOSED-OUTPUT = *YES
Only the files which have already been closed are exported.

CLOSED-OUTPUT = *NO
Only the output files which have been opened in a program (OPEN OUTIN, INOUT or OUTPUT) and files which were not closed in an earlier system run or because a job was aborted are selected for export.

REPAIR-NEEDED = *ANY / *YES
Specifies whether files which were not closed in an earlier system run and not reconstructed with REPAIR-DISK-FILES are to be selected.

REPAIR-NEEDED = *ANY
Exports files regardless of whether or not they need to be repaired.

REPAIR-NEEDED = *YES
Only the files which were not closed in an earlier system run and which have not yet been reconstructed are selected for export.

BASIC-ACL = *ANY / *NONE / *YES / *PARAMETERS(...)
The BASIC-ACL entry in the file catalog is used as a selection criterion.

BASIC-ACL = *ANY
The BASIC-ACL entry is not to be used as a selection criterion.

BASIC-ACL = *NONE
Only the files that have no BASIC-ACL entry in the catalog are to be exported.

BASIC-ACL = *YES
Only the files which have a BASIC-ACL entry in the catalog are to be exported.

BASIC-ACL = *PARAMETERS(...)
Only the files which have the specified BASIC-ACL entry in the catalog are to be exported. NO-ACCESS means that no access rights were granted.

i Access rights specified with the OWNER, GROUP and OTHERS operands within the *PARAMETERS(...) structure are logically ORed.

OWNER = *ANY / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights must already be defined for the owner.

OWNER = *PARAMETERS(...)
Access rights that must be present for the owner (logically ORed together):

READ = <u>*ANY</u> / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = <u>*ANY</u> / *NO / *YES
Specifies whether write access authorization must be present.

EXEC = <u>*ANY</u> / *NO / *YES
Specifies whether execute access authorization must be present.

GROUP = <u>*ANY</u> / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights should already be defined for the owner's user group.

GROUP = *PARAMETERS(...)
Access rights that must be present for the owner's user group (logically ORed together):

READ = <u>*ANY</u> / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = <u>*ANY</u> / *NO / *YES
Specifies whether write access authorization must be present.

EXEC = <u>*ANY</u> / *NO / *YES
Specifies whether execute access authorization must be present.

OTHERS = <u>*ANY</u> / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights should already be defined for all other users.

OTHERS = *PARAMETERS(...)
Access rights that must be present for all other users (logically ORed together):

READ = <u>*ANY</u> / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = <u>*ANY</u> / *NO / *YES
Specifies whether write access authorization must be present.

EXEC = <u>*ANY</u> / *NO / *YES
Specifies whether execute access authorization must be present.

PROTECTION-ACTIVE = <u>*ANY</u> / list-poss(3): *LEVEL-0 / *LEVEL-1
The highest activated access control method (protection level) is used as a selection criterion. Only the files which have access protection at specified protection level are exported. Protection level 2 cannot be explicitly specified for EXPORT-FILE.

When the file is accessed, the highest activated protection level applies. The following table shows the method used for access control, the protection attributes, and the job variable protection hierarchy (protection levels):

Access control	**Protection attribute**	**Prot. level**
Standard access control	ACCESS and USER-ACCESS	0
Basic access control list	BASIC-ACL	1
Access control via guards	PASSWORD	2

Table 52: Hierarchy of access control methods

All other protection attributes of the file (e.g. passwords) are evaluated independently, without regard to the implemented protection level.

PROTECTION-ACTIVE = *ANY
The access control method is not to be used as a selection criterion.

PROTECTION-ACTIVE = *LEVEL-0
Only the files for which access is controlled via standard access control are to be exported.

PROTECTION-ACTIVE = *LEVEL-1
Only the files for which access is controlled via a basic access control list (BASIC-ACL protection) are to be exported.

ACCESS-COUNTER = *ANY / <integer 0..2147483647> / *INTERVAL(...)
The access counter for the file is used as a selection criterion (see the *ACC-COUNT* output field of the SHOW-FILE-ATTRIBUTES command).

ACCESS-COUNTER = *ANY
The access counter is not to be used as a selection criterion.

ACCESS-COUNTER = <integer 0..2147483647>
Only the files for which the access counter exactly matches the specified value are selected for export.

ACCESS-COUNTER = *INTERVAL(...)
Only the files for which the access counter lies in the specified value range which follows are to be exported.

FROM = 0 / <integer 0..2147483647>
Only those files for which the access counter ≥ the specified value are exported.

TO = 2147483647 / <integer 0..2147483647>
Only those files for which the access counter ≤ the specified value are exported.

HIGHEST-USED-PAGE = *ANY / <integer 0..2147483647> / *INTERVAL(...)
The number of PAM pages used serves as a selection criterion here, i.e. the last page pointer points to the specified page (see also the *HIGH-US-PA* output field of the SHOW-FILE-ATTRIBUTES command).

HIGHEST-USED-PAGE = <u>*ANY</u>
The number of PAM pages used is not used as a selection criterion.

HIGHEST-USED-PAGE = <integer 0..2147483647>
Only the files for which the specified number of PAM pages are used (i.e. files for which the last page pointer points to the specified page) are exported.

HIGHEST-USED-PAGE = *INTERVAL(...)
Exports only those files which use a number of pages that falls within the specified range.

FROM = 0 / <integer 0..2147483647>
Only the files in which the number of used pages ≥ the specified number are exported.

TO = 2147483647 / <integer 0..2147483647>
Only the files in which the number of used pages ≤ the specified number are exported.

TYPE-OF-FILES = <u>*ANY</u> / list-poss(2): *FILE / *PLAM-LIBRARY
Specifies whether files are to be selected on the basis of information specific to the file type. The files to be exported can restricted to normal files or to PLAM libraries.

TYPE-OF-FILES = <u>*ANY</u>
Exports files without taking the file type into account.

TYPE-OF-FILES = *FILE
Only files are exported.

TYPE-OF-FILES = *PLAM-LIBRARY
Only PLAM libraries are exported.

DIALOG-CONTROL = <u>*STD</u> / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE / *CATALOG-CHANGE/ *USER-ID-CHANGE
Specifies whether and under what conditions a verification dialog is to be conducted with the user during the export process. A control dialog is only possible in dialog mode but, in this mode, may also be used in procedures. The only operand value that you can specify in batch mode is *STD or *NO.

The user can intervene with the following inputs:
- Y: the specified file or file set will then be exported.
- N: the specified file or file set will not be exported.
- T: processing of the command will be terminated.
- ?: the possible responses will be listed, with an explanation of each.

In addition, the following options can be specified, separated by commas:

- ,CHECK = NO
  The DIALOG-CONTROL mode will be changed to '*NO'.
- ,CHECK = PVS
  The DIALOG-CONTROL mode will be changed to '*CATALOG-CHANGE'.
- ,CHECK = MULTIPLE
  The DIALOG-CONTROL mode will be changed to '*MORE-THAN-ONE-FILE'.
- ,CHECK = SINGLE
  SINGLE The DIALOG-CONTROL mode will be changed to '*FILE-CHANGE'.
- ,CHECK = ERROR
  The DIALOG-CONTROL mode will be changed to '*ERROR'.
- ,IGNORE = list-poss(3): RDPASS / WRPASS / EXPASS *Only privileged users may make this specification.* Specifies the type of password protection that is to be ignored on export. The specification only applies to a single file in the control dialog.
- ,PASSWORD = list-poss(3): <c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647>
  Enables password-protected files to be exported (maximum of 3 passwords). The specification only applies to a single file in the control dialog.

DIALOG-CONTROL = *STD
The default *STD setting is equivalent to *MORE-THAN-ONE-FILE in an interactive dialog (when SYSCMD is connected to the terminal) and to *NO in procedures and in batch mode.

DIALOG-CONTROL = *NO
The user cannot intervene in EXPORT-FILE processing; all the specified files will be exported (without a verification dialog).

DIALOG-CONTROL = *ERROR
If exporting of the selected files proceeds without error, they will be exported immediately, as when *NO is specified (i.e. no verification dialog). However, if a user-correctable error occurs during export, then a verification dialog takes place as for DIALOG-CONTROL = *FILE-CHANGE. DIALOG-CONTROL = *ERROR applies implicitly if DIALOG-CONTROL = *FILE-CHANGE is set. In the event of an error, the user may acknowledge the error message, abort EXPORT-FILE processing or attempt to rectify the error. If he wishes, he can also change the DIALOG-CONTROL mode (see also the possible forms of intervention listed under the first DIALOG-CONTROL operand).

DIALOG-CONTROL = *FILE-CHANGE
For each file which is to be exported, the user has the intervention options described for the first of the DIALOG-CONTROL operands. For each file which is to be processed, the user can decide interactively whether it should be exported or not (response: YES/NO). If in the verification dialog he specifies protection attributes under "IGNORE", or one or more passwords under "PASSWORD", these will be taken into account for any selected file and, if satisfied, the file will be exported without further queries ("YES" must also be specified). The user can also abort EXPORT-FILE processing, or change the DIALOG-CONTROL mode.

The files which have been exported will be listed in alphanumeric order.

DIALOG-CONTROL = *MORE-THAN-ONE-FILE
If exactly one file is specified, this will be exported immediately. If the file is specified in partially qualified form, which means that more than one file is selected, or if the file name contains wildcards, the user can decide, each time the catalog ID changes, whether or not files from the new catalog are to be exported (see the intervention options described for the first of the DIALOG-CONTROL operands). He must respond to the question issued by the system with “YES” or “NO”. DIALOG-CONTROL=*MORE-THAN-ONE-FILE is useful if wildcards are specified for “catid” in the FILE-NAME. In the verification dialog, EXPORT-FILE processing can be terminated, or the DIALOG-CONTROL mode can be changed.

DIALOG-CONTROL = *CATALOG-CHANGE
As with DIALOG-CONTROL = *MORE-THAN-ONE-FILE, the EXPORT-FILE processing routine branches to a verification dialog if files in different catalogs (pubsets) are affected. The user can determine whether the files in the current pubset should be exported (YES/NO), EXPORT-FILE processing should be terminated, or the DIALOG-CONTROL mode should be changed.

DIALOG-CONTROL = *USER-ID-CHANGE
Whenever the user ID is changed when deleting the catalog entries, a branch is made to guided dialog.

OUTPUT = <u>*STD</u> / *NO / *SYSOUT
The user can specify whether a message (`DMS0800`) with the name of the exported file is to be output to SYSOUT for each successfully exported file. The default setting *STD is equivalent to OUTPUT=*NO.

OUTPUT = *NO
No messages are output to SYSOUT for successfully exported files.

OUTPUT = *SYSOUT
For each file that is successfully exported, a message with the name of the file is output to SYSOUT.

IGNORE-PROTECTION = <u>*NONE</u> / list-poss(3): *WRITE-PASSWORD / *READ-PASSWORD / *EXEC-PASSWORD
*Only privileged users may enter operand values other than *NONE.* Systems support may specify that password protection is to be ignored. The specification IGNORE-PROTECTION in the EXPORT-FILE command thus makes it unnecessary to issue MODIFY-FILE-ATTRIBUTES commands to reset the protection attributes before the files can be exported.

IGNORE-PROTECTION = *WRITE-PASSWORD
System support staff are authorized to ignore the protection attribute **write password** when deleting the catalog entry.

IGNORE-PROTECTION = *READ-PASSWORD
System support staff are authorized to ignore the protection attribute **read password** when deleting the catalog entry.

IGNORE-PROTECTION = *EXEC-PASSWORD
System support staff are authorized to ignore the protection attribute **execute password** when deleting the catalog entry.

PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> / <integer -2147483648..2147483647>
The user can specify one or more passwords, which will permit files protected by these passwords to be exported. The passwords entered here are not recorded in the password table for the job, and are valid only for the current EXPORT-FILE processing. Up to 3 passwords may be specified in the form of a list.
In order to export a password-protected file, the password at the highest access level must be specified (see the ADD-PASSWORD command).
The operand has the following special characteristics:

– The password entered is not logged.
– The input field is automatically blanked out in the guided dialog.
– In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

PASSWORDS-TO-IGNORE = *NONE
No passwords are specified.

SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
In procedures the user can specify whether the spin-off mechanism or SDF-P error handling is to be triggered every time an error occurs (apart from syntax errors), or whether specific error conditions are to be ignored.

SUPPRESS-ERRORS = *NONE
All errors will trigger the spin-off mechanism or SDF-P error handling.

SUPPRESS-ERRORS = list-poss(3): <alphanum-name 7..7>
The user can define which errors are to be ignored by means of their DMS error codes (alphanum-name 7..7). If the specified error occurs, the spin-off mechanism will not be triggered. A maximum of 3 error codes may be specified.
DMS error code: 7 characters, of which the first three are always “DMS”; the last 4 characters identify the error; the digits 0...9 and letters A...F are permitted. When error codes are entered, no check is made to verify that valid error codes have been specified.

A detailed list of valid DMS error codes can be found on the manual server (URL: *http://manuals.ts.fujitsu.com*) by means of an HTML application and on the “BS2000 SoftBooks” DVD.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
1	0	CMD0001	No action required
2	0	DMS05F7	File generation does not exist, but group entry is updated
2	0	DMS06D6	Error on deleting certain files
	1	CMD0202	Syntax or semantic error in command
	32	DMS0584	A state that does not allow the function to continue was reported during processing
	64	CMD0102	Interrupted by K2 key
	64	CMD0216	Privilege errors
	64	DMS0501	Requested catalog not available
	64	DMS0512	Requested catalog not found
	64	DMS051B	Requested user ID not in pubset
	64	DMS051C	User not authorized to access pubset
	64	DMS0535	Specified file not shareable
	64	DMS055C	The catalog entry could not be found on the assigned volume
	64	DMS057B	Invalid operand for migrated file
	64	DMS057C	Processing not possible due to HSMS error
	64	DMS057D	File has been migrated and cannot be recalled without delay
	64	DMS057E	File has been migrated, and HSMS is not available
	64	DMS0585	Error detected when processing catalog or multiprocessor system
	64	DMS0586	It is not possible to access or reserve a volume at present
	64	DMS0587	Use of the specified command has been restricted by the system administrator
	64	DMS05FC	Specified user ID not in HOME pubset
	64	DMS0609	Access to system file not permitted
	64	DMS06FF	BCAM connection severed
	130	DMS0524	System address space exhausted
	130	DMS0582	File is currently locked or being used and cannot be processed
	130	DMS0585	Error detected when processing catalog or multiprocessor system
	130	DMS0586	It is not possible to access or reserve a volume at present
	130	DMS0594	Not enough virtual memory available

Examples

For examples of using the EXPORT-FILE command see the IMPORT-FILE command.

Notes

- If a file that is stored on a private disk or a Net-Storage volume is to be exported, then the device on which this disk is mounted will be requested for the job. After deletion of the file, the device will be returned to the system.

– For files stored on private disks, if OPTION=*DESTROY-ALL or MOUNT=*ALL-DISK is specified then all the volumes which hold the file must be already mounted at the time the command is executed. Otherwise, only the first volume which holds the file has to be mounted. The following volumes will not be requested until the command is being executed.

  This rule also applies if a partially qualified file name is specified in the DELETE-FILE command, thus addressing a number of files. In this situation, it is not necessary that all the volumes for all of the files are mounted simultaneously. The system will determine which of the files requires the most devices, and will request the corresponding number of devices.

EXPORT-NODE-FILE

Delete catalog entry of node files (export node files)

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION TSOS
Routing code:	$ (bei NBCONOPI=N) bzw. E (bei NBCONOPI=Y)

Function

The EXPORT-NODE-FILE command deletes the catalog entries for node files (files of the type NODE-FILE) which are located on a Net-Storage volume. When a node file is exported, the catalog entry is removed from the file catalog (TSOSCAT) and from the catalog of the Net-Storage volume. The node files and their data are retained on the Net-Storage.

The IMPORT-NODE-FILE command enables the catalog entry to be recreated (the node files to be imported).

Privileged functions

System support staff may export files belonging to any user ID. It may ignore the passwords of protected files and also branch to guided dialog whenever the user ID is changed.

By default, systems support (TSOS privilege) is a co-owner of all the files (and can therefore export files under any user ID). When SECOS is used, this co-ownership can be restricted.

Systems support can generate files under any user ID (TSOS privilege).In conjunction with the SECOS software product a user can allow other user IDs to act as co-owners. Co-owners of a user ID are then also allowed to export a file.

Overview of functions

	Function / Meaning	Level 1 operands	Level 2/3 operands
3-324	Identifier of the Net-Storage volume whose node files are to be exported	VOLUME	
3-324	Name of the file to be exported	FILE-NAME	
3-324	Selection criteria for the files to be deleted	SELECT= *BY-ATTRIBUTES	
	Date specification		
3-324	– creation date		CREATION-DATE
3-326	– Expiration date (implicit retention period)		EXPIRATION-DATE
3-328	– date of last access		LAST-ACCESS-DATE
3-337	– Number of accesses to the file		ACCESS-COUNTER
3-330	– Reserved storage space		SIZE
3-331	– Size of reserved storage space not yet used		NUMBER-OF-FREE-PAGES
3-338	– Number of the last page used (last page pointer)		HIGHEST-USED-PAGE
	File security/file protection		
3-331	– Access		ACCESS
3-332	– Shareability		USER-ACCESS
3-335	– Basic access control list		BASIC-ACL
3-336	– Highest activated access control		PROTECTION-ACTIVE
3-332	– Password protection		GUARDS
3-333	– BACKUP level		BACKUP-CLASS
3-333	– File format		BLOCK-CONTROL-INFO
3-334	– Status of the file – Closed – Not closed properly and not yet reconstructed		STATUS CLOSED-OUTPUT REPAIR-NEEDED
3-334	– Ability to migrate from processing level (S0) to background level (S1 or S2) in conjunction with HSMS		MIGRATE

Table 53: Overview of EXPORT-NODE-FILE command functions (Part 1 of 2)

	Function / Meaning	Level 1 operands	Level 2/3 operands
	Control parameters	DIALOG-CONTROL	
3-339	– User intervention not permitted (default setting for procedures and batch mode)	=*NO	
3-339	– Dialog when more than one file is referenced if the file name specified is not fully qualified (default setting for interactive mode)	=*MORE-THAN-ONE-FILE	
3-339	– Dialog when an error occurs	=*ERROR	
3-340	– Dialog when catalog ID changes	=*CATALOG-CHANGE	
3-339	– For each file selected, there will be an interactive dialog with the user to determine whether or not the current EXPORT-NODE-FILE command should be processed.	=*FILE-CHANGE	
3-340	– Dialog when user ID changes	=*USER-ID-CHANGE	
3-340	Message for successfully exported files	OUTPUT	
	– Suppress	=*NO	
	– Output to SYSOUT	=*SYSOUT	
3-340	Ignore protection attributes	IGNORE-PROTECTION	
	– Ignore write protection provided by ACCESS=READ, BASIC-ACL or GUARDS	=*ACCESS	
	– Ignore retention periods	=*EXPIRATION-DATE	
	– Ignore password protection	=*WRITE-PASSWORD / *READ-PASSWORD / *EXEC-PASSWORD	
3-341	Ignore password protection provided by specified passwords	PASSWORDS-TO-IGNORE	
3-341	Suppress error situations	SUPPRESS-ERRORS	

Table 53: Overview of EXPORT-NODE-FILE command functions (Part 2 of 2)

Format

```
EXPORT-NODE-FILE

VOLUME = <vsn 6..6>
FILE-NAME = *ALL / <filename 1..54 with-wild(80)>
,SELECT = *ALL / *BY-ATTRIBUTES(...)
  *BY-ATTRIBUTES(...)
    ⏐ CREATION-DATE = *ANY / *NONE / *TODAY / *YESTERDAY / <integer -99999..991231> /
    ⏐         <date> / [*INTERVAL](...)
    ⏐   [*INTERVAL](...)
    ⏐     ⏐ FROM = *EARLIEST / *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
    ⏐     ⏐ ,TO = *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
    ⏐ ,EXPIRATION-DATE = *ANY / *NONE / *TOMORROW / *TODAY / *YESTERDAY /
    ⏐                    <integer -99999..991231> / <date> / [*INTERVAL](...)
    ⏐   [*INTERVAL](...)
    ⏐     ⏐ FROM = *EARLIEST / *TOMORROW / *TODAY / *YESTERDAY /
    ⏐     ⏐           <integer -99999..991231> / <date>
    ⏐     ⏐ ,TO = *LATEST / TODAY / *TOMORROW / *YESTERDAY /
    ⏐     ⏐       <integer -99999..991231> / <date>
    ⏐ ,LAST-ACCESS-DATE = *ANY / *NONE / *TODAY / *YESTERDAY / <integer -99999..991231> /
    ⏐                     <date> / [*INTERVAL](...)
    ⏐   [*INTERVAL](...)
    ⏐     ⏐ FROM = *EARLIEST / *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
    ⏐     ⏐ ,TO = *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
    ⏐ ,SIZE = *ANY / *FREESIZE / <integer 0..2147483647> / [*INTERVAL](...)
    ⏐   [*INTERVAL](...)
    ⏐     ⏐ FROM = 0 / <integer 0..2147483647>
    ⏐     ⏐ ,TO = 2147483647 / <integer 0..2147483647>
    ⏐ ,NUMBER-OF-FREE-PAGES = *ANY / *SIZE / <integer 0..2147483647> / *ALL-ALLOCATED /
    ⏐                         [*INTERVAL](...)
    ⏐   [*INTERVAL](...)
    ⏐     ⏐ FROM = 0 / <integer 0..2147483647>
    ⏐     ⏐ ,TO = 2147483647 / <integer 0..2147483647>
```

(Part 1 of 3)

```
,ACCESS = *ANY / *READ / *WRITE
,PASSWORD = *ANY / list-poss(4): *NONE / *READ-PASSWORD / *WRITE-PASSWORD /
                  *EXEC-PASSWORD
,USER-ACCESS = *ANY / list-poss(3): *OWNER-ONLY / *ALL-USERS / *SPECIAL
,BACKUP-CLASS = *ANY / list-poss(5): *A / *B / *C / *D / *E
,BLOCK-CONTROL-INFO = *ANY / list-poss(9): *NONE / *NO / *WITHIN-DATA-BLOCK /
                                *WITHIN-DATA-2K-BLOCK / *WITHIN-DATA-4K-BLOCK / *PAMKEY /
                                 *NK / *NK2 / *NK4
,MIGRATE = *ANY / list-poss(3): *ALLOWED / *INHIBITED / *FORBIDDEN
,STATUS = *ANY / [*PARAMETERS](...)
   [*PARAMETERS](...)
        CLOSED-OUTPUT = *ANY / *YES / *NO
       ,REPAIR-NEEDED = *ANY / *YES
,BASIC-ACL = *ANY / *NONE / *YES / [*PARAMETERS](...)
   [*PARAMETERS](...)
        OWNER = *ANY / *NO-ACCESS / [*PARAMETERS](...)
          [*PARAMETERS](...)
               READ = *ANY / *NO / *YES
              ,WRITE = *ANY / *NO / *YES
              ,EXEC = *ANY / *NO / *YES
       ,GROUP = *ANY / *NO-ACCESS / [*PARAMETERS](...)
          [*PARAMETERS](...)
               READ = *ANY / *NO / *YES
              ,WRITE = *ANY / *NO / *YES
              ,EXEC = *ANY / *NO / *YES
       ,OTHERS = *ANY / *NO-ACCESS / [*PARAMETERS](...)
          [*PARAMETERS](...)
               READ = *ANY / *NO / *YES
              ,WRITE = *ANY / *NO / *YES
              ,EXEC = *ANY / *NO / *YES
,PROTECTION-ACTIVE = *ANY / list-poss(2): *LEVEL-0 / *LEVEL-1
,ACCESS-COUNTER = *ANY / <integer 0..2147483647> / [*INTERVAL](...)
   [*INTERVAL](...)
        FROM = 0 / <integer 0..2147483647>
       ,TO = 2147483647 / <integer 0..2147483647>
```

(Part 2 of 3)

```
      ,HIGHEST-USED-PAGE = *ANY / <integer 0..2147483647> / [*INTERVAL](...)
         [*INTERVAL](...)
              FROM = 0 / <integer 0..2147483647>
             ,TO = 2147483647 / <integer 0..2147483647>
,DIALOG-CONTROL = *STD / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE /
                          *CATALOG-CHANGE / *USER-ID-CHANGE
,OUTPUT = *STD / *NO / *SYSOUT
,IGNORE-PROTECTION = *NONE / list-poss(5): *ACCESS / *EXPIRATION-DATE / *WRITE-PASSWORD /
                           *READ-PASSWORD / *EXEC-PASSWORD
,PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> /
                             <integer -2147483648..2147483647>
,SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
```

(Part 3 of 3)

Operands

VOLUME = <vsn 6..6>
The VSN of the Net-Storage volume on which the node files to be exported are located.

From the set of node files selected by the FILE-NAME operand, only those which are located on the specified Net-Storage volume will be exported.

FILE-NAME = *ALL / <filename 1..54 with-wild(80)>
Defines the node files to be exported. The specification *ALL selects all node files which are located on the specified Net-Storage volume.

SELECT = *ALL
All the files in the set specified by the FILE-NAME operand are selected.

SELECT = *BY-ATTRIBUTES(...)
Restricts the node files in the set specified in the FILE-NAME operand to those which satisfy the criteria which follow. The default values, *ANY and ANY, each indicate that the file set is not to be restricted to any particular values for the attribute concerned.

CREATION-DATE = *ANY / *NONE / *TODAY / *YESTERDAY / <integer -99999..991231> / <date> / *INTERVAL(...)
The user can select the files to be exported by their creation date. See also the *CRE-DATE* and *CRE-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

CREATION-DATE = *ANY
The creation date is not to be used as a selection criterion.

CREATION-DATE = *NONE
Only those files which have the value NONE entered in the CREATION-DATE field in their catalog entry are exported, i.e. files which have never been opened.

CREATION-DATE = *TODAY
Only those files which have today's date entered in the CREATION-DATE field in their catalog entry are exported.

CREATION-DATE = *YESTERDAY
Only those files which have yesterday's date entered in the CREATION-DATE field in their catalog entry are exported.

CREATION-DATE = <integer -99999..991231>
Only those files which have the specified date entered in the CREATION-DATE field in their catalog entry are exported. Here, the user can specify the creation date in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value*: (6 digits, with preceding sign) the number of days from today's date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ ±0)

CREATION-DATE = <date>
Only those files which have the specified date entered in the CREATION-DATE field in their catalog entry are exported. The user can specify the creation date in the form [yy]yy-mm-dd.

CREATION-DATE = *INTERVAL(...)
Only those files which were created within the specified time period are exported. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified using <integer> values). It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). Whichever of the operands is not specified will be replaced by the default value for use as the limit of the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = *EARLIEST / *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
Only those files which have been created since the specified date (i.e. CREATION-DATE ≥ specified date) are exported. If EARLIEST is specified, the system will use the earliest possible date.

FROM = *TODAY
Exports only those files which were created on the current date.

FROM = *YESTERDAY
Exports only those files which were created as of yesterday’s date (i.e. files for which CREATION-DATE ≥ date of preceding day).

FROM = <integer -99999..991231>
Exports only those files which were created after the specified date (i.e. files for which CREATION-DATE ≥ specified date).

FROM = <date>
Exports only those files which were created after the specified date (i.e. files for which CREATION-DATE ≥ specified date).

TO = *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
Exports only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ specified date).

TO = *TODAY
Exports only those files which were created up to today’s date (i.e. files for which CREATION-DATE ≤ current date).

TO = *YESTERDAY
Exports only those files which were created up to yesterday’s date (i.e. files for which CREATION-DATE ≤ date of preceding day).

TO = <integer -99999..991231>
Exports only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ specified date).

TO = <date>
Exports only those files which were created before the specified date (i.e. files for which CREATION-DATE ≤ specified date).

EXPIRATION-DATE = *ANY / *NONE / *TOMORROW / *TODAY / *YESTERDAY / <integer -99999..991231> / <date> / *INTERVAL(...)
The user can select the files to be exported by their release date (expiration date). See also the *EXPIR-DATE* and *EXPIR-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

EXPIRATION-DATE = *ANY
The expiration date is not to be used as a selection criterion.

EXPIRATION-DATE = *NONE
Only those files which have no expiration date (i.e. the value NONE) entered in the EXPIRATION-DATE field in their catalog entry are exported.

EXPIRATION-DATE = *TOMORROW
Only those files which have tomorrow’s date entered as the EXPIRATION-DATE in their catalog entry are exported.

EXPIRATION-DATE = *TODAY
Only those files which have today's date entered as the EXPIRATION-DATE in their catalog entry are exported.

EXPIRATION-DATE = *YESTERDAY
Only those files which have yesterday's date entered as the EXPIRATION-DATE in their catalog entry are exported.

EXPIRATION-DATE = <integer -99999..991231>
Only those files which have the specified date entered as the EXPIRATION-DATE in their catalog entry are exported. Here, the user can specify the expiration date in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value:* (6 digits with preceding sign), the number of days from today's date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ ±0 or TOMORROW ≙ +1)

EXPIRATION-DATE = <date>
Only those files which have the specified date entered as the EXPIRATION-DATE in their catalog entry are exported. The user can specify the expiration date in the form [yy]yy-mm-dd.

EXPIRATION-DATE = *INTERVAL(...)
Only those files for which the expiration date lies within the specified time period, i.e. files for which the retention period ends within the specified time period, will be exported. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified for EXPIRATION-DATE = <integer...>. It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). Whichever of the operands is not specified will be replaced by the default value for use as the limit of the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = *EARLIEST / *TOMORROW / *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
If EARLIEST is specified, the system will use the earliest possible date. Only those files for which the retention period expires on or after the specified date (EXPIRATION-DATE ≥ specified date) are exported.

FROM = *TOMORROW
Exports only those files for which the EXPIRATION-DATE Ï the next day.

FROM = *TODAY
Exports only those files for which the EXPIRATION-DATE Ï the current date.

FROM = *YESTERDAY
Exports only those files for which the EXPIRATION-DATE Ï the specified date.

FROM = <integer -99999..991231>
Exports only those files for which the EXPIRATION-DATE Ï the specified date.

FROM = <date>
Exports only those files for which the EXPIRATION-DATE Ï the specified date.

TO = *LATEST / *TOMORROW / *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
Only files for which the retention period expires on or before the specified date (EXPIRATION-DATE ≤ specified date) will be exported.

TO = *TOMORROW
Exports only those files for which the EXPIRATION-DATE Î the next day.

TO = *TODAY
Exports only those files for which the EXPIRATION-DATE Î the current date.

TO = *YESTERDAY
Exports only those files for which the EXPIRATION-DATE Î the specified date.

TO = <integer -99999..991231>
Exports only those files for which the EXPIRATION-DATE Î the specified date.

TO = <date>
Exports only those files for which the EXPIRATION-DATE Î the specified date.

LAST-ACCESS-DATE = *ANY / *NONE / *TODAY / *YESTERDAY / <integer -99999..991231> / <date> / *INTERVAL(...)
The user can select the files to be exported by the date when they were last accessed. See also the *ACC-DATE* and *ACC-TIME* output fields in the SHOW-FILE-ATTRIBUTES command.

LAST-ACCESS-DATE = *ANY
The date of last access is not to be used as a selection criterion.

LAST-ACCESS-DATE = *NONE
Exports only those files which have the value NONE entered in the LADATE field in their catalog entry, i.e. files which have never been opened.

LAST-ACCESS-DATE = *TODAY
Exports only those files for which today’s date has been entered as the LAST-ACCESS-DATE in the catalog entry.

LAST-ACCESS-DATE = *YESTERDAY
Exports only those files for which yesterday’s date has been entered as the LAST-ACCESS-DATE in the catalog entry.

LAST-ACCESS-DATE = <integer -99999..991231>
Exports only those files for which the specified date has been entered as the LAST-ACCESS-DATE in the catalog entry. Here, the user can specify the last access date in either of two ways:

1. as an *absolute date value:* (6 digits), a specific date in the form yymmdd (yy = year, mm = month, dd = day)
2. as a *relative date value*: (6 digits, with preceding sign) the number of days from today's date, in the form -n for dates in the past, and +n for dates in the future; (YESTERDAY ≙ -1, TODAY ≙ ±0)

LAST-ACCESS-DATE = <date>
Exports only those files for which the specified date has been entered as the LAST-ACCESS-DATE in the catalog entry. The user can specify the creation date in the form [yy]yy-mm-dd.

LAST-ACCESS-DATE = *INTERVAL(...)
Only files which were last accessed within the specified time period will be exported. The range of dates covered by the specification includes the two end dates (see also the explanation of how dates are specified using <integer> values). It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit.

FROM = *EARLIEST / *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
Only files which have been accessed since the specified date (i.e. for which LAST-ACCESS-DATE ≥ specified date) will be exported. If EARLIEST is specified, the system will use the earliest possible date.

FROM = *TODAY
Exports only those files which have been accessed up to the date of the current date.

FROM = *YESTERDAY
Exports only those files which have been accessed up to the date of the preceding day. Returns information on files for which the LAST-ACCESS-DATE ≥ the specified date.

FROM = <integer -99999..991231>
Exports only those files for which the LAST-ACCESS-DATE ≥ the specified date.

FROM = <date>
Exports only those files for which the LAST-ACCESS-DATE ≥ the specified date.

TO = *TODAY / *YESTERDAY / <integer -99999..991231> / <date>
Only those files which were last accessed on or before the specified date (i.e. files for which LAST-ACCESS-DATE ≤ the specified date) are exported.

TO = *TODAY
Exports only those files which have been accessed up to the date of the current date.

TO = *YESTERDAY
Exports only those files for which the LAST-ACCESS-DATE Î the date of preceding day.

TO = <integer -99999..991231>
Exports only those files for which the LAST-ACCESS-DATE Î the specified date.

TO = <date>
Exports only those files for which the LAST-ACCESS-DATE Î the specified date.

SIZE = *ANY / *FREESIZE / <integer 0..2147483647> / *INTERVAL(...)
The user can select which files are to be exported by the EXPORT-NODE-FILE command on the basis of their size or the size of their reserved storage area (= number of PAM pages).
The value of the SIZE operand specifies the number of PAM pages: the upper and lower limits are both included in the range specified.

SIZE = *ANY
The file size is not to be used as a selection criterion.

SIZE = *FREESIZE
Only files for which storage space has been reserved but which do not yet occupy any of it (*HIGH-US-PA=0*) are to be exported.

SIZE = <integer 0..2147483647>
Only files for which the number of PAM pages reserved is exactly equal to the number specified here are to be exported.

SIZE = *INTERVAL(...)
Only files for which the number of pages lies within the specified range are to be exported. The upper and lower limits are both included in the range specified (see description of SIZE=<integer...>). It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit. Only files for which the number of PAM pages reserved is at least equal to the FROM value and at most equal to the TO value will be processed (FROM ≤ SIZE ≤ TO).

FROM = 0 / <integer 0..2147483647>
Only files for which the number of PAM pages reserved is at least equal to the specified number are to be exported (SIZE ≥ specified value).

TO = 2147483647 / <integer 0..2147483647>
Only files for which the number of PAM pages reserved is at most equal to the specified number are to be exported (SIZE ≤ specified value).

NUMBER-OF-FREE-PAGES = *ANY / *SIZE / <integer 0..2147483647> / *INTERVAL(...)
The user can select the files to be processed according to their number of free PAM pages, i.e. the number of pages reserved but unused.

NUMBER-OF-FREE-PAGES = *ANY
The number of free PAM pages is not to be used as a selection criterion.

NUMBER-OF-FREE-PAGES = *SIZE
Only files which actually occupy no storage space are to be exported (i.e. no PAM page has been written yet).

NUMBER-OF-FREE-PAGES = <integer 0..2147483647>
Only files which have exactly the number of reserved but unused (=free) PAM pages specified by <integer...> are to be exported.

NUMBER-OF-FREE-PAGES = *INTERVAL(...)
Selects all files for which the number of unused PAM pages lies within the specified range. The upper and lower limits are both included in the range specified (see description of NUMBER-OF-FREE-PAGES = <integer...>). It is also possible to specify limits using only the operand FROM (lower limit) or TO (upper limit). For whichever operand is not specified, the default value will be used as the limit for the range. The use of range limits for export can only be meaningful if the chosen lower limit ≤ the upper limit.
Only disk files which have at least as many free PAM pages as specified by the FROM value and at most as many as given by the TO value will be exported (FROM ≤ FREE Î TO).

FROM = 0 / <integer 0..2147483647>
Only files for which the number free PAM pages is at least equal to the specified number are to be exported (FREE Ï integer).

TO = 2147483647 / <integer 0..2147483647>
Only files for which the number free PAM pages is at most equal to the specified number are to be exported (FREE Î integer).

ACCESS = *ANY / *READ / *WRITE
The specified access type will be used as a selection criterion (*ACCESS* output field).

ACCESS = *ANY
The ACCESS value is not to be used as a selection criterion.

ACCESS = *READ
Only files for which write access is prevented by ACCESS=READ, i.e. for which only read access is permitted, will be exported.

ACCESS = *WRITE
Only files for which write access is permitted will be exported.

PASSWORD = *<u>ANY</u> / list-poss(4): *NONE / *READ-PASSWORD / *WRITE-PASSWORD / *EXEC-PASSWORD
The user can select files with specific password types for processing by the EXPORT-NODE-FILE command.

PASSWORD = <u>*ANY</u>
Password protection is not to be used as a selection criterion.

PASSWORD = *NONE
Only files for which there is no password protection are to be exported.

PASSWORD = *READ-PASSWORD
Only files which are protected by a read password are to be exported.

PASSWORD = *WRITE-PASSWORD
Only files which are protected by a write password are to be exported.

PASSWORD = *EXEC-PASSWORD
Only files which are protected by an execute password are to be exported.

USER-ACCESS = *<u>ANY</u> / listposs(3): *OWNER-ONLY / *ALL-USERS / *SPECIAL
The user can select files using their access authorizations as a criterion.

USER-ACCESS = <u>*ANY</u>
The access authorization is not to be used as a selection criterion.

USER-ACCESS = *OWNER-ONLY
Only files which the owner alone may access are to be exported.

USER-ACCESS = *ALL-USERS
Only files which can be accessed by all users are to be exported.

USER-ACCESS = *SPECIAL
Only files which can be accessed by all user IDs including the maintenance IDs (i.e. user IDs with HARDWARE-MAINTENANCE privilege) are to be exported.

BACKUP-CLASS = *ANY / list-poss(5): *A / *B / *C / *D / *E
The user can select files for export by their BACKUP-CLASS level.

BACKUP-CLASS = *ANY
The BACKUP-CLASS level is not to be used as a selection criterion.

BACKUP-CLASS = *A
Only the files for which the value A is entered in the catalog as the BACKUP-CLASS (most frequent backup) are exported.

BACKUP-CLASS = *B
Only the files for which the value B is entered in the catalog as the BACKUP-CLASS are exported.

BACKUP-CLASS = *C
Only the files for which the value C is entered in the catalog as the BACKUP-CLASS are exported.

BACKUP-CLASS = *D
Only the files for which the value D is entered in the catalog as the BACKUP-CLASS are exported.

BACKUP-CLASS = *E
Only the files for which the value E is entered in the catalog as the BACKUP-CLASS are exported.

BLOCK-CONTROL-INFO = ANY / list-poss(4): NONE / NO / WITHIN-DATA-BLOCK / WITHIN-DATA-2K-BLOCK / WITHIN-DATA-4K-BLOCK / PAMKEY / NK / NK2 / NK4
The user can select files for export by their file format (*BLK-CONTR* output field).

BLOCK-CONTROL-INFO = *ANY
The BLOCK-CONTROL entry is not to be used as a selection criterion.

BLOCK-CONTROL-INFO = *NONE
Only files for which no BLK-CNTRL value has been defined, i.e. files which have not yet been opened, are to be exported.

BLOCK-CONTROL-INFO = *NO
Only files which contain no block control field are to be exported.

BLOCK-CONTROL-INFO = *WITHIN-DATA-BLOCK
Only files which were created with BLOCK-CONTROL-INFO=WITHIN-DATA-BLOCK (i.e. files for which block control information is held in a block control field at the start and within the data block) are to be exported.

BLOCK-CONTROL-INFO = *WITHIN-DATA-2K-BLOCK
Only files which were created with BLOCK-CONTROL-INFO=WITHIN-DATA-2K-BLOCK (i.e. files for which block control information is located at the start of each 2K block) are to be exported.

BLOCK-CONTROL-INFO = *WITHIN-DATA-4K-BLOCK
Only files which were created with BLOCK-CONTROL-INFO=WITHIN-DATA-4K-BLOCK (i.e. files for which block control information is located at the start of each 4K block) are to be exported.

BLOCK-CONTROL-INFO = *PAMKEY
Only files which use a separate PAM key for the block control field (i.e the block control information is held in a separate key field, outside the PAM block) are to be exported.

BLOCK-CONTROL-INFO = *NK
Only the NK files are to be exported, i.e. files which can also be stored on NK volumes (NK2 and NK4).

BLOCK-CONTROL-INFO = *NK2
Only files which can also be stored on NK2 volumes (but not NK4 volumes) are to be exported.

BLOCK-CONTROL-INFO = *NK4
Only files which can also be stored on NK4 volumes are to be exported.

MIGRATE = <u>*ANY</u> / list-poss(2): *ALLOWED / *INHIBITED / *FORBIDDEN
The user can specify which files are to be processed by the EXPORT-NODE-FILE command by the migration entry in the catalog (see the CREATE-FILE command, MIGRATE operand).

MIGRATE = <u>*ANY</u>
The specified files are to be exported, irrespective of the value in the MIGRATE operand in each of their catalog entries.

MIGRATE = *ALLOWED
Only files for which the catalog entry specifies the appropriate operand value, i.e. files which may be migrated to storage levels S1 and S2, are to be exported.

MIGRATE = *INHIBITED
Only files for which the catalog entry specifies the appropriate operand value, i.e. files which may not be migrated, are to be processed.

MIGRATE = *FORBIDDEN
Only files for which the catalog entry specifies the appropriate operand value are to be exported, i.e. files for which an intensified migration lock is declared. The files may not even be migrated for a brief period (e.g. for reorganization purposes).

STATUS = <u>*ANY</u> / *PARAMETERS(...)
The current file status is used as a selection criterion.

STATUS = <u>*ANY</u>
The file status is not to be used as a selection criterion.

STATUS = *PARAMETERS(...)
Only the files which have the specified status are selected for export. The following selection criteria are possible:

i The selection criteria within the *PARAMETERS(...) structure are logically ORed.

CLOSED-OUTPUT = *ANY / *YES / *NO
Specifies whether the “file closed” status is to be used as a selection criterion.

CLOSED-OUTPUT = *YES
Only the files which have already been closed are exported.

CLOSED-OUTPUT = *NO
Only the output files which have been opened in a program (OPEN OUTIN, INOUT or OUTPUT) and files which were not closed in an earlier system run or because a job was aborted are selected for export.

REPAIR-NEEDED = *ANY / *YES
Specifies whether files which were not closed in an earlier system run and not reconstructed with REPAIR-DISK-FILES are to be selected.

REPAIR-NEEDED = *ANY
Exports files regardless of whether or not they need to be repaired.

REPAIR-NEEDED = *YES
Only the files which were not closed in an earlier system run and which have not yet been reconstructed are selected for export.

BASIC-ACL = *ANY / *NONE / *YES / *PARAMETERS(...)
The BASIC-ACL entry in the file catalog is used as a selection criterion.

BASIC-ACL = *ANY
The BASIC-ACL entry is not to be used as a selection criterion.

BASIC-ACL = *NONE
Only the files that have no BASIC-ACL entry in the catalog are to be exported.

BASIC-ACL = *YES
Only the files which have a BASIC-ACL entry in the catalog are to be exported.

BASIC-ACL = *PARAMETERS(...)
Only the files which have the specified BASIC-ACL entry in the catalog are to be exported. NO-ACCESS means that no access rights were granted.

i Access rights specified with the OWNER, GROUP and OTHERS operands within the *PARAMETERS(...) structure are logically ORed.

OWNER = *ANY / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights must already be defined for the owner.

OWNER = *PARAMETERS(...)
Access rights that must be present for the owner (logically ORed together):

READ = <u>*ANY</u> / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = <u>*ANY</u> / *NO / *YES
Specifies whether write access authorization must be present.

EXEC = <u>*ANY</u> / *NO / *YES
Specifies whether execute access authorization must be present.

GROUP = <u>*ANY</u> / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights should already be defined for the owner’s user group.

GROUP = *PARAMETERS(...)
Access rights that must be present for the owner’s user group (logically ORed together):

READ = <u>*ANY</u> / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = <u>*ANY</u> / *NO / *YES
Specifies whether write access authorization must be present.

EXEC = <u>*ANY</u> / *NO / *YES
Specifies whether execute access authorization must be present.

OTHERS = <u>*ANY</u> / *NO-ACCESS / *PARAMETERS(...)
Specifies which access rights should already be defined for all other users.

OTHERS = *PARAMETERS(...)
Access rights that must be present for all other users (logically ORed together):

READ = <u>*ANY</u> / *NO / *YES
Specifies whether read access authorization must be present.

WRITE = <u>*ANY</u> / *NO / *YES
Specifies whether write access authorization must be present.

EXEC = <u>*ANY</u> / *NO / *YES
Specifies whether execute access authorization must be present.

PROTECTION-ACTIVE = <u>*ANY</u> / list-poss(3): *LEVEL-0 / *LEVEL-1
The highest activated access control method (protection level) is used as a selection criterion. Only the files which have access protection at specified protection level are exported. Protection level 2 cannot be explicitly specified for EXPORT-NODE-FILE.

When the file is accessed, the highest activated protection level applies. The following table shows the method used for access control, the protection attributes, and the job variable protection hierarchy (protection levels):

Access control	**Protection attribute**	**Prot. level**
Standard access control	ACCESS and USER-ACCESS	0
Basic access control list	BASIC-ACL	1
Access control via guards	PASSWORD	2

Table 54: Hierarchy of access control methods

All other protection attributes of the file (e.g. passwords) are evaluated independently, without regard to the implemented protection level.

PROTECTION-ACTIVE = *ANY
The access control method is not to be used as a selection criterion.

PROTECTION-ACTIVE = *LEVEL-0
Only the files for which access is controlled via standard access control are to be exported.

PROTECTION-ACTIVE = *LEVEL-1
Only the files for which access is controlled via a basic access control list (BASIC-ACL protection) are to be exported.

ACCESS-COUNTER = *ANY / <integer 0..2147483647> / *INTERVAL(...)
The access counter for the file is used as a selection criterion (see the *ACC-COUNT* output field of the SHOW-FILE-ATTRIBUTES command).

ACCESS-COUNTER = *ANY
The access counter is not to be used as a selection criterion.

ACCESS-COUNTER = <integer 0..2147483647>
Only the files for which the access counter exactly matches the specified value are selected for export.

ACCESS-COUNTER = *INTERVAL(...)
Only the files for which the access counter lies in the specified value range which follows are to be exported.

FROM = 0 / <integer 0..2147483647>
Only those files for which the access counter ≥ the specified value are exported.

TO = 2147483647 / <integer 0..2147483647>
Only those files for which the access counter ≤ the specified value are exported.

HIGHEST-USED-PAGE = *ANY / <integer 0..2147483647> / *INTERVAL(...)
The number of PAM pages used serves as a selection criterion here, i.e. the last page pointer points to the specified page (see also the *HIGH-US-PA* output field of the SHOW-FILE-ATTRIBUTES command).

HIGHEST-USED-PAGE = *ANY
The number of PAM pages used is not used as a selection criterion.

HIGHEST-USED-PAGE = <integer 0..2147483647>
Only the files for which the specified number of PAM pages are used (i.e. files for which the last page pointer points to the specified page) are exported.

HIGHEST-USED-PAGE = *INTERVAL(...)
Exports only those files which use a number of pages that falls within the specified range.

FROM = 0 / <integer 0..2147483647>
Only the files in which the number of used pages ≥ the specified number are exported.

TO = 2147483647 / <integer 0..2147483647>
Only the files in which the number of used pages ≤ the specified number are exported.

DIALOG-CONTROL = *STD / *NO / *ERROR / *FILE-CHANGE / *MORE-THAN-ONE-FILE / *CATALOG-CHANGE/ *USER-ID-CHANGE
Specifies whether and under what conditions a verification dialog is to be conducted with the user during the export process. A control dialog is only possible in dialog mode but, in this mode, may also be used in procedures. The only operand value that you can specify in batch mode is *STD or *NO.

The user can intervene with the following inputs:

– Y: the specified file or file set will then be exported.
– N: the specified file or file set will not be exported.
– T: processing of the command will be terminated.
– ?: the possible responses will be listed, with an explanation of each.

In addition, the following options can be specified, separated by commas:

– ,CHECK = NO
  The DIALOG-CONTROL mode will be changed to ‘*NO’.
– ,CHECK = PVS
  The DIALOG-CONTROL mode will be changed to ‘*CATALOG-CHANGE’.
– ,CHECK = MULTIPLE
  The DIALOG-CONTROL mode will be changed to ‘*MORE-THAN-ONE-FILE’.
– ,CHECK = SINGLE
  SINGLE The DIALOG-CONTROL mode will be changed to ‘*FILE-CHANGE’.
– ,CHECK = ERROR
  The DIALOG-CONTROL mode will be changed to ‘*ERROR’.

- ,IGNORE = list-poss(3): RDPASS / WRPASS / EXPASS *Only privileged users may make this specification.* Specifies the type of password protection that is to be ignored on export. The specification only applies to a single file in the control dialog.
- ,PASSWORD = list-poss(3): <c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647>
  Enables password-protected files to be exported (maximum of 3 passwords). The specification only applies to a single file in the control dialog.

DIALOG-CONTROL = *STD

The default *STD setting is equivalent to *MORE-THAN-ONE-FILE in an interactive dialog (when SYSCMD is connected to the terminal) and to *NO in procedures and in batch mode.

DIALOG-CONTROL = *NO

The user cannot intervene in EXPORT-NODE-FILE processing; all the specified files will be exported (without a verification dialog).

DIALOG-CONTROL = *ERROR

If exporting of the selected files proceeds without error, they will be exported immediately, as when *NO is specified (i.e. no verification dialog). However, if a user-correctable error occurs during export, then a verification dialog takes place as for DIALOG-CONTROL = *FILE-CHANGE. DIALOG-CONTROL = *ERROR applies implicitly if DIALOG-CONTROL = *FILE-CHANGE is set. In the event of an error, the user may acknowledge the error message, abort EXPORT-NODE-FILE processing or attempt to rectify the error. If he wishes, he can also change the DIALOG-CONTROL mode (see also the possible forms of intervention listed under the first DIALOG-CONTROL operand).

DIALOG-CONTROL = *FILE-CHANGE

For each file which is to be exported, the user has the intervention options described for the first of the DIALOG-CONTROL operands. For each file which is to be processed, the user can decide interactively whether it should be exported or not (response: YES/NO). If in the verification dialog he specifies protection attributes under “IGNORE”, or one or more passwords under “PASSWORD”, these will be taken into account for any selected file and, if satisfied, the file will be exported without further queries (“YES” must also be specified). The user can also abort EXPORT-NODE-FILE processing, or change the DIALOG-CONTROL mode.

The files which have been exported will be listed in alphanumeric order.

DIALOG-CONTROL = *MORE-THAN-ONE-FILE

If exactly one file is specified, this will be exported immediately. If the file is specified in partially qualified form, which means that more than one file is selected, or if the file name contains wildcards, the user can decide, each time the catalog ID changes, whether or not files from the new catalog are to be exported (see the intervention options described for the first of the DIALOG-CONTROL operands). He must respond to the question issued by the system with “YES” or “NO”. DIALOG-CONTROL=*MORE-THAN-ONE-FILE is useful if wildcards are specified for “catid” in the FILE-NAME. In the verification dialog, EXPORT-NODE-FILE processing can be terminated, or the DIALOG-CONTROL mode can be changed.

DIALOG-CONTROL = *CATALOG-CHANGE
As with DIALOG-CONTROL = *MORE-THAN-ONE-FILE, the EXPORT-NODE-FILE processing routine branches to a verification dialog if files in different catalogs (pubsets) are affected. The user determines whether the files in the current pubset should be exported (YES/NO), EXPORT-NODE-FILE processing should be terminated, or the DIALOG-CONTROL mode should be changed.

DIALOG-CONTROL = *USER-ID-CHANGE
Whenever the user ID is changed when deleting the catalog entries, a branch is made to guided dialog.

OUTPUT = <u>*STD</u> / *NO / *SYSOUT
The user can specify whether a message (`DMS0800`) with the name of the exported file is to be output to SYSOUT for each successfully exported file. The default setting *STD is equivalent to OUTPUT=*NO.

OUTPUT = *NO
No messages are output to SYSOUT for successfully exported files.

OUTPUT = *SYSOUT
For each file that is successfully exported, a message with the name of the file is output to SYSOUT.

IGNORE-PROTECTION = <u>*NONE</u> / list-poss(5): *ACCESS / *EXPIRATION-DATE / *WRITE-PASSWORD / *READ-PASSWORD / *EXEC-PASSWORD
The user can specify whether any defined file protection against write access or any defined retention period is to be ignored. Systems support staff can also ignore password protection.
The specification IGNORE-PROTECTION in the EXPORT-NODE-FILE command thus makes it unnecessary to issue the MODIFY-FILE-ATTRIBUTES command to reset the protection attributes before the files can be deleted.

IGNORE-PROTECTION = <u>*NONE</u>
The protection attributes “read-only” (ACCESS=READ or no write access with BASIC-ACL or GUARDS protection) and “retention period” (EXPIRATION-DATE) will be observed during export.

IGNORE-PROTECTION = *ACCESS
Files for which write access by the owner is not allowed at the highest activated protection level may still be exported (see the selection criterion PROTECTION-ACTIVE, ACCESS, BASIC-ACL or GUARDS).

IGNORE-PROTECTION = *EXPIRATION-DATE
Files which are still within their retention period (*EXPIR-DATE* > current date) may also be exported.

IGNORE-PROTECTION = *WRITE-PASSWORD
System support staff are authorized to ignore the protection attribute **write password** when deleting the catalog entry.

IGNORE-PROTECTION = *READ-PASSWORD
System support staff are authorized to ignore the protection attribute **read password** when deleting the catalog entry.

IGNORE-PROTECTION = *EXEC-PASSWORD
System support staff are authorized to ignore the protection attribute **execute password** when deleting the catalog entry.

PASSWORDS-TO-IGNORE = *NONE / *SECRET / list-poss(3): <x-string 1..8> / <c-string 1..4> / <integer -2147483648..2147483647>
The user can specify one or more passwords, which will permit files protected by these passwords to be exported. The passwords entered here are not recorded in the password table for the job, and are valid only for the current EXPORT-NODE-FILE processing. Up to 3 passwords may be specified in the form of a list.
In order to export a password-protected file, the password at the highest access level must be specified (see the ADD-PASSWORD command).
The operand has the following special characteristics:

– The password entered is not logged.
– The input field is automatically blanked out in the guided dialog.
– In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

PASSWORDS-TO-IGNORE = *NONE
No passwords are specified.

SUPPRESS-ERRORS = *NONE / list-poss(3): <alphanum-name 7..7>
In procedures the user can specify whether the spin-off mechanism or SDF-P error handling is to be triggered every time an error occurs (apart from syntax errors), or whether specific error conditions are to be ignored.

SUPPRESS-ERRORS = *NONE
All errors will trigger the spin-off mechanism or SDF-P error handling.

SUPPRESS-ERRORS = list-poss(3): <alphanum-name 7..7>
The user can define which errors are to be ignored by means of their DMS error codes (alphanum-name 7..7). If the specified error occurs, the spin-off mechanism will not be triggered. A maximum of 3 error codes may be specified.
DMS error code: 7 characters, of which the first three are always “DMS”; the last 4 characters identify the error; the digits 0...9 and letters A...F are permitted. When error codes are entered, no check is made to verify that valid error codes have been specified.

A detailed list of valid DMS error codes can be found on the manual server (URL: *http://manuals.ts.fujitsu.com*) by means of an HTML application and on the “BS2000 SoftBooks” DVD.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed
1	0	CMD0001	No action required
2	0	DMS06D6	Error on exporting certain files Guaranteed messages: DMS0800, DMS0801, DMS06D5, DMS0666, DMS05C6, DMS05BF, DMS05C3, DMS053F
	1	CMD0202	Syntax or semantic error in command
	32	DMS0584	A state that does not allow the function to continue was reported during processing
	64	CMD0102	Interrupted by K2 key
	64	CMD0216	Privilege errors
	64	DMS0501	Requested catalog not available
	64	DMS0512	Requested catalog not found
	64	DMS051B	Requested user ID not in pubset Guaranteed message: DMS051B
	64	DMS051C	User not authorized to access pubset Guaranteed message: DMS051C
	64	DMS0535	Specified file not shareable
	64	DMS055C	The catalog entry could not be found on the assigned volume
	64	DMS0585	Error detected when processing catalog or multiprocessor system
	64	DMS0586	It is not possible to access or reserve a volume at present
	64	DMS0587	Use of the specified command has been restricted by the system administrator
	64	DMS05FC	Specified user ID not in home pubset
	64	DMS0609	No access to system file
	64	DMS06FF	BCAM connection severed
	130	DMS0524	System address space exhausted
	130	DMS0582	File is currently locked or being used and cannot be processed
	130	DMS0585	Error detected when processing catalog or multiprocessor system
	130	DMS0586	It is not possible to access or reserve a volume at present
	130	DMS0594	Not enough virtual memory available

EXPORT-PUBSET

Export previously imported pubset

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Pubset and MRSCAT management
Domain:	MULTI-CATALOG-AND-PUBSET-MGMT
Privileges:	TSOS OPERATING
Routing code:	R

Function

The EXPORT-PUBSET command enables systems support personnel to remove a pubset as a resource from the local system. In a shared pubset network based on HIPLEX MSCF, a choice can be made as to whether exporting the shared pubset on the master system also entails exporting it on the slave systems, or whether the master function should pass to one of the slave systems.
Under control of the calling task, the command generates its own (new) task. This task performs EXPORT processing **asynchronously** to the calling task.
Successful generation of the EXPORT task and its messages are logged at the operator terminal.

During the export phase, the number of tasks that are still using the pubset is output. The TSN of the tasks involved can be determined using the SHOW-PUBSET-OCCUPATION command (and also the SHOW-PUBSET-PARAMETERS command) and used to force the tasks to terminate their activities.
The user catalog (SYSSRPM or TSOSJOIN file) is closed. The information currently stored in main memory regarding user IDs and disk storage occupancy is transferred. If this information cannot be transferred, reconstruction of the F5 label is initiated the next time the IMPORT-PUBSET command is issued.

The home pubset may not be exported using this command. The home pubset is exported automatically at system termination.

Format

```
EXPORT-PUBSET

 PUBSET = <cat-id 1..4>
,USE = *STD / *BY-REMOTE
,SHARER-TYPE = *STD / *SLAVE / *MASTER(...)
   *MASTER(...)
     |  MASTER-CHANGE = *NO / *YES
,TERMINATE-JOBS = *NO / YES
,MONJV = *NONE / <filename 1..54 without-gen-vers>
,JV-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / <integer -2147483639..2147483639>
```

Operands

PUBSET = <cat-id 1..4>
Catalog ID of the pubset to be exported.

USE = *STD / *BY-REMOTE
Specifies the status that the pubset being exported has to have.

USE = *STD
The pubset cannot be exported unless it is in “imported” or “remote imported” status.
If the local system has imported the pubset as master in a shared pubset network, then exporting it with the setting “MASTER-CHANGE=*NO” causes all slave systems to export it as well.

USE =*BY-REMOTE
The pubset cannot be exported unless it is in “remote-imported” status (i.e., only catalog accesses are possible).

SHARER-TYPE = *STD / *SLAVE / *MASTER(...)
Specifies which type of sharer as defined on pubset import will be accepted on pubset export.

SHARER-TYPE = *STD
The pubset will be exported regardless of sharer type.

SHARER-TYPE = *SLAVE
The pubset will be exported only if the local system imported the pubset as a slave sharer.

SHARER-TYPE = *MASTER(...)
The pubset will be exported only if the local system is a temporary owner of the pubset (master).

MASTER-CHANGE = *NO
Exporting the pubset implies a request to all slave sharers to export the pubset as well.

MASTER-CHANGE = *YES
An automatic change of master is to be initiated. The system defined as the backup master is to take over the role of master. The slave sharers are not forced to export the pubset. If no backup master is defined, or if the system defined as backup master is not active, pubset export is rejected.

Use of MASTER-CHANGE=*YES is conditional on the presence of a version ≥ BS2000/OSD-BC V2.0 on all systems in the network.

TERMINATE-JOBS = *NO / *YES
Terminates the tasks that are using the pubset or waits until their allocations are cleared.

TERMINATE-JOBS = *NO
The pubset is made inaccessible. If the pubset is still allocated when this command is entered, the command does not take effect until all active allocations are cleared. New allocations are rejected.

TERMINATE-JOBS = *YES
Export processing cancels all tasks that are using the pubset. Note however that system tasks which are using the pubset cannot be terminated. That means that the system administration must ensure that all such allocations are cleared before EXPORT-PUBSET is called. Otherwise export processing cannot be completed.
Processing is continued once all allocations are cleared. This must be preceded by an EXPORT-PUBSET command with TERMINATE-JOBS=NO. The wait state of an export job previously initiated with the TERMINATE-JOBS=NO operand is also canceled once all allocations have been cleared.

MONJV = *NONE / <filename 1..54 without gen>
Specifies whether a monitoring job variable is set. This operand should only be specified if the JV software product is used. The job variable is set to the following values:

$E	during export processing
$T	after successful completion of export processing
$A	if export processing was terminated owing to an error or if EXPORT-PUBSET was terminated with TERMINATE-JOBS=*YES

The job variable must already be cataloged, otherwise it is not set. However, EXPORT processing is continued even if no job variables have been defined.

JV-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / <integer -2147483639..2147483639>

Password of the monitoring job variable.

The operand has the following special characteristics:

– The password entered is not logged.

– The input field is automatically blanked out in the guided dialog.

– In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
1	0	DMS0355	Same export already active
1	0	DMS0364	Pubset currently unavailable
	1	CMD0202	Syntax error
	32	DMS035C	Invalid operands
	32	DMS0363	MRSCAT access error
	64	DMS035C	IMPORT-PUBSET required
	64	DMS0360	No authorization for command
	64	DMS0366	Export attempt for home pubset
	64	DMS036D	Invalid operand sequence
	130	DMS0351	Other import/export task active
	130	DMS035C	Maximum number of tasks reached
	130	DMS0362	Class 4 memory error

Notes

During system termination, all imported pubsets are always exported, in the following order:

1. During termination of HIPLEX MSCF, all shared imported pubsets are exported.
   A pubset imported as master is imported with the MASTER-CHANGE option set to *YES.

2. Export all non-shared imported pubsets (except the home pubset):
   – EXPORT-PUBSET with TERMINATE-JOBS=*YES
   – wait until all pubsets have been exported (maximum approx. 1 minute)
   – FORCE-PUBSET-EXPORT for the pubsets that have not yet been exported
   – wait until all pubsets have been exported (maximum 1 minute)

3. Export the home pubset:
   – EXPORT-PUBSET with the TERMINATE-JOBS=*YES
   – wait until the pubset has been exported (maximum 1 minute)
   – FORCE-PUBSET-EXPORT if the home pubset has not yet been exported
   – wait until the home pubset has been exported (maximum 1 minute)

The generated task executes in 2 phases:

Phase 1: Wait until all files of the pubset are closed.

This state only occurs if, in addition to the open system files (e.g. the user catalog), other files of this pubset are open. This also implicitly involves the private files that are addressed via this pubset.
If only the system files are opened, the wait state does not occur. The system files are open for each imported pubset and are not closed until the pubset is actually exported. If the wait state occurs, the number of tasks that are still using the pubset is output. The tasks involved can be displayed by means of the SHOW-PUBSET-OCCUPATION command (and also the SHOW-PUBSET-PARAMETERS command) and be forced to terminate their activities. By default the wait state is canceled once all files (except the system files) are closed and Phase 2, the export phase proper, is initiated. In addition, the following events may cause the wait state to be canceled:

– CANCEL-PUBSET-EXPORT command
  The wait state is canceled immediately. The EXPORT task terminates with an error message and the pubset remains available.

– EXPORT-PUBSET command with the TERMINATE-JOBS=*YES operand
  An attempt is made to terminate all jobs that are still using the pubset. Once all allocations are cleared, the wait state is canceled and the second phase - the export phase - is initiated. The EXPORT task acknowledges the command.

– FORCE-PUBSET-EXPORT command
  The wait state is canceled. The export task acknowledges the command and Phase 2 - the export phase - is initiated, even though not all the files of this pubset are closed.

Phase 2: Export pubset

The pubset is made inaccessible. SPOOL is informed and all spoolout jobs whose files to be output are cataloged on the pubset involved are moved from TYPE5/AC to TYPE5/KP. The system files are closed and all resources are released.
Although wait states may occur during this phase, they are limited to approximately 10 minutes. The FORCE-PUBSET-EXPORT command reduces the wait state to 1 minute.

If an export job, with options for terminating occupying tasks or waiting for their allocations to be cleared, is issued at the master processor in shared pubset mode in a multiprocessor network, it is also effective at all the slave processors in the network.
FORCE-PUBSET-EXPORT commands, however, are only effective at the local processor, i.e. they are not passed on to all the slave processors in the network.

EXTEND-PAGING-AREA

Extend paging area

Description status:	BS2000 OSD/BC V10.0A
Functional area:	System control and optimization
Domain:	SYSTEM-MANAGEMENT SYSTEM-TUNING
Privileges:	TSOS OPERATING
Routing code:	R

Function

The EXTEND-PAGING AREA command enables systems support to extend the paging area to a maximum size of 4 Tbytes by adding one or more paging files (paging areas).

Format

EXTEND-PAGING-AREA
VOLUME = *NONE / list-poss(256): <vsn 1..6> ,UTILIZATION = *MEDIUM / *HIGH / *LOW ,LATER-REDUCTION = *NOT-REQUESTED / REQUESTED

Operands

VOLUME = *NONE / list-poss(256): <vsn 1..6>
Specifies the volume serial number (VSN) of the disk on which the paging file being added is located. Up to 256 disks can be specified.

UTILIZATION = *MEDIUM / *HIGH / *LOW
Defines the utilization of the paging file. This affects both use of the paging file and the duration and I/O rate of a paging file reduction.
Paging management takes this specification into account as far as possible, but it cannot guarantee that a paging file will be utilized in the desired manner. This ultimately depends on the actual paging rate and on the behavior of the paging load over time.

UTILIZATION = *MEDIUM

This default setting should be changed only in exceptional circumstances.

The paging file is to be used in accordance with the paging strategy, which means that paging files will be utilized evenly in proportion with their size.

UTILIZATION = *HIGH

This setting should be selected only in exceptional circumstances.

The paging file is to be utilized more intensively than envisaged in the paging strategy (the paging file will hold pages more frequently). This is typically helpful with small paging files, as it is a way to reduce the load on disks with large files, allowing the paging files to be utilized evenly, irrespective of size. It is also possible to cut the time required for the reduction of large paging files.

UTILIZATION = *LOW

This setting should be selected only in exceptional circumstances.

The paging file is to be utilized less intensively than envisaged in the paging strategy (the paging file will hold pages less frequently). This setting is particularly advisable if the paging file is intended to be removed from the paging area (reduction) or if frequent DMS I/O is to be expected on a paging disk in addition to the paging I/O (primarily applies to paging files in the home pubset).

LATER-REDUCTION = *NOT-REQUESTED / *REQUESTED

This operand specifies how likely it is that the paging area will subsequently be reduced by removal of the paging file(s). This specification is evaluated when the paging files attached by this command are detached.

LATER-REDUCTION = *NOT-REQUESTED

This default setting should be changed only in exceptional circumstances.

Subsequent reduction of the paging area by removal of the paging files is not intended. If reduction is carried out nonetheless, the acceleration mechanism will not be used (no additional resources will be requested).

LATER-REDUCTION = *REQUESTED

This setting should be selected only in exceptional circumstances.

It is possible that the paging area will later be reduced by removal of the paging file(s) attached by this command.

When assigning this operand value you should bear in mind that main memory utilization will be greater the whole time that the paging file is in the paging area if later reduction of the paging area is envisaged (performance accelerated by requesting additional resources) and that this is very likely to result in a higher paging rate, which in turn may have a negative impact on the overall performance of the system. Consequently it is possible that extension of the paging area will have to be rejected. So *REQUESTED should be used with caution.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command successfully executed.
1		CMD0001	The paging file on the specified disk is already in use. Guaranteed messages: EMM2830
	32	EMM2800	The function cannot be executed owing to an internal error. Guaranteed messages: EMM2818, EMM2828
	64	EMM2802	The specified disk is unknown, the paging file does not have the required file attributes, or the name of the paging, or the paging file is greater than 32 Gbytes. Guaranteed messages: EMM2817, EMM2837, EMM2316
	64	EMM2803	The paging file which is to be added to the paging area is on GS volume (disk emulationn global storage) and not on disk. Guaranteed message: EMM2835
	64	EMM2803	The pubset is a shared pubset. Guaranteed message: EMM2834
	64	EMM2804	There is no paging file on the disk with the specified name. Guaranteed messages: EMM2832
	64	EMM2805	The pubset has not been imported, the SM pubset has not been fully imported, the master catalog entry of the pubset is missing, or the disk cannot be reserved, a disk error has occurred, the paging file cannot be processed owing to an I/O error. Guaranteed messages: EMM2825, EMM2836, EMM2838, EMM2839
	64	EMM2806	Adding the paging file takes the size of the paging area over the 32-Tbyte limit, or the number of paging files or paging partitions is more than the maximum permitted number of 4096 paging files. Guaranteed messages: EMM2840, EMM2842
	128	EMM2807	The function cannot be implemented because there are not enough resources available. Guaranteed messages: EMM2819, EMM2829

i If a list is specified command processing is aborted in the event of an error with a return code of EMM2800, EMM2806 or EMM2807. In all other cases processing continues.

Notes

Message EMM2301 is issued for each paging file added. On completion of command processing, message EMM2850 is issued. This message indicates the current availability of paging space and the current paging space requirements.

The pubset to which the disk containing the paging file belongs must have been imported. It must not be a shared pubset or an SM pubset.
The disk from which the paging file is taken may also be a DRV (Dual Recording by Volume) disk, but in this case the pages will be written just once to one of the two DRV disks.

Examples

Adding the paging file on the disk with the volume serial number 2OSW.0 to the paging area:

```
/EXTEND-PAGING-AREA VOLUME=2OSW.0
```

EXTEND-TAPE-SET

Extend volume serial number set (tape set)

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING
Routing code:	$ (with NBCONOPI=N) or E (with NBCONOPI=Y)

Function

The EXTEND-TAPE-SET command allows a set of VSNs, which can be used for the creation of tape files, to be extended by the user. The use of this command implies that all the files which are linked to the same TAPE-SET-NAME use the same tape set. The set of VSNs is recorded in the tape set table (TST). Within the TST, the TAPE-SET-NAME has the same function as the file link name within the TFT.
A TST entry can hold no more than 5455 VSNs.

Format

```
EXTEND-TAPE-SET

 TAPE-SET-NAME = <alphanum-name 1..4>
,VOLUME = [*ANY](...) / list-poss(255): <alphanum-name 1..6>
    [*ANY](...)
      |    NUMBER-OF-DEVICES = 1 / <integer 1..9>
,DEVICE-TYPE = *TAPE / <device>
```

Operands

TAPE-SET-NAME = <alphanum-name 1..4>
The name of the tape set.

The tape set name may not be used simultaneously as a file link name within the same task.

VOLUME = *ANY(...) / list-poss(255): <alphanum-name 1..6>
The VSNs of the additional tapes which are to be allocated to the existing tape set.

VOLUME = *ANY(...)
Allows the operator, or MAREN if available, to mount any tape on the specified device type (no particular VSN is required).

NUMBER-OF-DEVICES = 1 / <integer 1..9>
Number of tapes required.

DEVICE-TYPE = *TAPE / <structured-name 1..8>
The device type to which the tapes are assigned.
The value to be specified is the “volume type” of the tapes, from which the Device Management System determines the device type which is to be made available for the tape processing.
Only device types or volume types known in the system are accepted. In interactive mode, DEVICE-TYPE=? calls up a list of the available device and volume types.
The default value is TAPE, i.e. for tape processing the devices to be used should support a recording density of 6250 bpi (bytes per inch). Other permissible specifications can be found in section “Device types for DMS tape processing” on page 1-84.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
	1	CMD0202	Syntax or semantic error in command
	32	DMS0584	A state that does not allow the function to continue was reported during processing
	32	DMS05C7	Unexpected internal error in DMS
	64	DMS0586	It is not possible to access or reserve a volume at present
	64	DMS0587	Use of the specified command has been restricted by the system administrator
	64	DMS06DD	Tape set not found
	64	DMS06FF	BCAM connection severed
	130	DMS0524	System address space exhausted
	130	DMS0586	It is not possible to access or reserve a volume at present
	130	DMS0594	Not enough virtual memory available

FORCE-DESTROY-CACHE

Force deletion of PCA cache area

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Caching media control Pubset and MRSCAT management
Domain:	MULTI-CATALOG-AND-PUBSET-MGMT
Privileges:	TSOS

Function

Using this command, the system administration can force a PCA cache area in a cache medium to be deleted. As part of the “PFA concept” (Performant File Access) the user is given the possibility of using the DMS interfaces to provide buffer storage for files (“caching” them). The cache media supported via the PFA interfaces are: main memory (MM) and global storage (GS). The DAB subsystem is required as the cache handler driver software for the use of these media.

The FORCE-DESTROY-CACHE command enables systems support to force PFA cache areas to be deleted without writing the data in the cache out to disk. No allowance is made for resultant data inconsistency on the disks which are affected. Hence the command is intended only for exceptional circumstances (such as disk or cache defects) in which the buffered data can no longer be transferred to disk by a STOP-PUBSET-CACHING or EXPORT-PUBSET command).

Before the command is executed a dialog (message ECC0501) asks if the action is really to be performed in spite of the risk of data inconsistency.
Once the command has executed successfully, message ECC0502 is sent to SYSOUT and the operator console.

The effect of the command is global; i.e. it applies to all the systems which are attached to the cache area.

The command should be applied only to exported pubsets. FORCE-DESTROY-CACHE cannot be used to delete ADM-PFA cache areas. For the cache media main memory and global storage, these areas can be released using FORCE-STOP-DAB-CACHING (see the “DAB” User Guide [5]).

Format

FORCE-DESTROY-CACHE
CACHE-MEDIUM = *MAIN-MEMORY / *GS ,CACHE-ID = <alphanum-name 1..4>

Operands

CACHE-MEDIUM =
Identifies the storage medium which is used as the buffer store for the data.

CACHE-MEDIUM = *MAIN-MEMORY
The cache area to be deleted is in main memory.
If the identifier of the cache area is not found in this medium, the command is rejected with an error message.

CACHE-MEDIUM = *GS
The cache area to be deleted is in global storage (see also “The global storage (GS) medium” in the “Introduction to System Administration” [14]).
If the identifier of the cache area is not found in this medium, the command is rejected with an error message.

CACHE-ID = <alphanum-name 1..4>
The identifier of the cache area which is to be deleted from the specified cache medium (corresponds to the ID of the pubset whose data is being buffered).

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
	1	CMD2201	Error at the interface between SDF and the command server
	3	CMD2203	SDF version is not supported
	32	ECC0001	Internal error
	64	CMD0216	No authorization to invoke the command
	64	ECC0301	Cache area does not exist
	64	ECC0302	Cache area could not be deleted
	64	ECC0006	DAB cache handler not loaded
	64	ECC0502	Cache area has been deleted
	64	ECC0005	Command withdrawn by the user

Example

Cache area ABCD (the pubset ID of a pubset which is attached to the computer) is in global storage:

```
/force-destroy-cache cache-medium=*gs,cache-id=abcd
% ECC0501 DO YOU REALLY WANT TO DESTROY THE CACHE WITH CACHE-ID 'ABCD'?
REPLY (YES; NO)
yes
% ECC0502 CACHE WITH CACHE-ID 'ABCD' HAS BEEN DESTROYED
```

FORCE-DESTROY-GS-PARTITION

Force release of partition in global storage

Description status:	GSMAN V19.0A
Functional area:	Caching media control Global storage administration
Domain:	DEVICE
Privileges:	TSOS

Function

The FORCE-DESTROY-GS-PARTITION command enables systems support to remove a partition from global storage (GS).
Unlike DELETE-GS-PARTITION, this command can delete a partition even if it is still in use or the partition still contains old data.
It cannot, however, delete a partition with a name beginning with “VIRTGS” if the partition is still in use (ACCESS with a value other than NONE).

In global GS operation in an XCS network the command applies to all the systems on the network. The GSMAN subsystem is not available until XCS has been started.

The command should be used only if it is not possible to delete the partition with the DELETE-GS-PARTITION command. If the partition is still in use, an attempt should first be made to terminate the application in question or to release the GS partition.

Systems support can use the SHOW-GS-STATUS command to check up on the current GS configuration.

Format

FORCE-DESTROY-GS-PARTITION
PARTITION**-ID** = <name 1..8>

Operands

PARTITION-ID = <name 1..8>
Name of the GS partition to be deleted from GS.If there is no partition with this name, the command will be rejected and a message to that effect will be issued.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
	1	CMD0202	Syntax error
	32	CMD0221	Internal error
	64	CMD0216	No authorization to invoke command
	64	EGC0005	Command aborted by user
	64	EGC0112	No GS available
	64	EGC0301	The specified partition does not exist
	64	EGC0304	The partition is still in use
	64	EGC1000	GS is not available
	128	EGC0010	GSMAN subsystem is not loaded
	128	EGC0110	Command temporarily not executable
	129	CMD2280	Subsystem not available

FORCE-JOB-CANCEL

Cancel user job

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing
Domain:	JOB
Privileges:	TSOS OPERATING
Routing code:	P

Function

The FORCE-JOB-CANCEL command forces a user job to terminate.
The user job addressable by its TSN or a monitoring job variable is immediately canceled, without waiting for privileged system routines being used to run to completion. This procedure may in some circumstances lead to inconsistencies in some system tables. Hence it is advisable to use the FORCE-JOB-CANCEL command only if the CANCEL-JOB command is unsuccessful and continuation of the session depends on cancelation of the job. Observe the following guidelines on using the command:

- Use FORCE-JOB-CANCEL only as a last resort (if there is the risk of a deadlock).
- Leave 10 minutes between CANCEL-JOB and FORCE-CANCEL-JOB.
- Run SHOW-JOB-STATUS at the console beforehand to check the status of the job.
- Answer all unanswered messages at the console.
- FORCE-CANCEL-JOB must not be applied to jobs which are in a pass loop (Q13, Pendcode 04), as it will terminate the wrong job. It will terminate the job which is waiting for the lock situation to be cleared, not the job which caused the lock situation.

Execution of the command leads to termination of the task with the message

```
CANOKIL: ABNORMAL TASK TERMINATION ENFORCED BY COMMAND
```

and, in the case of privileged applications, to a system dump which is generated without any prior query to the operator.

ℹ Use of the command by the operator can be forbidden by means of the system parameter NRTKILL.

Format

```
FORCE-JOB-CANCEL

JOB-IDENTIFICATION = *TSN(...) / *MONJV(...)
   *TSN(...)
     |  TSN = <alphanum-name 1..4>
   *MONJV(...)
     |  MONJV = <filename 1..54 without-gen-vers>
,TEXT = *NO / <c-string 1..72>
```

Operands

JOB-IDENTIFICATION = *TSN(...) / *MONJV(...)
Selection of the job identification.

JOB-IDENTIFICATION = *TSN(...)
The job is addressed by its TSN.

TSN = <alphanum-name 1..4>
The TSN of the job in question.

JOB-IDENTIFICATION = *MONJV(...)
The job is not addressed by its TSN but via a monitoring job variable.

MONJV = <filename 1..54 without gen-vers>
Monitoring job variable of the job in question.

TEXT = *NO / <c-string 1..72>
Specifies whether a text should be output to SYSOUT, and if so which text, before the job is terminated. The specified string may be up to 72 characters in length. If the job which is to be terminated is monitored by a monitoring job variable, the first 51 characters are embedded in it (see the CANCEL-JOB command).

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	No error
2	0	CMD0002	Command executed with warning
	1	CMD0202	System error
	64	JMS0630	Semantic error
	64	JMS0640	MONJV error, access not allowed or wrong time to call command
	130	JMS0660	Repeat command later

FORCE-PUBSET-EXPORT

Force pubset export

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Pubset and MRSCAT management
Domain:	MULTI-CATALOG-AND-PUBSET-MGMT
Privileges:	TSOS OPERATING
Routing code:	R

Function

The FORCE-PUBSET-EXPORT command forces a pubset to be exported.
The specified pubset is exported without taking into account open files that cannot be restored until the next session. This command should therefore be used only in exceptional cases, since it cannot be guaranteed that the pubset involved can be imported again. This command cannot cancel the wait states for termination of the occupying tasks; it simply limits the wait time involved.

The command is rejected if it was not preceded by an EXPORT-PUBSET command with the operand TERMINATE-JOBS=*YES.

Format

FORCE-PUBSET-EXPORT
PUBSET = <cat-id 1..4> ,MONJV = *NONE / <filename 1..54 without-gen-vers> ,JV-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / <integer -2147483639..2147483639>

Operands

PUBSET = <cat-id 1..4>
Catalog ID of the pubset to be exported.

MONJV = *NONE / <filename 1..54 without-gen-vers>
Specifies whether a monitoring job variable is set.
This operand should only be specified if the JV software product is used.

Note

The job variable must already be cataloged, otherwise it is not set. However, EXPORT processing is continued even if the job variable is not set.

JV-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / <integer -2147483639..2147483639>
Password of the monitoring job variable, if this variable is protected by a write password.
The operand has the following special characteristics:

- The password entered is not logged.
- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
1	0	DMS0355	Same export already active
1	0	DMS0364	Pubset already unavailable
	1	CMD0202	Syntax error
	32	DMS035C	Invalid operands
	32	DMS0363	MRSCAT access error
	64	DMS035C	IMPORT-PUBSET required
	64	DMS0360	No authorization for command
	64	DMS0366	Export attempt for home pubset
	64	DMS036D	Invalid operand sequence
	64	DMS036E	FORCE attempt without TERMINATE
	130	DMS0351	Other import/export task active
	130	DMS035C	Maximum number of tasks reached
	130	DMS0362	Class 4 memory error

GET-JOB-FROM-VIRTUAL-DEVICE

Request print job from virtual printer

Description status:	SPOOL V4.9A
Functional area:	Controlling spoolout job
Domain:	SPOOL-PRINT-SERVICES
Privileges:	STD-PROCESSING PRINT-SERVICE-ADMINISTRATION

Function

The GET-JOB-FROM-VIRTUAL-DEVICE command requests a new print job from the virtual device. It is executed in batch mode only.

By default the original user file is copied into the calling task's user ID, and a definition of the print job is generated in a metadata file. Both files are created with a default name which contains the identification number of the print job as a name component. This identification number is stored in the specified S variable.

Optionally a user-specific file name prefix can be defined for each of the files. Creation of the files and S variable can also be suppressed.

The command is part of a set of four commands which enable an application to be created in the form of an S procedure (for an example, see the OPEN-VIRTUAL-DEVICE-DIALOG command). These commands manage the dialog between a virtual device and the application which was started in batch mode as an S procedure:

– OPEN-VIRTUAL-DEVICE-DIALOG
– GET-JOB-FROM-VIRTUAL-DEVICE
– RETURN-JOB-TO-VIRTUAL-DEVICE
– CLOSE-VIRTUAL-DEVICE-DIALOG

Format

GET-JOB-FROM-VIRTUAL-DEVICE
FILE-PREFIX = ***STD** **/ *NONE /** <partial-filename 1..25>
,**METADATA-PREFIX** = ***STD** **/ *NONE /** <partial-filename 1..25>
,**OUTPUT-DOC-ID** = ***STD** **/ *NONE /** <structured-name 1..20>

Operands

FILE-PREFIX = *STD / *NONE / <partial-filename 1..25>
Specifies whether the original user file is to be copied and which prefix should be assigned to the name of the copy.

FILE-PREFIX = *STD
The original user file is copied under the following file name:
$<userid>.S.VIRTUAL.DOC.<document-id>.000
where <userid> is the user ID of the calling task and <document-id> is the identification number of the print job (see the OUTPUT-DOC-ID operand).

FILE-PREFIX = *NONE
The original user file is not to be copied.

FILE-PREFIX = <partial-filename 1..25>
The original user file is copied with a prefix under the following file name:
$<userid>.<file-prefix>.<document-ID>.000
where <userid> is the user ID of the calling task and <document-id> is the identification number of the print job (see the OUTPUT-DOC-ID operand).

METADATA-PREFIX = *STD / *NONE / <partial-filename 1..25>
Specifies whether metadata of the print job is to be created and which prefix should be assigned to the metadata file.

METADATA-PREFIX = *STD
The metadata of the print job is to be created in a file with the following name:
$<userid>.<metadata.prefix>.<document-id>
where <userid> is the user ID of the calling task and <document-id> is the identification number of the print job (see the OUTPUT-DOC-ID operand).

METADATA-PREFIX = *NONE
No file containing the metadata of the print job is to be created.

METADATA-PREFIX = <partial-filename 1..25>
The metadata of the print job is to be created in a file with the specified prefix in the name:
$<userid>.<metadata-prefix>.<document-id>
where <userid> is the user ID of the calling task and <document-id> is the identification number of the print job (see the OUTPUT-DOC-ID operand).

OUTPUT-DOC-ID = *STD / *NONE / <structured-name 1..20>
Specifies whether the identification number of the print job is to be output in an S variable.

OUTPUT-DOC-ID = *STD
The identification number of the print job is generated in the DOC-ID variable.

OUTPUT-DOC-ID = *NONE
The identification number of the print job is not generated.

OUTPUT-DOC-ID = <structured-name 1..20>
The identification number of the print job is generated in the specified S variable.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error; command successfully processed
	32	SCP0974	Unexpected command
5	32	SCP0974	Memory request error

Notes

1. If the original user file was copied (value of the FILE-PREFIX operand not equal to *NONE), normalization of this file takes place in the target file. This means that the original user file is copied into a SAM file (conversion, library member is extracted, etc.). This file is stored as not shareable under the calling task's user ID.
2. The user ID assigned to the application (i.e. the user ID under which the calling application task is executed) corresponds to the user ID (default: SYSSPOOL) specified for the device record (PROGRAM-NAME operand).
3. The metadata file is created as an ISAM file with the file attributes KEY-POS=5, KEY-LEN=64, BUF-LEN=STD(2) and USER-ACC=OWNER-ONLY. The ISAM key (64 characters) corresponds to the attribute name and the field (consisting of 1024 characters) corresponds to the attribute value.
4. If the dialog cannot be initialized, the command is rejected and a return code is set.
5. If no initialization took place (OPEN-VIRTUAL-DEVICE-DIALOG command), the command is rejected and a return code is set.
6. If the print job has already been returned (RETURN-JOB-TO-VIRTUAL-DEVICE command), the command is rejected and a return code is set.
7. This command can only be used in batch mode.
8. Spin-off processing is activated each time an error is detected.

Output in S variable

Output information	Name of the S variable	T	Contents	Condition
Identification number of the associated print job	var(*LIST).DOC-ID	S	<document-id>	

GOTO

Branch to specified destination (S procedure)

Description status:	SDF-P-BASYS V2.5E
Functional area:	Procedures
Domain:	PROCEDURE
Privileges:	STD-PROCESSING OPERATING HARDWARE-MAINTENANCE SECURITY-ADMINISTRATION SAT-FILE-MANAGEMENT SAT-FILE-EVALUATION

Function

GOTO is an SDF-P control flow command.

A GOTO command within an *S procedure* causes a branch to a specified destination.
Command processing resumes with the command introduced by the specified label, which may only occur once within the relevant procedure. Only labels in SDF-P format (S labels) may be used.
Branches are only possible within the same command block or, in the case of block nesting, in a command block on a lower nesting level.
The GOTO command corresponds to the SKIP-COMMANDS command (without conditions) in a *non-S procedure*.

Restrictions

Users with SECURITY-ADMINISTRATION, SAT-FILE-EVALUATION or SAT-FILE-MANAGEMENT privilege can use the command in procedures only.

Format

GOTO
LABEL = <structured-name 1..255>

Operands

LABEL = <structured-name 1..255>
This defines the branch destination. Specification takes the form of a label in SDF-P format (S label). Command processing resumes with the command line introduced by this label.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	No error
	1	CMD0202	Syntax error
	1	SDP0118	Command in incorrect context
	1	SDP0223	Incorrect environment
	3	CMD2203	Incorrect syntax file
	32	CMD0221	System error (internal error)
	130	SDP0099	No further address space available

HELP-MSG-INFORMATION

Display text of system message

Description status:	MIP V19.0A
Functional area:	Job processing Message processing
Domain:	JOB MESSAGE-PROCESSING
Privileges:	STD-PROCESSING TSOS HARDWARE-MAINTENANCE OPERATING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION
Routing code:	@

Function

The HELP-MSG-INFORMATION command displays the text of a system message on SYSOUT. In addition, the user can request explanations of messages and specify the language in which message texts should be issued.

Format

HELP-MSG-INFORMATION	Alias: **HP** / **HPMSGI**
MSG-IDENTIFICATION = ***LAST** / <alphanum-name 4..7> ,**INF**ORMATION-**LEV**EL = ***MAX**IMUM / ***MED**IUM / ***MIN**IMUM ,**LANG**UAGE = ***STD** / <name 1..1>	

Operands

MSG-IDENTIFICATION = *LAST / <alphanum-name 4..7>
Number of the system message to be displayed.
Both 4-digit and 7-digit message numbers are permitted.
See “Note” on page 3-372.

MSG-IDENTIFICATION = *LAST
Repeats the system message last issued by the system during the current job. This does not include system messages previously requested with HELP-MSG-INFORMATION.

INFORMATION-LEVEL =
Amount of information to be displayed.

INFORMATION-LEVEL = *MAXIMUM
Displays the system message in unabbreviated form; in addition, an explanation of the message text is given.

INFORMATION-LEVEL = *MEDIUM
Displays the system message in unabbreviated form.

INFORMATION-LEVEL = *MINIMUM
Displays the system message in coded form.

LANGUAGE = *STD / <name 1..1>
Language identifier. The system message is to be given in the language assigned to the language identifier (D = German, E = English).

LANGUAGE = *STD
The default value defined by the MSGLPRI system parameter applies unless a different value has meanwhile been set using the MODIFY-MSG-ATTRIBUTES command.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	No error Guaranteed message: NMH1120
2	0	NMH9006	Warning Guaranteed messages: NMH1165, NMH1166
	1	NMH1174	Syntax error Guaranteed message: NMH1155
	32	NMH1121	Internal error
	64	CMD0216	Semantic error
	64	NMH1155	Semantic error Guaranteed message: NMH1155
	64	NMH1199	Semantic error

Note

If MIP itself sent the message to SYSOUT, any inserts in the message will have been replaced by current values. If it was some other system component, not MIP, that sent the message to SYSOUT, the insert texts in the message are those that MIP placed in the message when the other system component requested the message text. These values may also be internal labels set by the requester for future processing.

Example

```
/cpf xx.dummy,aaaaa  ——————————————————————————————————————————————————————————————  (1)
%  DMS0812 '0533' DMS ERROR CODE OCCURRED DURING EXECUTION OF THE JOB FOR THE FI
LE NAME ':2OSG:$USER1.XX.DUMMY' WITH THE FILE NAME ':2OSG:$USER1.AAAAA'. FURTHER
 INFORMATION: /HELP-MSG DMS0533
/help-msg-information language=d  ————————————————————————————————————————————————  (2)
%  DMS0812 DMS-FEHLERCODE '0533' BEI AUFTRAGSAUSFUEHRUNG FUER DEN DATEINAMEN ':2
OSG:$USER1.XX.DUMMY' MIT DATEINAME ':2OSG:$USER1.AAAAA' AUFGETRETEN. WEITER INFO
RMATION: /HELP-MSG DMS0533
%  ? Fuer das vorausgegangene Kommando wurde ein Fehler-Protokoll angefordert.
%    (&00): DMS-Returncode von der Funktionsausfuehrung.
%    (&01): Ausgewaehlter, vollqualifizierter Dateiname.
%    (&02): Zweiter bzw. neuer, ggf. konstruierter Dateiname.
%  ! Siehe Massnahme-Text zum Fehlercode DMS(&00).
/help-msg-information msg-id=dms0533,language=d  ———————————————————————————————  (3)
%  DMS0533 ANGEGEBENE DATEI IN PUBSET '(&00)' NICHT GEFUNDEN. KOMMANDO BEENDET
%  ? Diese Meldung wird von DVS-Kommandos ausgegeben. Die angeforderte Datei
%    ist im gewuenschten Pubset nicht katalogisiert.
%  MASSNAHME : KEINE
/help-msg-info info-level=*medium,language=d  ———————————————————————————————————  (4)
%  DMS0812 DMS-FEHLERCODE '0533' BEI AUFTRAGSAUSFUEHRUNG FUER DEN DATEINAMEN ':2
OSG:$USER1.XX.DUMMY' MIT DATEINAME ':2OSG:$USER1.AAAAA' AUFGETRETEN. WEITER INFO
RMATION: /HELP-MSG DMS0533
```

(1) The system issues message DMS0812 in response to an invalid COPY-FILE command.

(2) The message last issued by the system is to be displayed together with an explanation in German. The message text is output in unabbreviated form.

(3) The German message text of message DMS0533 is to be displayed. Here, the command does not refer to the message last issued by the system and therefore may be entered at any point in the job.

(4) The message last issued by the system is requested in its German form: DMS0812. Message DMS0533 is not considered, since the other message was explicitly requested.

HELP-SDF

Provide information on SDF

Description status:	SDF V4.7D
Functional area:	SDF control
Domain:	SDF
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The HELP-SDF command displays information regarding SDF.

The information available is subdivided into sections that can be accessed via the individual operands of the command.

When the user enters HELP-SDF without operands or only with the preset operand values, a brief introduction to SDF is displayed (for a description see “HELP-SDF with preset operand values” on page 3-376).

Entering /HELP-SDF? displays the operand form for the command (temporarily guided dialog), and the information required can be selected.

Format

```
HELP-SDF                                                          Alias: HPSDF

 GUIDANCE-MODE = *NO / *YES
,SDF-COMMANDS = *NO / *YES
,ABBREVIATION-RULES = *NO / *YES
,GUIDED-DIALOG = *YES (...)

   *YES(...)
     |    SCREEN-STEPS = *NO / *YES
     |   ,SPECIAL-FUNCTIONS = *NO / *YES
     |   ,FUNCTION-KEYS = *NO / *YES
     |   ,NEXT-FIELD = *NO / *YES

,UNGUIDED-DIALOG = *YES (...) / *NO

   *YES(...)
     |    SPECIAL-FUNCTIONS = *NO / *YES
     |   ,FUNCTION-KEYS = *NO / *YES
```

Operands

GUIDANCE-MODE = *NO / *YES
Type of dialog guidance (output see page 3-377).

SDF-COMMANDS = *NO / *YES
SDF-specific commands (output see page 3-378).

ABBREVIATION-RULES = *NO / *YES
Abbreviation rules for the command syntax (output see page 3-379).

GUIDED-DIALOG = *YES(...) / *NO
Use of guided dialog (output see page 3-380).

GUIDED-DIALOG = *YES(...)
The description of guided dialog is subdivided into 4 blocks of information which can be accessed individually as follows:

> **SCREEN-STEPS = *NO / *YES**
> Sequence and contents of the menus (application domain menu, command menu, operand form, subform; output see page 3-380).
>
> **SPECIAL-FUNCTIONS = *NO / *YES**
> Special functions within the menu (output see page 3-380).

FUNCTION-KEYS = *NO / *YES
Function keys for menu control (output see page 3-381).

NEXT-FIELD = *NO / *YES
Input in the NEXT line for menu control (output see page 3-383).

UNGUIDED-DIALOG = *NO / *YES(...)
Use of unguided dialog (output see page 3-384).

UNGUIDED-DIALOG = *YES(...)
The description of unguided dialog is subdivided into 2 blocks of information which can be accessed individually as follows:

SPECIAL-FUNCTIONS = *NO / *YES
Special functions within the unguided dialog menu (output see page 3-384).

FUNCTION-KEYS = *NO / *YES
Function keys for controlling the unguided dialog menu (output see page 3-385).

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed
1	32	CMD0500	Syntax description in current syntax file invalid
1	64	CMD0810	Error during execution. Guaranteed messages: CMD0500, CMD0810

Information displayed with HELP-SDF:

HELP-SDF with preset operand values

```
%
% Introduction
%
% SDF is a convenient command interpreter and dialogue manager
% If the user desires, input can be entered as before.
% But the user can also utilize the advantages of SDF:
%  - Abbreviation mechanism
%  - Block input
%  - Guided dialogue
%  - Command bufferization (history)
%  - Definition of task-specific default values
%
% In addition, a more comprehensible command language has
% has been developed:
% The names of commands, operands and constant operand
% values were chosen so as to indicate clearly their function.
% Similar features (for example, the name of a file) are
% named accordingly similar naming rules (for example, FILE-NAME=
% or FROM-FILE= or TO-FILE=).
% Incorrect input is presented for correction along with
% a message.
% Commands which are part of the old commando language (ISP)
% can be input in the form "<old cmd>?". The new SDF-command
% is then displayed with its new syntax, i.e. a menu driven
% dialog allows to set the appropriate operand values.
% For further information, the user may enter "HELP-SDF?".
% The operand form for the command will then be displayed.
% Further information for an operand may then be obtained
% by setting that operand to "YES".
%
```

*HELP-SDF with GUIDANCE-MODE=*YES*

```
%
% Types of dialogue:
%
% 1. Guided dialogue
% Guided dialogue : supplies menus in the following order :
% Domain menu, Command menu, Operand form menu, sub-form menu.
% The extent of information offered in a menu can be controlled via
% the GUIDANCE operand of the MODIFY-SDF-OPTIONS command :
% - GUIDANCE=*MAXIMUM  all operand values with options (limit
%                     values etc.); help texts for commands
%                      and operands
% - GUIDANCE=*MEDIUM   all operand values;
%                      help texts only for commands
% - GUIDANCE=*MINIMUM  only default values for operands;
%                      no options; no help texts.
%
%
% 2. Unguided dialogue
% Unguided dialogue : no menus. There are 2 types of unguided
% dialogue, which can be controlled via the GUIDANCE operand
% of the MODIFY-SDF-OPTIONS cmd :
% - GUIDANCE=*NO:   the system requests command input with
%                 "%CMD:"; syntax error dialogue : correction
%                 of incorrect input without repetition
%                 of the entire command, detailed error
%                 messages; grouped command input :
%                 several commands, separated by logical
%                 end-of-line characters (LZE), can be issued
%                  at one time.
% - GUIDANCE=*EXPERT: the system requests command input with "/",
%                   no syntax error dialogue, detailed error
%                   messages; grouped command input.ch.
%
```

*HELP-SDF with SDF-COMMANDS=*YES*

```
%
% SDF-specific commands:
%
% - HELP-SDF
%    Displays information about SDF.
% - MODIFY-SDF-OPTIONS
%    Changes the task-specific SDF options.
% - REMARK
%    Inserts remarks into the command files.
% - RESET-INPUT-DEFAULTS
%    Deletes task-specific default values.
% - RESTORE-SDF-INPUT
%    Reprompts a previous input.
% - SHOW-CMD
%    Displays the syntax description of a command to SYSOUT/SYSLST.
% - SHOW-SDF-OPTIONS
%    Displays the task-specific information for the SDF options.
% - SHOW-INPUT-DEFAULTS
%    Outputs the task-specific default values.
% -  SHOW-INPUT-HISTORY
%    Displays the list of the latest inputs.
% - SHOW-SYNTAX-VERSIONS
%    Displays information about the contents of the active
%    syntax files.
% - WRITE-TEXT
%    Writes text to SYSOUT/SYSLST.
```

*HELP-SDF with ABBREVIATION-RULES=*YES*

```
%
% Abbreviation rules for the command syntax:
%
% -  Command name, operand name and constant operand values
%   (not basic data types) may be abbreviated as follows:
%   - Characters may be dropped from right to left
%     e.g.: TRANSFER-FILE abbreviated to TRANS
%   - Characters may be dropped from right to left
%     within the name elements.
%     e.g.: TRANSFER-FILE abbreviated to T-F
%          Restriction : Abbreviations must be unambiguous;
%          e.g.: CREATE as an abbreviation is ambiguous
%          between CREATE-FILE,CREATE-FILE-GENERATION,
%          CREATE-FILE-GROUP and CREATE-JV.
%
% - Operand values may also be specified without the
%   corresponding operand names (positional operands).
%   Here the following must be observed:
%   - A comma must be given for each omitted operand
%     preceding the positional operand.
%   - If, in the series of operands for a command,
%     an operand is specified completely (operand name and
%     operand value), subsequent operands may no longer be
%     specified as positional operands.
%
% - Guided dialogue: Operands for a structure need not be
%   in a sub-form, but may instead be entered in the operand
%   form. They must be enclosed in parentheses and must directly
%   follow the operand value introducing the structure.
%
% - Guided dialogue: In the NEXT line of the menu, complete
%   commands can also be entered. If a command name ends with
%   a question mark (<command>?), the operand form for the
%   specified command appears. It lists the operand values that
%   have already been specified (temporary guided dialogue).
%   If the specified command has no operands, the command help
%   text is displayed.
%
% - Ambiguity menus: entering a command name containing wildcards
%   and ending with a question mark causes SDF to display a menu
%   listing the commands which match the wildcard pattern.
%
```

*HELP-SDF with GUIDED-DIALOG=*YES(SCREEN-STEPS=*YES)*

```
%
% Guided dialogue: Order and content of the menus
%
% Domain menu    : Lists all domains (commands are assigned
%                 to particular domains according to their
%                 function). A domain can be selected by
%                number or name. The domain *ALL-COMMANDS lists all
%                 the commands.
% Command menu  : Lists all commands of the selected domain.
%                 A command can be selected by number or name
%                 A command followed by ! has no operands.
% Operand form  : Lists the operands of the selected command.
%                 The operand values desired are to be
%                 specified.
% Sub-form      : Lists the operands of a structure
%                 If an operand value introducing a structure
%                 is specified in an operand form, a sub-form
%                 for the corresponding structure is
%                 displayed.
% Command menu  : If a command has been executed or aborted,
%                 the current command menu appears again.
%
```

*HELP-SDF with GUIDED-DIALOG=*YES(SPECIAL-FUNCTIONS=*YES)*

```
%
% ? as operand value:
%         Supplies help text and options (limit values, etc.)
%         for the operand; in addition, detailed error
%         messages in the case of incorrect input.
% ?? as operand value:
%         calls up a help text, a display of the range of
%         values for the relevant operand and an help text
%         giving information about the data types associated
%          with the operand.
% !  as operand value:
%         Enters the default value for the operand.
% (  following an operand value introducing a structure:
%        Displays the sub-form for the structure associated
%         with the operand value.
% () following an operand value introducing a structure:
%        Suppresses the sub-form and enters the default
%        values for the operands of the structure.
%  -   as the last character in an input line:
%        A continuation line is displayed.
% LZF key :
%        Deletes all characters in the input line, starting
%         at the cursor.
%
```

*HELP-SDF with GUIDED-DIALOG=*YES(FUNCTION-KEYS=*YES)*

```
%
% Guided dialogue: Function keys
%
% The effect of the function keys depends on the FUNCTION-KEYS
% option (MODIFY-SDF-OPTIONS command/statement).
% 2 modes are possible : the *OLD-MODE and the *STYLE-GUIDE-MODE
% which offers a richer functionality.
%
% *OLD-MODE
%
% K1  EXIT function
%    Equivalent to *EXIT in NEXT line.
%    Equivalent to F3 in *STYLE-GUIDE-MODE (see below).
% K2  INTERRUPT function.
%    Interrupts the program or procedure which is currently
%    running or interrupt of command output.
% K3  REFRESH function
%    Screen refresh function.
%    Equivalent to *REFRESH in NEXT line.
%    Equivalent to F5 in *STYLE-GUIDE-MODE.
% F1  EXIT-ALL function
%    Equivalent to *EXIT-ALL in NEXT line.
%    Equivalent to F6 in *STYLE-GUIDE-MODE (see below).
% F2  TEST function
%    Checks inputs for syntax errors. Equivalent to *TEST in the
%    NEXT line.
% F3  EXECUTE function
%    Executes the current operation. Equivalent to *EXECUTE in
%    the NEXT line.
%    Equivalent to F11 in *STYLE-GUIDE-MODE (see below).
%
```

```
% *STYLE-GUIDE-MODE
%
% K2  INTERRUPT function.
%     Interrupts the program or procedure which is currently
%     running or interrupt of command output.
% F1  HELP function.
%     Causes the display of a guided screen.
%     Equivalent to entering "?" in the NEXT line.
% F3  EXIT function.
%     Exits the current menu or form and switches to the
%     higher-ranking menu.
%     Equivalent to *EXIT in NEXT line.
%     Equivalent to K1 in *OLD-MODE.
% F5  REFRESH
%     Screen refresh function.
%     Equivalent to *REFRESH in NEXT line.
%     Equivalent to K3 in *OLD-MODE.
% F6  EXIT-ALL function
%     Exits the current menu or form and switches
%     to the highest-ranking menu.
%     Equivalent to *EXIT-ALL in NEXT line.
%     Equivalent to F1 in *OLD-MODE.
% F7  BACKWARD paging function
%     Pages backward in the menu or form.
%     Equivalent to "-" in NEXT line.
% F8  FORWARD paging function
%     Pages forward in the menu or form.
%     Equivalent to "+" in NEXT line.
% F9  REST-SDF-IN function.
%     Causes the last command / statement to be restored.
%     Equivalent to RESTORE-SDF-INPUT.
% F11 EXECUTE function
%     Executes the current operation. Equivalent to *EXECUTE in
%     the NEXT line.
%     Equivalent to F3 in *OLD-MODE.
% F12 Cancel function.
%     Exits the current menu or form and switches
%PLEASE ACKNOWLEDGE
%     to the higher-ranking menu or form.
%     Equivalent to *CANCEL in NEXT line.
%
```

HELP-SDF with GUIDED-DIALOG=*YES(NEXT-FIELD=*YES)

```
%
% Guided dialogue: NEXT line
%
% +,++,-,--  : Pages forward and back in the menu.
%             + and - equivalent to F8 and F7 respectively
%              in *STYLE-GUIDE-MODE.
% *EXECUTE   : Executes the current command. Equivalent
%              to the F3 key in *OLD-MODE and F11 in
%              *STYLE-GUIDE-MODE.
% *CONTINUE  : Combination of + or *EXECUTE depending of the
%             value given in the actual questionnaire.
% *TEST     : Checks input for syntax errors. Equivalent
%              to the F2 key in *OLD-MODE.
% *EXIT     : exits the current menu or form and switches
%             to the higher-ranking menu. Equivalent
%             to the K1 key in *OLD-MODE and F3 key in
%              *STYLE-GUIDE-MODE.
% *EXIT-ALL  : exits the current menu or form and switches
%             to the highest-ranking menu. Equivalent
%PLEASE ACKNOWLEDGE
%             to the F1 key in *OLD-MODE and F6 key in
%              *STYLE-GUIDE-MODE.
% *CANCEL    : exits the current menu or form and switches
%             to the higher-ranking menu or form. Equivalent
%             to the F12 key in *STYLE-GUIDE-MODE.
% *REFRESH   : Repeats the last menu displayed. Equivalent
%              to the K3 key in *OLD-MODE and F5 in
%              *STYLE-GUIDE-MODE.
% *DOM-MENU  : Switches to the domain menu.
% (<domain>) : Displays the command menu of the domain <domain>.
% <command>? : Executes the current operation and displays
%             the operand form for the command <command>.
%             Operand values that have already been specified
%              are entered in the form.
% <command>  : Executes the current operation and then,
%             in addition, the specified command <command>.
% !<command> : Executes the current operation and defines the
%            specified operand values as task-specific
%            default values of the command. When used in
%            combination with <command>? allows the default
%            values to be entered in the operand form.
% *DOWN(<operand>) : Displays the sub-form for the operand
%              value <operand>.
% *UP       : Switches from the sub-form back to the
%              operand form.
% ?         : Increases the guidance level.
% ??        : Further increases the guidance level.
%
```

*HELP-SDF with UNGUIDED-DIALOG=*YES(SPECIAL-FUNCTIONS=*YES)*

```
%
% Unguided dialogue: Special functions
%
%  ?  as operand value :
%       Supplies help text and options
%       (limit values, etc.)
% ?  as command :
%       Displays the domain menu.
%       (temporary guided dialogue)
% ?? as operand value :
%       calls up a help text, a display of the range of
%       values for the relevant operand and an help text
%       giving information about the data types associated
%       with the operand.
% <command>?  as command :
%       Displays the operand form for the command
%       <command> (temporary guided dialogue).
%       Operand values that have already been
%       specified are filled in.
%       If there are no operands, the help text of the
%       command is displayed.
% !<command> as command :
%       Defines the specified operand values as task-specific
%       default values of the command. When used in
%       combination with <command>? allows the default
%       values to be entered in temporarily unguided dialog.
% ^ or *SECRET as operand value (only for passwords):
%       Causes the input field for the password to
%       be kept blank.
% LZF key :
%       Deletes all characters in the input line, starting
%       at the cursor.
% LZE key :   enables blocked input. Several commands can be
%       given separated by LZEs and executed at the same time
%       by a terminating DUE
%
```

*HELP-SDF with UNGUIDED-DIALOG=*YES(FUNCTION-KEY=*YES)*

```
%
% Unguided dialogue: Function keys
%
% The effect of the function keys depends on the FUNCTION-KEYS
% option (MODIFY-SDF-OPTIONS command/statement).
% 2 modes are possible : the *OLD-MODE and the *STYLE-GUIDE-MODE
% which offers a richer functionality.
%
% *OLD-MODE
%
% K1  EXIT function.
%    Terminates the program which is currently running.
%    A message is issued asking the user whether or not
%    he wants to terminate.
%    Equivalent to F3 in *STYLE-GUIDE-MODE.
% K2  INTERRUPT function.
%    Interrupts the program or procedure which is currently
%    running or interrupts the command output.
% F1  EXIT-ALL function.
%    Terminates the program which is currently running.
%    A message is issued asking the user whether or not
%    he wants to terminate.
%    Equivalent to F6 in *STYLE-GUIDE-MODE.
%
% *STYLE-GUIDE-MODE
%
% K2  INTERRUPT function.
%    Interrupts the program or procedure which is currently
%    running or interrupt of command output.
% F1  HELP function.
%    Causes the display of a guided screen.
%    Equivalent to entering "?".
%
%
% F3  EXIT function.
%    Terminates the program which is currently running.
%    A message is issued asking the user whether or not
%    he wants to terminate.
%    Equivalent to K1 in *OLD-MODE.
% F6  EXIT-ALL function.
%    Terminates the program which is currently running.
%    A message is issued asking the user whether or not
%    he wants to terminate.
%    Equivalent to F1 in *OLD-MODE.
% F9  RESTORE-SDF-INPUT function.
%     Causes the last command / statement to be restored.
%     Equivalent to RESTORE-SDF-INPUT.
% F12 CANCEL function.
%     Terminates the program which is currently running.
%     A message is issued asking the user whether or not
%     he wants to terminate.
%
```

HOLD-ALIAS-SUBSTITUTION

Suspend substitution functions of alias catalog

Description status:	ACS V19.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The HOLD-ALIAS-SUBSTITUTION command halts alias substitutions for the current task. The prefix insertion and SPOOL-FILE-PUBSET functions of ACS are not affected by this command.
The user can cancel the effect of this command and resume alias substitution with the RESUME-ALIAS-SUBSTITUTION command.
The HOLD-ALIAS-SUBSTITUTION command can be specified several times in sequence, but each HOLD-ALIAS-SUBSTITUTION command must be canceled with a separate RESUME-ALIAS-SUBSTITUTION command.

Format

HOLD-ALIAS-SUBSTITUTION

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed normally
2	0	ACS0042	Command has no effect: HOLD counter overflow
	128	ACS0018	ACS is not available
	130	ACS0036	Resource bottleneck

Example

Interrupting the replacement function, preventing replacement

```
/show-alias-cat                  ─────────────────────────────────────────────  (1)
%       ALIAS FILE NAME         ->     FILE NAME
%UB : ASS                       -> $.ASSEMBH
%UB : BSP                       -> LST.BSP.2
%SF : COB85                     -> $RZ.COBOL85
%SF : FORTRAN                   -> $RZ.FOR1
%UF : SF.SDF                    -> $TSOS.SYS.SDF.SYSTEM.SYNTAX
%  ACS0037 NUMBER OF ALIAS CATALOG ENTRIES: 5 (FOR SYSTEM: 2, FOR USER: 3)
/hold-alias-substitution         ─────────────────────────────────────────────  (2)
%  ACS0007 /HOLD-ALIAS-SUBSTITUTION PROCESSED
/show-acs-opt                    ─────────────────────────────────────────────  (3)
%
% ALIAS CATALOG SYSTEM V19.0
% =============================
%
% STATUS: IN HOLD
%
% ACTIVATED ALIAS CATALOG FILE(S):
%  ID                    U-INFO   DESCR    DATE       FILE NAME
% ACS-FOR-USER1         SYSADM04 ACS0193  2012-02-09 :2OST:$TSOS.OTTY.XY
% ENTRIES ADDED/MODIFIED BY INDIVIDUAL COMMANDS:       3
%
% LOGGING: ALIAS-SUBSTITUTION=STD, PREFIX-INSERTION=NO
% SUCCESS-MSG OPTIONS: USER-FILE=YES, SYSTEM-FILE=YES
% COMPLETE-ALIAS-NAMES=NOT-ALLOWED (USER-MODIF=NOT-ALLOWED)
% ALIAS-USERID        =NOT-ALLOWED (USER-MODIF=NOT-ALLOWED)
% STANDARD-RANGE=BOTH
/show-file-attr sf.sdf           ─────────────────────────────────────────────  (4)
%  DMS0533 REQUESTED FILE NOT CATALOGED IN PUBSET '2OS2'. COMMAND TERMINATED
/copy-file sf.robar,sf.sdf       ─────────────────────────────────────────────  (5)
/show-file-attr sf.sdf           ─────────────────────────────────────────────  (6)
%        48 :2OS2:$USER1.SF.SDF
%:2OS2: PUBLIC:      1 FILE  RES=        48 FRE=         7 REL=         0 PAGES
/resume-alias-substitution       ─────────────────────────────────────────────  (7)
%  ACS0008 /RESUME-ALIAS-SUBSTITUTION PROCESSED
%  ACS0009 ALIAS CATALOG OPERATION CONTINUES
/show-file-attr sf.sdf           ─────────────────────────────────────────────  (8)
%      5328 :2OST:$TSOS.SYS.SDF.SYSTEM.SYNTAX
%:2OST: PUBLIC:      1 FILE  RES=      5328 FRE=         1 REL=         0 PAGES
/show-file-attr $USER1.sf.sdf    ─────────────────────────────────────────────  (9)
%        48 :2OS2:$USER1.SF.SDF
%:2OS2: PUBLIC:      1 FILE  RES=        48 FRE=         7 REL=         0 PAGES
```

(1) The SHOW-ALIAS-CATALOG-ENTRY command displays all entries of the task-local alias catalog.

(2) The HOLD-ALIAS-SUBSTITUTION command is used to halt the ACS substitution function.

(3) The SHOW-ACS-OPTIONS command shows the current ACS settings for the task. The output field *STATUS* has the value *IN HOLD*, which means that the ACS substitution function has been interrupted.

(4) The specified file name *SF.SDF* in the SHOW-FILE-ATTRIBUTES command is not replaced, despite the fact that it matches a defined alias. A file with the actual name *SF.SDF* does not exist.

(5) The COPY-FILE command copies the contents of the file *SF.ROBAR* to the file *SF.SDF*. The alias *SF.SDF* is not replaced in this case either, but a file with the actual name *SF.SDF* is created.

(6) The specified file name *SF.SDF* in the SHOW-FILE-ATTRIBUTES command is replaced.

(7) The interrupted ACS substitution function is resumed with the RESUME-ALIAS-SUBSTITUTION command.

(8) The defined actual file name *$TSOS.SYS.SDF.SYSTEM.SYNTAX* is now substituted for the alias *SF.SDF* in the SHOW-FILE-ATTRIBUTES command before that command is executed.

(9) In order to retrieve the catalog entry for the file *SF.SDF*, the file name is specified along with the user ID (*$USER1.SF.SDF*). No substitution takes place in this case, since the specified name differs from that of the alias due to the user ID. Such a differentiation can also be achieved by specifying the catalog ID or the complete path name.

For further examples see the commands ADD-ALIAS-CATALOG-ENTRY, LOAD-ALIAS-CATALOG and SET-FILE-NAME-PREFIX.

HOLD-HARDWARE-AUDIT

Suspend AUDIT mode

Description status: BS2000 OSD/BC V10.0A

Functional area: AUDIT mode control

Domain: PROGRAM

Privileges: STD-PROCESSING
TSOS

Function

The HOLD-HARDWARE-AUDIT command suspends AUDIT mode, which was switched on using the START-HARDWARE-AUDIT command. The hold function always acts on the user’s own job, i.e. the AUDIT mode was switched on with SCOPE=*OWN-JOB.
The AUDIT tables are retained.
AUDIT mode may be resumed by entering the RESUME-HARDWARE-AUDIT command.

Privileged functions

Systems support personnel (TSOS privilege) can suspend hardware AUDIT mode for the privileged processor state TPR (STATE=*SYSTEM).

Format

HOLD-HARDWARE-**AUDIT**
STATE = ***USER** / ***SYS**TEM

Operands

STATE =
Processor state for which the AUDIT mode is to be suspended.
If the hardware AUDIT mode is to be deactivated for both processor states, the command has to be issued twice.

STATE = *USER
The hardware AUDIT mode is to be suspended for the nonprivileged processor state TU.
The hardware AUDIT table and the backup table, if there is one, are retained.

STATE = *SYSTEM
This operand value is reserved for privileged users.

The hardware AUDIT mode is to be suspended for the privileged processor state TPR. The hardware AUDIT table is retained.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error
	64	CMD0216	User does not have authorization
	64	IDA0001	Hardware is not active
	64	IDA0011	Privilege error
	64	IDA0023	Hardware AUDIT is not available because of missing authorization in the user entry
	64	IDA0024	Hardware AUDIT is not available because of the test options currently set for the active task

HOLD-JOB

Place user job in wait state

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing
Domain:	JOB
Privileges:	TSOS OPERATING
Routing code:	J

Function

This command places a user job that has not yet been started in the wait state HELD-BY-COMMAND.
The specified job, identified by its TSN or a defined job variable, is skipped by the job scheduler when it selects the jobs to be started. This wait state can only be revoked by the RESUME-JOB command or by changing the start attributes to START=*IMMEDIATE (MODIFY-JOB command). Information on the jobs in wait state can be retrieved using the SHOW-JOB-STATUS command. In this case, the wait state HELD-BY-COMMAND is displayed as TYPE1/HO.
Successful processing of the HOLD-JOB command is indicated at the operator terminal.
The command is rejected in the following cases:

- The job scheduler has already released the job for starting; jobs which have already started can be placed in the wait state by means of the HOLD-TASK command;
- The job to be placed in the wait state is an interactive or transaction job (category DIA or TP).
- The job to be placed in the wait state has the START=*IMMEDIATE attribute.

Format

```
HOLD-JOB
--------------------------------------------------------------------------
JOB-IDENTIFICATION = *TSN(...) / *MONJV(...) / <alphanum-name 1..4>

  *TSN(...)

    |  TSN = <alphanum-name 1..4>

  *MONJV(...)

    |  MONJV = <filename 1..54 without-gen-vers>
```

Operands

JOB-IDENTIFICATION =
The batch job can be identified by means of either its TSN or a defined monitoring job variable.

JOB-IDENTIFICATION = *TSN(...)
The job is identified by means of its TSN.

TSN = <alphanum-name 1..4)
TSN of the job to be placed in the wait state.

JOB-IDENTIFICATION = *MONJV(...)
The job to be suspended is addressed by means of a monitoring job variable.

MONJV = <filename 1..54 without-gen-vers>
Defined job variable for the job to be placed in the wait state.

JOB-IDENTIFICATION = <alphanum-name 1..4>
The job number of the job to be suspended.
This value is supported for reasons of compatibility only and is not available in guided dialog.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
	1	CMD0202	Syntax error
	32	CMD0221	System error
	64	JMS0630	Semantic error
	64	JMS0640	Command cannot be executed

HOLD-JOB-CLASS

Put job class on hold

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing
Domain:	JOB
Privileges:	TSOS OPERATING
Routing code:	J

Function

The HOLD-JOB-CLASS command puts a job class on hold.
Batch jobs for the suspended job class are placed in the job queue of the responsible job scheduler but are not released for starting. The command affects only job classes in which batch jobs are assembled; the system job class $SYSJC is also permitted.

The HOLD status is temporary and can be canceled by means of the RESUME-JOB-CLASS command. Successful execution of the command is acknowledged in a message issued at the console. The SHOW-SYSTEM-STATUS provides an overview of the job classes currently on hold.

Format

HOLD-JOB-CLASS
CLASS-NAME = <name 1..8>

Operands

CLASS-NAME = <name 1..8>
Name of the job class to be placed on hold.
The system administration defines the name using the JMU statement DEFINE-JOB-CLASS.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
	1	CMD0202	Syntax error
	32	CMD0221	System error
	64	JMS0630	Semantic error
	64	JMS0640	Command cannot be executed

Notes

- The command is also accepted for system job class $SYSJC.
- If the particular job scheduler is inactive at the time the command is issued, the command is rejected with an error message.
- Batch jobs submitted for a suspended job class are entered in the job queue of the appropriate job scheduler. These jobs are not released for starting.

HOLD-JOB-STREAM

Put job stream on hold

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing
Domain:	JOB
Privileges:	TSOS OPERATING
Routing code:	J

Function

The HOLD-JOB-STREAM command places a job stream on hold. The job scheduler that was active in the suspended job stream cannot carry out its function until the HOLD state has been canceled by means of the RESUME-JOB-STREAM command. During this time user jobs for the job classes assigned to the job stream continue to be accepted but are not released for starting.
The HOLD-JOB-STREAM command is also permitted for the system job stream $SYSJS, for which a total of 16 job streams can be managed.

The job stream is terminated normally if it takes place during the HOLD phase, and a message to this effect is issued at the operator terminal (see the STOP operand of the JMU statement DEFINE-JOB-STREAM in the “Utility Routines” manual [9]).

If a job stream cannot be placed in the wait state, a message to that effect is issued at the operator terminal.

The SHOW-SYSTEM-STATUS command can be used to show which job streams are in the HOLD state (INFORMATION operand, value *JOB-STREAM).

Format

HOLD-JOB-STREAM
STREAM-NAME = <name 1..8>

Operands

STREAM-NAME = <name 1..8>
Name of the job stream to be placed on hold.
The system administration defines the name using the JMU statement DEFINE-JOB-STREAM.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
	1	CMD0202	Syntax error
	32	CMD0221	System error
	64	JMS0630	Semantic error
	64	JMS0640	Command cannot be executed

HOLD-LINKAGE-AUDIT

Suspend linkage AUDIT mode

Description status:	BS2000 OSD/BC V10.0A
Functional area:	AUDIT mode controln
Domain:	PROGRAM
Privileges:	STD-PROCESSING TSOS

Function

The HOLD-LINKAGE-AUDIT command suspends the linkage AUDIT mode which had earlier been activated by a START-LINKAGE-AUDIT command. The hold function always relates to its own job, i.e. the linkage AUDIT mode was activated with SCOPE=OWN-JOB. The linkage AUDIT table and the backup table if there is one are retained.

The linkage AUDIT mode can be resumed with the RESUME-LINKAGE-AUDIT command.

Privileged functions

Systems support personnel (TSOS privilege) can suspend linkage AUDIT mode for the privileged processor state TPR (STATE=*SYSTEM).

Format

HOLD-LINKAGE-AUDIT
STATE = ***USER** / ***SYS**TEM

Operands

STATE =
Processor state for which the linkage AUDIT mode is to be suspended.
If the linkage AUDIT mode is to be suspended for both processor states at the same time, the command has to be issued twice.

STATE = *USER
The linkage AUDIT mode is to be suspended for the nonprivileged processor state TU. The linkage AUDIT table is retained.

STATE = *SYSTEM
This operand value is reserved for privileged users.
The linkage AUDIT mode is to be suspended for the privileged TPR processor state. The linkage AUDIT table and the backup table if there is one are retained.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error
	64	CMD0216	User does not have authorization
	64	IDA0052	Linkage AUDIT is not available because of missing authorization in the user entry
	64	IDA0053	Linkage AUDIT is not available because of the test options currently set for the active task
	64	IDA0060	AUDIT functions are not available on the entire system in this session

HOLD-PRINT-JOB

Suspend active spoolout job

Description status:	SPOOL V4.9A
Functional area:	Controlling spoolout job
Domain:	SPOOL-PRINT-ADMINISTRATION
Privileges:	STD-PROCESSING OPERATING PRINT-SERVICE-ADMINISTRATION SAT-FILE-MANAGEMENT SAT-FILE-EVALUATION SECURITY-ADMINISTRATION
Routing code:	O

Function

The HOLD-PRINT-JOB command suspends or resumes an active spoolout job.

The following comments apply to replay jobs assigned with START-TAPE-REPLAY and the operand value OUTPUT-MODE=*DIRECT:

– The tape containing the output file is positioned at the desired point, and output is resumed immediately via the specified device.

– Current output is stopped. All the spoolout jobs are written back to the SR.vsn file and provided with information relevant for resuming interrupted output. Resumption may then only be initiated by means of the START-TAPE-REPLAY command.

For output to a laser printer the spoolout job is resumed from the last file page correctly output by SPOOL. With laser printers which print on a page-by-page basis there are always a number of pages which are in an undefined state (i.e. somewhere between file page and print page). For this reason, if output is to resume within a file a maximum of 45 pages will be printed twice in the case of HP and HP90 printers.

Privileged functions

RSO device administrators and spool and cluster administrators can suspend or resume any print job scheduled for printing on any device that they manage. Only users with the PRINT-SERVICE-ADMINISTRATION or OPERATING privilege are allowed to specify a local printer (DEVICE-NAME=<alphanum-name 2..2>).
For more information on these user groups see the manuals “RSO” [32], “SPOOL” [43] and “Distributed Print Services” [10].

Format

```
HOLD-PRINT-JOB

 DEVICE-NAME = <alphanum-name 1..8> / <alphanum-name 2..2>
,RESUME-CONDITION = *IMMEDIATE / *BY-PRIORITY(...) / *BY-OPERATOR
    BY-PRIORITY(...)
      |  PRIORITY = *UNCHANGED / <integer 30..255>
,RESTART-POSITION = *BEGIN-OF-SPOOLOUT / *CURRENT-PAGE / *LAST-CHECKPOINT /
                    *PREVIOUS-CHECKPOINT / *PAGE(...) / *BACK(...)
    *PAGE(...)
      |  PAGE-NUMBER = <integer 1..10000000>
    *BACK(...)
      |  PAGES = <integer 1..10000000>
```

Operands

DEVICE-NAME = <alphanum-name 1..8> / <alphanum-name 2..2>

RSO device name or spool device name of the local printer on which the spoolout job is to be suspended.

RESUME-CONDITION = *IMMEDIATE / *BY-PRIORITY(...) / *BY-OPERATOR

The spoolout job is to be resumed immediately, resumed after a delay or suspended.

RESUME-CONDITION = *IMMEDIATE

The spoolout job is to be resumed immediately.

RESUME-CONDITION = *BY-PRIORITY(...)

The spoolout job is to be resumed after a delay.

PRIORITY = <integer 30..255>

Priority for the delayed resumption of the spoolout job.

PRIORITY = *UNCHANGED

The spoolout job is placed in the job queue with its original priority (TYPE 4).

RESUME-CONDITION = *BY-OPERATOR

The spoolout job is to be suspended until a RESUME-SPOOLOUT command is issued.

RESTART-POSITION =
Determines the point from which the file is to be output in the event of a restart. The restart point for a particular printer after a printer fault must not be confused with the CHECKPOINT-INTERVAL value from the SPOOL parameter file (M-S-P) which is the restart point after a SPOOL or system crash.

RESTART-POSITION = <u>*BEGIN-OF-SPOOLOUT</u>
The spoolout job is to be repeated from the start of file.

RESTART-POSITION = *CURRENT-PAGE
The spoolout job is to be restarted from the page which was being output at the time of the interruption.

RESTART-POSITION = *LAST-CHECKPOINT
The spoolout job is to be restarted from the last checkpoint. System administration uses the MODIFY-SPOOL-PARAMETERS statement in the SPSERVE utility routine to define the number of print pages after which a checkpoint is to be set.

RESTART-POSITION = *PREVIOUS-CHECKPOINT
The spoolout job is to be restarted from the last checkpoint but one.

RESTART-POSITION = *PAGE(...)
The spoolout job is to be restarted from a particular page; all print pages preceding this one are skipped.

> **PAGE-NUMBER = <integer 1..10000000>**
> Number of the page.

RESTART-POSITION = *BACK (...)
The spoolout job is to be restarted from the specified number of pages back from the current page.

> **PAGES = <integer 1..10000000>**
> Number of pages.

Note

> With SPS jobs, the operands *PAGE(...) and *BACK(...) must not be specified.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
	1	CMD0202	Syntax error
	1	SCP0973	Semantic error
	32	SCP0974	System error. Command rejected
	64	SCP0975	No authorization for command
	64	SCP0976	Invalid operand value

HOLD-PROCEDURE

Suspend procedure run to allow command input

Description status:	SYSFILE V19.0A
Functional area:	Procedures
Domain:	PROCEDURE
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The HOLD-PROCEDURE command suspends a procedure and issues a message:

```
TASK IS IN ESCAPE MODE AT LEVEL NUMBER i
```

where i = number of the current procedure level

The user can then enter commands at the terminal. This interrupt state is called ESCAPE mode. ESCAPE mode is maintained until it is terminated with a CANCEL-PROCEDURE or RESUME-PROCEDURE command.
The HOLD-PROCEDURE command is not permitted in batch mode. In interactive mode, it may be given only in procedure files, and so does not appear in the menu.

Format

HOLD-PROCEDURE

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
2	0	SSM2015	Command not permitted in batch mode
2	0	SSM2016	Command cannot be entered from the interactive terminal
	32	SSM1013	System error during command execution

Notes

- Nested interrupts are allowed (see Examples). A procedure called in ESCAPE mode can also be interrupted with HOLD-PROCEDURE. In this case the first interrupt is not lost (see Example 3 on page 3-406).
- In ESCAPE mode the symbolic operands that were defined in the interrupted procedure are available to the user and can be used from the terminal. However, a command with a symbolic operand that was replaced by the corresponding value is logged only if logging was requested for the interrupted procedure (see Example 4 on page 3-407).

Examples

Example 1: Execution of an EXIT-PROCEDURE command in ESCAPE mode

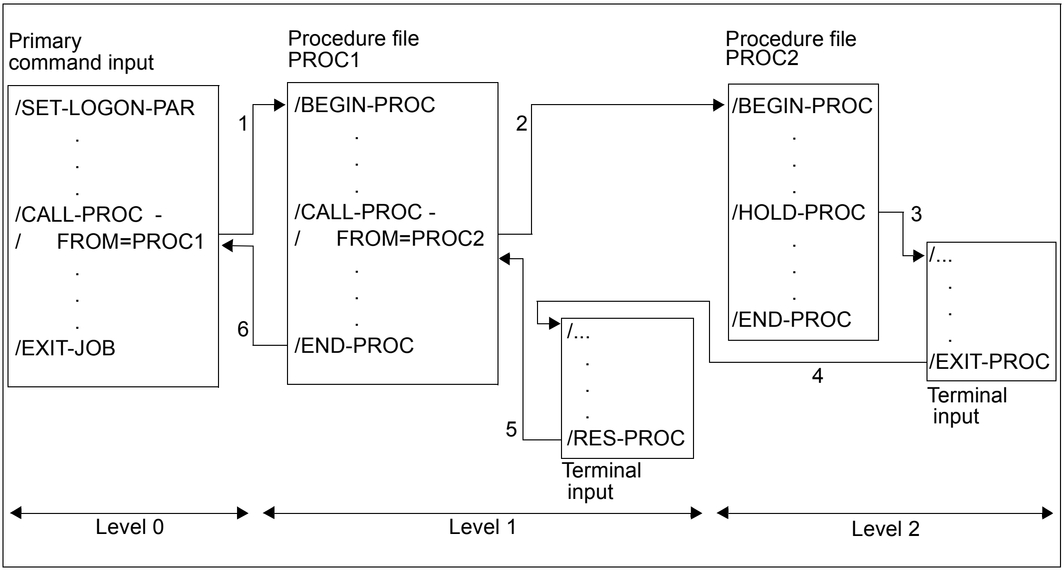


Figure 5: EXIT-PROCEDURE command in ESCAPE mode

The arrows indicate the sequence in which commands are processed (SYSCMD):

1. The command `/CALL-PROC FROM=PROC1` assigns SYSCMD to the procedure file PROC1.
2. The command `/CALL-PROC FROM=PROC2` assigns SYSCMD to the procedure file PROC2.

3. The command `/HOLD-PROC` activates ESCAPE mode: SYSCMD is assigned to the terminal. ESCAPE mode can also be activated by pressing K2 if this is allowed in the procedure. However, in this case it is not possible to precisely determine the interrupt point in advance.

4. EXIT-PROCEDURE causes a shift from procedure level 2 to procedure level 1; ESCAPE mode is maintained. (Note: If an RESUME-PROCEDURE command were entered instead of the EXIT-PROCEDURE command, PROC2 would be continued from the point of interruption.)

5. The RESUME-PROCEDURE command cancels ESCAPE mode. SYSCMD is again assigned to the procedure file PROC1.

6. SYSCMD is again assigned to primary command input.

Example 2: Execution of a CALL-PROCEDURE command in ESCAPE mode

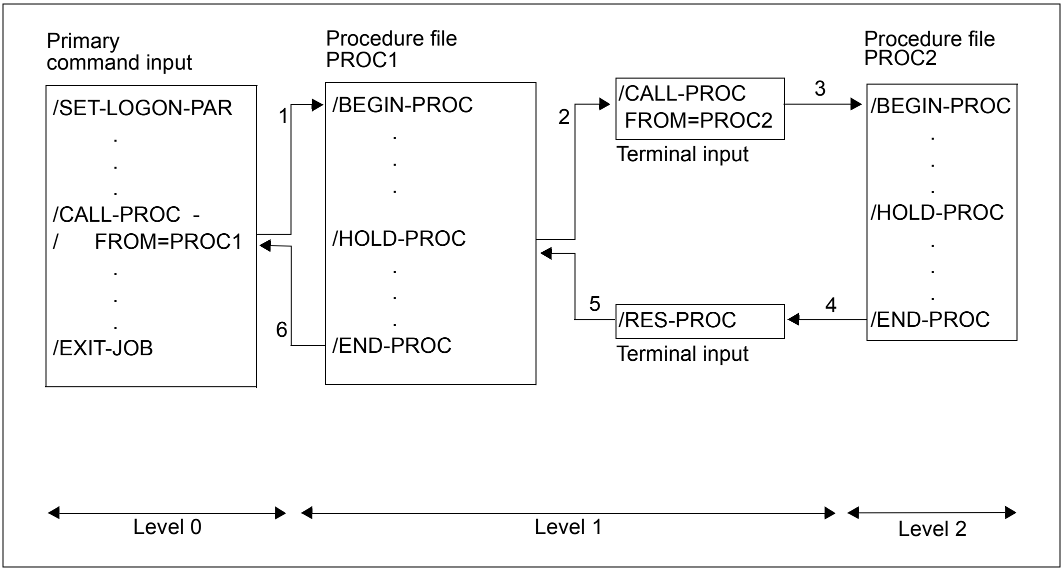


Figure 6: CALL-PROCEDURE command in ESCAPE mode

The arrows indicate the sequence in which commands are processed (SYSCMD):

1. The command `/CALL-PROC FROM=PROC1` assigns SYSCMD to the procedure file PROC1.

2. The command `/HOLD-PROC` activates ESCAPE mode: SYSCMD is assigned to the terminal. ESCAPE mode can also be activated by pressing K2 if this is allowed in the procedure. However, in this case it is not possible to precisely determine the interrupt point in advance.

3. The command `/CALL-PROC FROM=PROC2` assigns SYSCMD to the procedure file PROC2.
4. END-PROC causes a shift from procedure level 2 to procedure level 1; ESCAPE mode is still activated, i.e. SYSCMD is assigned to the terminal.
5. The RESUME-PROCEDURE command cancels ESCAPE mode. SYSCMD is again assigned to the procedure file PROC1.
6. SYSCMD is again assigned to primary command input.

Example 3: Nesting of ESCAPE levels

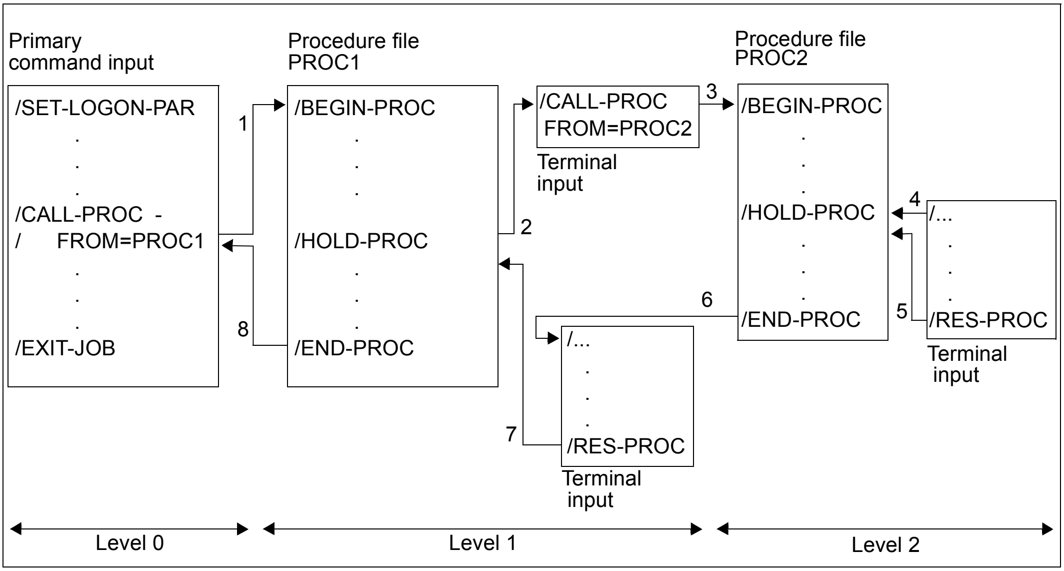


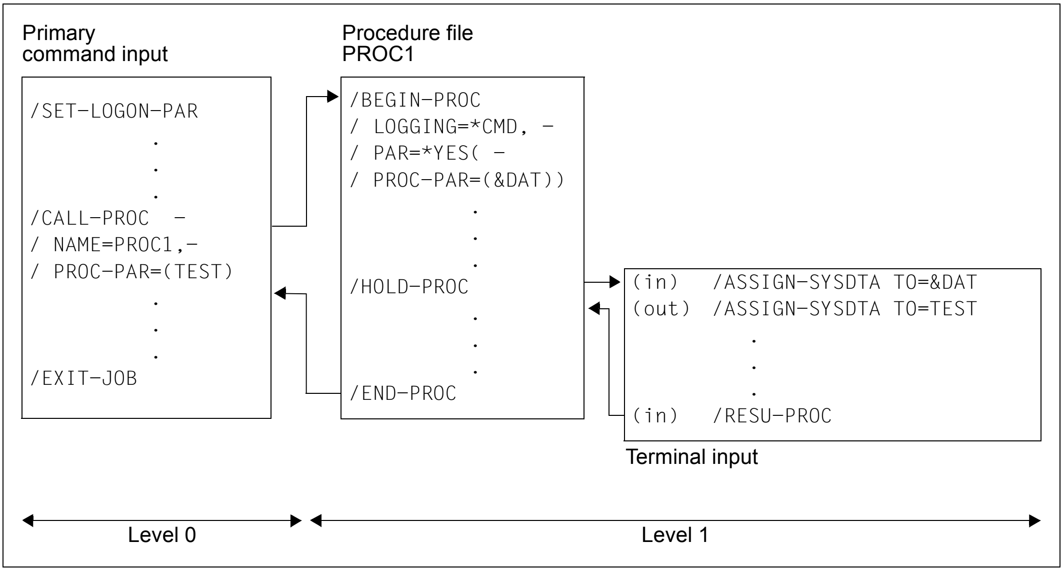
Figure 7: Nesting of ESCAPE levels

The arrows indicate the sequence in which commands are processed (SYSCMD):

1. The command `/CALL-PROC FROM=PROC1` assigns SYSCMD to the procedure file PROC1.
2. The command `/HOLD-PROC` activates ESCAPE mode: SYSCMD is assigned to the terminal. ESCAPE mode can also be activated by pressing K2 if this is allowed in the procedure. However, in this case it is not possible to precisely determine the interrupt point in advance.
3. The command `/CALL-PROC FROM=PROC2` assigns SYSCMD to the procedure file PROC2.

4. The command `/HOLD-PROC` activates ESCAPE mode: SYSCMD is assigned to the terminal. ESCAPE mode can also be activated by pressing K2 if this is allowed in the procedure. However, in this case it is not possible to precisely determine the interrupt point in advance.
5. SYSCMD is assigned to the procedure file PROC2.
6. SYSCMD is assigned to the terminal (ESCAPE mode is still activated at level 1).
7. The RESUME-PROCEDURE command causes SYSCMD to again be assigned to the procedure file PROC1.
8. SYSCMD is assigned to primary command input.

Example 4: Use of symbolic operands in ESCAPE mode



```
/SET-LOGON-PAR
          .
          .
          .
/CALL-PROC  -
/ NAME=PROC1,-
/ PROC-PAR=(TEST)
          .
          .
          .
/EXIT-JOB
```

```
/BEGIN-PROC
/ LOGGING=*CMD, -
/ PAR=*YES( -
/ PROC-PAR=(&DAT))
          .
          .
          .
/HOLD-PROC
          .
          .
          .
/END-PROC
```

```
(in)   /ASSIGN-SYSDTA TO=&DAT
(out)  /ASSIGN-SYSDTA TO=TEST
            .
            .
            .
(in)   /RESU-PROC
```

Figure 8: Input of symbolic operands in ESCAPE mode

The arrows indicate the sequence in which commands are processed (SYSCMD).

The file name TEST is entered for the symbolic operand “&DAT” when the procedure PROC1 is called with the CALL-PROCEDURE command. After leaving procedure mode with the HOLD-PROC command, an ASSIGN-SYSDTA command is issued at the terminal using the symbolic operand “&DAT”. Since logging of commands on SYSOUT was requested for the procedure file PROC1 (the operand LOGGING=*CMD specified in the BEGIN-PROCEDURE command), this command (supplied with the appropriate value TEST) is also logged on SYSOUT in ESCAPE mode.

HOLD-PROGRAM

Suspend program to allow command input

Description status: SYSFILE V19.0A

Functional area: Program control

Domain: PROGRAM

Privileges: STD-PROCESSING
SAT-FILE-EVALUATION
SAT-FILE-MANAGEMENT
SECURITY-ADMINISTRATION

Function

The HOLD-PROGRAM command suspends a program waiting for input data and permits the input of commands. This is only possible within a procedure or a batch job. In an interactive dialog the command will be rejected.
The command RESUME-PROGRAM terminates command input and effects a return to program mode.

Format

HOLD-PROGRAM

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
	64	EXC0100	HOLD-PROGRAM or BREAK command in SYSDTA data stream

HOLD-SUBSYSTEM

Put subsystem on hold

Description status:	DSSM V4.3B
Functional area:	Subsystem management
Domain:	SYSTEM-MANAGEMENT
Privileges:	OPERATING SUBSYSTEM-MANAGEMENT
Routing code:	R

Function

The HOLD-SUBSYSTEM command enables any required subsystem to be put on hold. No new connection is permitted to the specified subsystem; the required resources (holder task, address space) remain available. In addition, the FORCED operand enables the system to wait for all of the connected tasks to be disconnected or the subsystem to be put on hold immediately. Once the deinitialization phase is complete, the subsystem is on hold; this status can be canceled using the RESUME-SUBSYSTEM command.

The HOLD-SUBSYSTEM command is rejected if SUBSYSTEM-HOLD=*FORBIDDEN was specified when the subsystem was defined.

In order to ensure a high degree of parallelism and data integrity, time-consuming administration tasks are not performed under control of the calling task; instead they are transferred to a DSSM task.
As a rule, only the check of the requested function is carried out **synchronously** (i.e. contingent upon a wait state for the calling task). DSSM carries out the actual processing **asynchronously** and independent of the calling task.

Format

```
HOLD-SUBSYSTEM

 SUBSYSTEM-NAME = <structured-name 1..8>
,VERSION = *STD / <product-version mandatory-man-corr> / <product-version without-man-corr> / *HIGHEST
,SUBSYSTEM-PARAMETER = *NONE / <c-string 1..254>
,FORCED = *NO / *YES
,SYNCHRONOUS = *NO / *YES
```

Operands

SUBSYSTEM-NAME = <structured-name 1..8>
Name of the subsystem to be put on hold.

VERSION = *STD / <product-version mandatory-man-corr> / <product-version without-man-corr> / *HIGHEST
Specifies the version number.
If a version number is specified, the format specified here must be identical to the format used when the subsystem was defined (release and correction status mandatory or not allowed; see the description of the data type "product-version" on page 1-45).

VERSION = *STD
If there is only **one** loaded version of the subsystem, this version is selected.
If there are **several** suitable versions, the version must be specified.

VERSION = *HIGHEST
Selects the highest version of the subsystem.

SUBSYSTEM-PARAMETER = *NONE / <c-string 1..254>
Specifies whether special parameters, which can be evaluated only by the specified subsystem, are to be processed.

FORCED =
Determines the mode and urgency of command processing.

FORCED = *NO
The system waits for all tasks connected to the subsystem to be disconnected. When a task releases its connection depends on the subsystem. A connection to a nonprivileged subsystem (e.g. EDT) is released as soon as the program terminates.
Processing and hence normal termination of all the tasks accessing this subsystem is allowed to take its normal course.

FORCED = *YES
The subsystem is halted immediately.
In the case of a privileged subsystem, this can lead to system dumps of the tasks which are still connected; in the case of a nonprivileged subsystem, this can lead to user dumps or to the execution of the STXIT error handling routine which is offered by DSSM. Tasks that are connected to a nonprivileged subsystem can exit by means of the STXIT error handling routine which is offered by DSSM.

SYNCHRONOUS =
Enables synchronous or asynchronous processing to be selected.

SYNCHRONOUS = *NO
The command is to be processed asynchronously, i.e. there is no need to wait for it to execute before making another entry. Once the syntax of the command has been checked, the calling task is sent message ESM0216. No error messages relating to execution of the command will be output.

SYNCHRONOUS = *YES
The command must first be executed before another entry can be made. Accompanying error messages are output.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
1	0	CMD0001	No action necessary
	1	ESM0414	Syntax error: invalid version specified
	32	ESM0224	Command not processed
	32	ESM0228	Command terminated abnormally

HOLD-TASK

Put active batch task on hold

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing
Domain:	SYSTEM-TUNING
Privileges:	TSOS OPERATING
Routing code:	J

Function

The HOLD-TASK command puts a batch job that has already been started on hold. During this time, the devices reserved by the task remain allocated.
The task remains on hold until it is released explicitly using the RESUME-TASK command. Only after this release the job can be terminated with the CANCEL-JOB command if desired. The HOLD-TASK command is rejected in the case of interactive jobs, spoolout, ARCHIVE or transaction tasks.
The HOLD-TASK command refers exclusively to batch jobs which have already been started; jobs which have not yet been started can be managed by a HOLD-JOB command.

Format

```
HOLD-TASK

JOB-IDENTIFICATION = *TSN (...) / *MONJV(...)

  *TSN(...)
    |  TSN = <alphanum-name 1..4>

  *MONJV(...)
    |  MONJV = <filename 1..54 without-gen>
```

Operands

JOB-IDENTIFICATION =
The batch job which is to be put on hold can be identified by either its task sequence number (TSN) or a declared monitoring job variable (MONJV), as preferred.

JOB-IDENTIFICATION = *TSN(...)
Preset value: the job is identified by its task sequence number.

TSN = <alphanum-name 1..4>
Task sequence number of the batch task to be placed in the wait state.

JOB-IDENTIFICATION = *MONJV(...)
The job is identified by its monitoring job variable.

MONJV = <filename 1..54 without gen>
Specifies the monitoring job variable for the batch job which is to be put into the wait state.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
2	0	EXC0708	HOLD-TASK command already being processed
12	64	EXC0707	Task is in process of LOGOFF (EXIT-JOB)
12	64	EXC0710	Task is a transaction task (TP)
12	64	EXC0711	Specified job number (TSN) not found
12	64	EXC0715	Command not allowed for this type of task

IF

Start IF block

Description status: SDF-P-BASYS V2.5E

Functional area: Procedures

Domain: PROCEDURE

Privileges: STD-PROCESSING
HARDWARE-MAINTENANCE
SECURITY-ADMINISTRATION
SAT-FILE-MANAGEMENT
SAT-FILE-EVALUATION

Function

IF is an SDF-P control flow command.

The IF command initiates an IF block within an *S procedure*. Depending on the value of a logical expression (see section “SDF-P-BASYS” on page 1-131), the IF block makes it possible to execute alternative command sequences. The IF block is terminated by the END-IF command.
If the logical expression is true, all subsequent commands until the associated ELSE-IF, ELSE or END-IF command are processed.
If the logical expression is false, the associated ELSE-IF conditions are checked. If they are also false, processing resumes with the ELSE command.
If the IF block does not contain an ELSE-IF command block, control branches to the ELSE command block. If there is no ELSE command block either, the IF block is exited, i.e. processing resumes with the command following the END-IF command.

The ELSE-IF command is only available with the chargeable SDF-P subsystem.

Restrictions

Users with SECURITY-ADMINISTRATION, SAT-FILE-EVALUATION or SAT-FILE-MANAGEMENT privilege can use the command in procedures only.

Format

IF
CONDITION = <text 0..1800 with-low *bool-expr*>

Operands

CONDITION = <text 0..1800 with-low *bool-expr*>
Logical expression whose value determines processing in the IF block. The rules governing the formation of logical expressions are described in “SDF-P-BASYS” [34].

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
	1	CMD0202	Syntax error
	1	SDP0118	Command in incorrect context
	1	SDP0223	Incorrect environment
	3	CMD2203	Incorrect syntax file
	32	CMD0221	System error (internal error)
	64	SDP0091	Semantic error
	130	SDP0099	No further address space available

Example

```
/COND-1:     IF ( A = B )
/               commandsequence1     "executed if condition true"
/                 .
/                 .
/                 .
/           ELSE
/              commandsequence2  "executed if condition false"
/                 .
/                 .
/                 .
/           END-IF
```

Command sequence 1 is executed if variable A equals variable B. Otherwise command sequence 2 is executed.

IF-BLOCK-ERROR

Start IF-BLOCK-ERROR block

Description status:	SDF-P-BASYS V2.5E
Functional area:	Procedures
Domain:	PROCEDURE
Privileges:	STD-PROCESSING OPERATING HARDWARE-MAINTENANCE SECURITY-ADMINISTRATION SAT-FILE-MANAGEMENT SAT-FILE-EVALUATION

Function

IF-BLOCK-ERROR is an SDF-P control flow command.

The IF-BLOCK-ERROR command initiates block-specific error handling within an *S procedure*. The command block initiated in this way must be terminated by an END-IF command.

Restrictions

Users with SECURITY-ADMINISTRATION, SAT-FILE-EVALUATION or SAT-FILE-MANAGEMENT privilege can use the command in procedures only.

Type of error handling

The type of error handling in an S procedure depends on the setting of the ERROR-MECHANISM operand of the SET-PROCEDURE-OPTIONS command.

If nothing is specified in the procedure with regard to ERROR-MECHANISM or ERROR-MECHANISM=*SPIN-OFF-COMPATIBLE is defined, SDF-P activates error handling if a command supplies spin-off. Error handling in this instance does **not** on the command return code.
This error mechanism is the default case and means that S procedures do not change their behavior in the event of errors if commands contained in such procedures supply a command-specific command return code for the first time in a future version (whereas previously this was formed by SDF by analogy with spin-off).

If ERROR-MECHANISM=*BY-RETURNCODE is defined, SDF-P activates error handling when a command returns a command return code with a nonzero subcode 1. Error handling is in this case **not** dependent on the spin-off behavior of the command.
This error mechanism must be defined explicitly.

SDF-P Ferror handling

Command processing resumes with the *next* IF-BLOCK-ERROR command; only the current command block, or in the case of nested blocks also the higher-level command blocks, is/are searched in the direction of the end of the file. If no IF-BLOCK-ERROR block is found, the procedure terminates when the end of the procedure file is reached. The error is passed on to the caller. The error situation is terminated when an IF-BLOCK-ERROR block or procedure level 0 is reached.

For compatibility reasons the error situation is also terminated when a SET-JOB-STEP command is encountered.

Note

> Error handling can be performed for the command currently being terminated with the aid of the IF-CMD-ERROR command (contained in the chargeable SDF-P subsystem).

In *non-S procedures*, error handling continues to be controlled by the spin-off (see error handling using the SET-JOB-STEP command).

Format

IF-BLOCK-ERROR

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
	1	CMD0202	Syntax error
	1	SDP0118	Command in incorrect context
	1	SDP0223	Incorrect environment
	3	CMD2203	Incorrect syntax file
	32	CMD0221	System error (internal error)
	130	SDP0099	No further address space available

Example

Error recovery in an S procedure is illustrated using the *PROC.SORT2* procedure.

Contents of procedure PROC.SORT2:

```
/BEG-PAR-DECL
/    DECL-PAR   INPUT-FILE-1  (INIT = *PROMPT)
/    DECL-PAR   OUTPUT-FILE-1 (INIT = *PROMPT)
/END-PAR-DECL
/INPUT-FILE-EXIST:   SHOW-FILE-ATTRIBUTES  &(INPUT-FILE-1)
/COND-1:    IF  ( INPUT-FILE-1 = OUTPUT-FILE-1 )
/               WRITE-TEXT 'Input file = output file !!!'
/               WRITE-TEXT 'Output file with suffix COPY.<tsn>'
/               SET-VAR  OUTPUT-FILE-1 = '&(OUTPUT-FILE-1).COPY.&(TSN())'
/               CREATE-FILE &(OUTPUT-FILE-1)
/ERR-1:         IF-BLOCK-ERROR
/                  WRITE-TEXT 'New output file not possible'
/                  HELP-MSG  &(MC)
/                  EXIT-PROC  ERROR=*YES
/               ELSE
/                  WRITE-TEXT 'New output file &(OUTPUT-FILE-1) generated'
/ERR-1-END:     END-IF
/           ELSE   "Input file unequal output file"
/               SHOW-FILE-ATTRIBUTES  &(OUTPUT-FILE-1)
/ERR-2:         IF-BLOCK-ERROR
/                  WRITE-TEXT 'Output file being regenerated'
/                  CREATE-FILE &(OUTPUT-FILE-1)
/               ELSE
/                  WRITE-TEXT 'Output file already exists. Cancel'
/                  EXIT-PROC  ERROR=*YES
/ERR-2-END:     END-IF
/COND-1-END: END-IF
/WORK-1:         "Sorting file 1"
/           ADD-FILE-LINK  LINK=SORTIN, -
/                          FILE-NAME= &(INPUT-FILE-1)
/           ADD-FILE-LINK  LINK=SORTOUT, -
/                          FILE-NAME= &(OUTPUT-FILE-1)
/           SORT-FILE
/WORK-1-ERROR:  IF-BLOCK-ERROR
/                  WRITE-TEXT 'Input file does not exist or'
/                  WRITE-TEXT 'Error in WORK-1:  SC1 = &(SC1)'
/                  HELP-MSG  &(MC)
/                ELSE
/                  WRITE-TEXT 'WORK-1 terminated without errors'
/                END-IF
```

Case 1:

```
/call-proc alf.proc.sort2,log=*yes
%          1  1 /BEG-PAR-DECL
%          2  1 /DECL-PAR   INPUT-FILE-1  (INIT = *PROMPT)
%          3  1 /DECL-PAR   OUTPUT-FILE-1 (INIT = *PROMPT)
%          4  1 /END-PAR-DECL
%INPUT-FILE-1: abk.v110
%          5  1 /INPUT-FILE-EXIST:
%          5  1 /   SHOW-FILE-ATTRIBUTES  ABK.V110
%        99 :2OSG:$USER1.ABK.V110
%:2OSG: PUBLIC:      1 FILE  RES=        99 FRE=         1 REL=         0 PAGES
%          6  1 /COND-1:
%          6  1 /    IF  ( INPUT-FILE-1 = OUTPUT-FILE-1 )
%OUTPUT-FILE-1: abk.v110
%          7  1 /WRITE-TEXT 'Input file = output file !!!'
Input file = output file !!!
%          8  1 /WRITE-TEXT 'Output file with suffix COPY.<tsn>'
Output file with suffix COPY.<tsn>
%          9  1 /SET-VAR  OUTPUT-FILE-1 = 'ABK.V110.COPY.3ZAA'
%         10  1 /CREATE-FILE ABK.V110.COPY.3ZAA
%         11  1 /ERR-1:
%         11  1 /         IF-BLOCK-ERROR
%         15  1 /ELSE
%         16  1 /WRITE-TEXT 'New output file ABK.V110.COPY.3ZAA generated'
New output file ABK.V110.COPY.3ZAA generated
%         17  1 /ERR-1-END:
%         17  1 /     END-IF
%         27  1 /COND-1-END:
%         27  1 / END-IF
%         28  1 /WORK-1:
%         29  1 /ADD-FILE-LINK  LINK=SORTIN,                          FILE-NAME
= ABK.V110
%         31  1 /ADD-FILE-LINK  LINK=SORTOUT,                          FILE-NAM
E= ABK.V110.COPY.3ZAA
%         33  1 /SORT-FILE
%  BLS0523 ELEMENT 'SRT80', VERSION '079', TYPE 'L' FROM LIBRARY ':1OSH:$TSOS.SY
SLNK.SORT.079' IN PROCESS
%  BLS0524 LLM 'SRT80', VERSION '079' OF '2009-03-09 15:55:15' LOADED
%  BLS0551 COPYRIGHT (C) FUJITSU TECHNOLOGY SOLUTIONS 2009. ALL RIGHTS RESERVED
%  SRT1001  2012-04-28/17:59:44/000000.00 SORT/MERGE STARTED, VERSION 07.9C00/BS
2000V19.0
%  SRT1130  PLEASE ENTER SORT STATEMENTS
%  SRT1016  SORT/MERGE INPUT RECORDS:...........................2.924 (FROM 01)
%  SRT1030  SORT/MERGE OUTPUT RECORDS:..........................2.924
%  SRT1002 2012-04-28/17:59:45/000000.43 SORT/MERGE COMPLETED
%         34  1 /WORK-1-ERROR:
%         34  1 /  IF-BLOCK-ERROR
%         38  1 /ELSE
%         39  1 /WRITE-TEXT 'WORK-1 terminated without errors!'
WORK-1 terminated without errors!
%         40  1 /END-IF
%             1 /EXIT-PROCEDURE ERROR=*NO
```

A new output file is generated because the input file was also specified as the output file. Processing is error-free, i.e. when an error handling block is reached, the ELSE branch is processed.

Case 2:

```
/call-proc proc.sort2,log=*yes
%          1  1 /BEG-PAR-DECL
%          2  1 /DECL-PAR   INPUT-FILE-1  (INIT = *PROMPT)
%          3  1 /DECL-PAR   OUTPUT-FILE-1 (INIT = *PROMPT)
%          4  1 /END-PAR-DECL
%INPUT-FILE-1: abk.v111
%          5  1 /INPUT-FILE-EXIST:
%          5  1 /   SHOW-FILE-ATTRIBUTES  ABK.V111
%  DMS0533 REQUESTED FILE NOT CATALOGED IN PUBSET '20SG'. COMMAND TERMINATED
%  SDP0004 ERROR DETECTED AT COMMAND LINE:           5 IN PROCEDURE ':20SG:$USER
1.ALF.PROC.SORT2'
%         34  1 /WORK-1-ERROR:
%         34  1 /  IF-BLOCK-ERROR
%         35  1 /WRITE-TEXT 'Input file does not exist or'
Input file does not exist or
%         36  1 /WRITE-TEXT 'Error in WORK-1:  SC1 = 64'
Error in WORK-1:  SC1 = 64
%         37  1 /HELP-MSG  DMS0533
%  DMS0533 REQUESTED FILE NOT CATALOGED IN PUBSET '(&00)'. COMMAND TERMINATED
%  ? This message is issued by DMS commands. The requested file is not
%    cataloged in the requested pubset.
%  RESPONSE : NONE
%         40  1 /END-IF
%             1 /EXIT-PROCEDURE ERROR=*NO
```

The input file *ABK.V111* does not exist and leads to an error upon SHOW-FILE-ATTRIBUTES. A branch is made to the next error handling block, in this case the command block initiated with the tag *WORK-1-ERROR*. The error handling blocks in the subordinate IF block (IF and ELSE branches) are **not** taken into account. Since the error was “handled” the procedure is implicitly terminated with ERROR=*NO.

Case 3:

```
/call-proc proc.sort2,log=*yes
%          1  1 /BEG-PAR-DECL
%          2  1 /DECL-PAR   INPUT-FILE-1  (INIT = *PROMPT)
%          3  1 /DECL-PAR   OUTPUT-FILE-1 (INIT = *PROMPT)
%          4  1 /END-PAR-DECL
%INPUT-FILE-1: abk.v110
%          5  1 /INPUT-FILE-EXIST:
%          5  1 /   SHOW-FILE-ATTRIBUTES  ABK.V110
%        99 :20SG:$USER1.ABK.V110
%:20SG: PUBLIC:      1 FILE  RES=        99 FRE=         1 REL=         0 PAGES
%          6  1 /COND-1:
%          6  1 /    IF  ( INPUT-FILE-1 = OUTPUT-FILE-1 )
%OUTPUT-FILE-1: abk.v110.sort
%         18  1 /ELSE  "Input file unequal output file"
%         19  1 /SHOW-FILE-ATTRIBUTES  ABK.V110.SORT
%       138 :20SG:$USER1.ABK.V110.SORT
%:20SG: PUBLIC:      1 FILE  RES=       138 FRE=        40 REL=        39 PAGES
%         20  1 /ERR-2:
%         20  1 /         IF-BLOCK-ERROR
%         23  1 /ELSE
%         24  1 /WRITE-TEXT 'Output file already exists. Cancel'
Output file already exists. Cancel
%         25  1 /EXIT-PROC  ERROR=*YES
```

The output file exists already. The SHOW-FILE-ATTRIBUTES command for file *ABK.V110.SORT* does not lead to an error. Consequently, the ELSE branch in the subsequent error handling block (at *ERR-2*) is processed (explicit EXIT-PROCEDURE with ERROR=*YES).

IMPORT-FILE

Create catalog entry for files (file import)

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE FILE-GENERATION-GROUP
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT TSOS
Routing code:	$ (bei NBCONOPI=N) bzw. E (bei NBCONOPI=Y)

Function

The IMPORT-FILE command creates the catalog entries for files which are contained on private volumes or Net-Storage volumes (i.e. imports them). The catalog entries for files held on private volumes or Net-Storage volumes can be deleted (i.e. they can be exported) using the EXPORT-FILE command. If there is already a catalog entry for a file on private disk Net-Storage volume, it can be updated on the basis of the F1 label of the private disk or the file catalog of the Net-Storage volume (SUPPORT=*BY-FILENAME(...)).

When importing file generation groups with generations stored on different disks, it should be noted that generations will be cataloged only if the group entry already exists in the system catalog or is kept on the first disk to be imported. Otherwise, the catalog entries for the generations imported before the group entry will be missing. These generations must then be cataloged afterwards using an IMPORT-FILE command.

Privileged functions

For catalog entries on private volumes or Net-Storage volumes systems support (TSOS privilege) can use the the NEW-USER operand to specify a user ID under which the files are to be cataloged.

i If the IMPORT-FILE command with SUPPORT=*DISK is entered without a file name being explicitly specified (this corresponds to FILE-NAME=*ALL), unlike with nonprivileged users the entire F1 label of the private disk or the entire catalog of the Net-Storage volume is imported.

By default, systems support (TSOS privilege) is a co-owner of all the files (and can therefore create files under all user logons). When SECOS is used, this co-ownership can be restricted for permanent files.

Systems support can generate files under any user ID (TSOS privilege).In conjunction with the SECOS software product a user can allow other user IDs to act as co-owners. Co-owners of a user ID are then also allowed to import files under that ID.

Notes

- The functions of the commands IMPORT-FILE and EXPORT-FILE are not exact opposites: When a volume is exported, DMS deletes the catalog entries for all files which occupy storage space on this volume.
- If the same volume is re-imported, DMS creates catalog entries only for those files which begin on the volume (i.e. files which were given space on this volume by their primary allocation).

The IMPORT-FILE command can be used, for example, to catalog files which were created under the user ID of the calling job and which are contained on private disks or Net-Storage volumes. DMS copies the file attributes from the F1 label of the private disk or the catalog of the Net-Storage volume into the catalog entry. Partially qualified file names can be processed, so that one command can be used to import a number of files.

- Locked entries can be imported from the F1 label or from the catalog of the Net-Storage volume. However, if entries in the user catalog have to be replaced (REPLACE-OLD-FILES=*YES/*ABS), the entries must not be locked and write access must be permitted.
- When importing or exporting file generation groups which are stored on private disks, the following must be noted:
  - An IMPORT-FILE command will only catalog generations for which there is a group entry, either on the designated disk or already present in the system catalog. For a file generation group which is split across several disks and is not yet cataloged, this has the following consequences: If the first disk which is imported is not the one containing the group entry, and this latter is only subsequently imported, then the catalog entries for the generations held on the first disk will be missing. Remedy: issue an IMPORT-FILE command for each of the volumes affected, or for the uncataloged generations.
  - If a private disk only holds individual generations from a file generation group, but not the group entry, then when this volume is exported, “gaps” can result in the file generation group (i.e. if the VOLUME operand specifies a private disk which contains the file generation but not its group entry). This gap affects only the catalog entry, the file generation group remains complete on the disk.

Format

```
IMPORT-FILE

SUPPORT = *DISK(...) / *TAPE(...) / *BY-FILE-NAME(...)
   *DISK(...)
      | VOLUME = <alphanum-name 1..6>
      |,DEVICE-TYPE = *BY-VOLUME-CATALOG / <device>
      |,FILE-NAME = *ALL / <partial-filename 2..50 without-cat-id> / <filename 1..51 without-cat-id>
      |,TYPE-OF-FILE = [ *ANY ] (...)
      |    [ *ANY ](...)
      |       | GENERATIONS = *YES / *NO
      |,REPLACE-OLD-FILES = *NO / *YES / *ABS
      |,OUTPUT = *SYSLST / *SYSOUT / *ALL / *NO
      |,NEW-USER = *NONE / <name 1..8>
      |,PUBSET = *STD / <cat-id 1..4>
   *TAPE(...)
      | VOLUME = *ANY / list-poss(255): <alphanum-name 1..6>
      |,DEVICE-TYPE = *BY-VOLUME-CATALOG / <device>
      |,FILE-NAME = <filename 1..54>
      |,PREMOUNT-LIST = *NONE / list-poss(255): <integer 0..255>
   *BY-FILE-NAME(...)
      | FILE-NAME = <filename 1..54 without-vers>
      |,INFORMATION = *NAME-AND-SPACE / *SPACE-SUMMARY / *ALL-ATTRIBUTES /
      |                 [*PARAMETERS](...) / *STATISTICS / *MINIMUM
      |    [*PARAMETERS](...)
      |       | STANDARD = *NO / *YES
      |       |,PROTECTION = *NO / *YES
      |       |,FILE = *NO / *YES / *FILE
      |       |,PASSWORDS = *NO / *YES
      |       |,ALLOCATION = *NO / *YES
      |       |,BACKUP = *NO / *YES
      |       |,HISTORY = *NO / *YES
      |       |,ORGANIZATION = *NO / *YES
      |       |,SECURITY = *NO / *YES
```

(Part 1 of 2)

,**OUTPUT** = ***SYSOUT** / ***NONE** / ***SYSLST**(...) / ***PRINT**ER(...) /
<filename 1..54 without-gen-vers>(...)

***SYSLST**(...)

FORM-NAME = ***STD** / ***F**IL**E**-**NAME**

***PRINT**ER(...)

FORM-NAME = ***STD** / ***F**IL**E**-**NAME**

<filename>(...)

FORM-NAME = ***STD** / ***F**IL**E**-**NAME**

(Part 2 of 2)

Operands

SUPPORT = *DISK(...) / *TAPE(...) / *BY-FILE-NAME(...)
Specifies whether the files are to be imported from disk or from tape, or whether the catalog entries of the files on private disk or Net-Storage volumes are to be updated using the catalog entry.

SUPPORT = *DISK(...)
The files to be imported are stored on private disk or on a Net-Storage volume.

VOLUME = <alphanum-name 1..6>
Volume serial number (VSN) of the volume on which the files to be imported are stored.

DEVICE-TYPE = *BY-VOLUME-CATALOG / <device>
Identifies the type of device on which the volume is to be mounted.
Only device types known within the system are accepted. In interactive mode, the possible device types are displayed with DEVICE-TYPE=?.
The volume type NETSTOR must be specified for Net-Storage volumes.
Every specification of a disk device type known in the system is handled like the STDDISK specification.
The permissible specifications for DEVICE-TYPE can also be found in the device table in section “Device type table” on page 1-82 (device type column)). The devices available on the system can be listed with the SHOW-DEVICE-CONFIGURATION command.

DEVICE-TYPE = *BY-VOLUME-CATALOG
The device type is determined via the MAREN device substitution function if the MAREN subsystem is available.

FILE-NAME = *ALL / <partial-filename 2..50 without-cat-id> / <filename 1..51 without-cat-id>
Specifies the files, file generation groups or file generations which are to be cataloged. When the default FILE-NAME=*ALL is specified, DMS catalogs all files etc. which are stored under the user ID of the current job on the volume specified in the VOLUME operand. For privileged users, see "Privileged functions" on page 3-422.

i For file generations/file generation groups, create the group entry before cataloging the generations!

Only files contained on the volume specified in VOLUME will be imported.

FILE-NAME = *ALL
All the files contained on the volume specified in VOLUME will be imported.

TYPE-OF-FILE = *ANY(...)
Specifies whether the file generations from the specified file generation group are to be imported also. It is only meaningful to use this operand if no file generation is to be imported (GENERATIONS=*NO).

GENERATIONS = *YES / *NO
For file generation groups GENERATIONS specifies whether only the group entry is to be cataloged, or the file generations stored on the same private disk are also to be cataloged.

GENERATIONS = *YES
If the group entry is on the private disk, DMS will catalog the FGG and all its generations which begin on this disk. If there is no group entry on the disk or in the user catalog, no file generations will be cataloged.

GENERATIONS = *NO
DMS transfers just the group entry for the FGG to the catalog.

REPLACE-OLD-FILES = *NO / *YES / *ABS
Specifies whether an existing "old" catalog entry is to be overwritten.

REPLACE-OLD-FILES = *NO
DMS does not overwrite the existing catalog entry.

REPLACE-OLD-FILES = *YES
The old catalog entry will be deleted if it does not match the details specified in the IMPORT-FILE command.

The following situations should be distinguished:

- The cataloged file is stored on a public disk: the catalog entry is overwritten, which means that the public file is deleted (providing the protection attributes permit this and the file is not locked; otherwise, the old catalog entry is left unchanged).
- The cataloged file is stored on a Net-Storage volume. A file with the same name on the same Net-Storage volume is not imported and the catalog entry is not deleted.
- The cataloged file is stored on a Net-Storage volume. A file on private disk, on a different Net-Storage volume or on the same Net-Storage volume but with a different name (not a node file) is imported. The catalog entry is overwritten and the file therefore deleted (providing the protection attributes permit this and the file is not locked; otherwise the old entry remains unchanged). In this case deleting the file means:
  - A BS2000 file on Net-Storage is also deleted on the Net-Storage volume.
  - A node file on the Net-Storage volume is retained.

  i In contrast to BS2000 files, node files cannot be imported to another user ID as the owner of a node file may not be changed.
- The cataloged file is on private disk, but begins on a volume other than the one specified in the VOLUME operand: the catalog entry is overwritten (exported) (providing the protection attributes permit this and the file is not locked; otherwise, the old entry remains unchanged).
- The cataloged file is on private disk, and begins on the disk specified in the VOLUME operand: The catalog entry will *not* be deleted (exported). A file with the same name will *not* be imported.

REPLACE-OLD-FILES = *ABS
The old catalog entry is overwritten even if the catalog entry and the specifications in the IMPORT-FILE command match. Special return codes show whether the entry was overwritten (special return code 8) or whether a file lock prevented overwriting (special return code 3).

The following situations should be distinguished:

- The cataloged file is stored on a public disk: the catalog entry is overwritten, which means that the public file is deleted (providing the protection attributes permit this and the file is not locked; otherwise, the old catalog entry is left unchanged).
- The cataloged file is stored on a Net-Storage volume. Difference to the operand value *YES: A file with the same name on the Net-Storage volume is imported. The catalog entry is overwritten. If a file lock or the protection attributes prevent this, the old entry is retained.

- The cataloged file is stored on a Net-Storage volume. A file on private disk, on another Net-Storage volume or with a different name on the same Net-Storage volume (not a node file) is imported: the catalog entry is overwritten and the file is consequently deleted. If a file lock or the protection attributes prevent this, the old entry is retained.

  i In contrast to BS2000 files, node files cannot be imported to another user ID as the owner of a node file may not be changed.
- The cataloged file is on private disk, but begins on a volume other than the one specified in the VOLUME operand: the catalog entry is overwritten (providing the protection attributes permit this and the file is not locked; otherwise, the old entry remains unchanged).
- The cataloged file is on private disk, and begins on the disk specified in the VOLUME operand. Difference to the operand value *YES:
  The catalog entry is overwritten (providing the protection attributes permit this and the file is not locked; otherwise the old catalog entry remains unchanged). Even if there is a file with the same name, the file will be imported.

OUTPUT = *SYSLST / *SYSOUT / *ALL / *NO
Specifies how the execution of the command is to be logged.

OUTPUT = *SYSLST
The execution of commands will be logged on SYSLST; this log contains the special return codes and message texts (see table “Special return codes and messages for SUPPORT=*DISK” on page 3-436).

OUTPUT = *SYSOUT
The execution of commands will be logged on SYSOUT; only the special return codes and file names are output in this log (see table “Special return codes and messages for SUPPORT=*DISK” on page 3-436).

OUTPUT = *ALL
The execution of commands will be logged on SYSLST and SYSOUT (see the OUTPUT=*SYSLST and OUTPUT=*SYSOUT operands).

OUTPUT = *NO
No return information on the execution of the command will be output.

NEW-USER = *NONE / <name 1..8>
User ID under which the file is to be cataloged.
A new user ID is assigned to a file on a private disk both in the file catalog and in the F1 label of the disk. The new user ID is assigned to a BS2000 file on a Net-Storage volume both in the file catalog and in the catalog of the Net-Storage volume. A node file, on the other hand, cannot be cataloged under the new user ID as the owner of a node file may not be changed. In this case the import is rejected with return code D.

PUBSET = *STD / <cat-id 1..4>
Specifies the pubset on which the files are to be cataloged. If none is specified, the catalog entries are created in the file catalog of the default pubset for the user's ID.

If a Net-Storage volume is specified in the VOLUME operand, the Net-Storage volume must be assigned to the pubset specified here. Only in this case is the file catalog of the specified pubset updated on the basis of the catalog entries of the Net-Storage volume.

PUBSET = *STD
The default setting is *STD, i.e. the catalog entries are set up under the file catalog of the default pubset of the user ID (see the output field DEFAULT-PUBSET in the SHOW-USER-ATTRIBUTES command).

PUBSET = <cat-id 1..4>
Specifies the pubset on which the files are to be cataloged.

SUPPORT = *TAPE(...)
Specifies that the file to be imported is located on a tape or tape cartridge.

VOLUME = list-poss(255): <alphanum-name 1..6> / *ANY
The VSN of the tape on which the file to be imported is stored.

VOLUME = *ANY
The volume is determined via the MAREN volume substitution function if the MAREN subsystem is available.

DEVICE-TYPE = *BY-VOLUME-CATALOG / <device>
The device type to which the required tape is assigned. Only device types or volume types known within the system are accepted. In interactive mode, the possible device and volume types are displayed with DEVICE-TYPE=?.

DEVICE-TYPE = *BY-VOLUME-CATALOG
The device type is determined via the MAREN device substitution function if the MAREN subsystem is available.

FILE-NAME = <filename 1..54>
The file will only be imported if it is held on the specified tape.

PREMOUNT-LIST = *NONE / list-poss(255): <integer 0..255>
The tape sequence number of the tape to be mounted.

PREMOUNT-LIST = *NONE
The first of the volumes specified in the VOLUME operand will be requested.

PREMOUNT-LIST = list-poss(255): <integer 0..255>
Issues a MOUNT message on the console requesting the mounting of tapes or suppresses the request. The tape sequence numbers specified here relate to the VSNs listed in the VOLUME operand. Specifying PREMOUNT-LIST=0 has the effect that no tapes will be requested. Otherwise use: PREMOUNT-LIST=(n[,n+1][,n+2]...) with n=1.

SUPPORT = *BY-FILE-NAME(...)
Specifies for which files on private disk or Net-Storage volumes the catalog entry in the TSOSCAT is to be updated on the basis of the entry on the volume (F1 label or catalog of the Net-Storage volume).

FILE-NAME = <filename 1..54 without-vers>
The name of the file whose catalog attributes are to be copied from the F1 label of a private disk or the catalog of the Net-Storage volume. If the entry in the F1 label of the private disk or the catalog of the Net-Storage volume no longer exists, the catalog entry is deleted.

Only the catalog entry is output for a file on public disk.

INFORMATION = <u>*NAME-AND-SPACE</u> / *SPACE-SUMMARY / *ALL-ATTRIBUTES / *PARAMETERS(...) / *STATISTICS / *MINIMUM
Type of information to be displayed.

INFORMATION = <u>*NAME-AND-SPACE</u>
For the selected files, the number of PAM pages used will be output with the file name, in alphabetical order of file name. This is the only possible operand value if the output is **not** routed to SYSOUT (see the OUTPUT operand).

INFORMATION = *SPACE-SUMMARY
Provides information on the storage space allocation of the files and file generations identified in the current command.

INFORMATION = *ALL-ATTRIBUTES
Returns all the information that is contained in the catalog (NAME-AND-SPACE, PARAMETERS) for the selected files. The output fields are listed in alphabetical order with a further explanation of their possible values in the table “Meaning of the output fields” at the end of the operand description section.

INFORMATION = *PARAMETERS(...)
Returns only the selected information for the files in question. The information from the catalog is logically arranged into information blocks. Only those information blocks which are explicitly selected by the user are output. Totals lines with the overall storage space allocation for each selected file are shown at the end of the returned information (see INFORMATION=SPACE-SUMMARY).

STANDARD = <u>*NO</u> / *YES
Specifies whether standard information is to be output.

i The STANDARD operand is only supported for compatibility reasons. It is not offered in guided dialog. STANDARD=*YES is equivalent to `INFORMATION=*PAR(ORGANIZATION=*YES, ALLOCATION=*YES)`.

PROTECTION = *NO / *YES
Specifies whether information concerning the file protection and expiration date is to be output.

i The PROTECTION operand is only supported for compatibility reasons. It is not offered in guided dialog. PROTECTION=*YES is equivalent to `INFORMATION=*PAR(ORGANIZATION=*YES,SECURITY=*YES, BACKUP=*YES)`.

FILE = *NO / *YES
Output: storage space used in PAM pages, and file attributes.

i The FILE operand is only supported for compatibility reasons. It is not offered in guided dialog. FILE=*YES is equivalent to `INFORMATION=*PAR(ORGANIZATION=*YES,ALLOCATION=*YES)`.

PASSWORDS = *NO / *YES
Specifies whether or not information on passwords is to be output. A header line with the title “PASSWORDS” and a line with the information on file passwords (which corresponds to the first information line of the security block) are output.

PASSWORDS = *NO
No PASSWORD information is to be output.

PASSWORDS = *YES
The first line of the security block is output.
This line provides information on the types of password which protect a file or file generation group. The passwords themselves are not shown. If a user forgets a file password, only the system administrator can help him. The following attributes are output:

Header line with the word “PASSWORDS”

EXEC-PASS	protection with execution password
READ-PASS	protection with read password
WRITE-PASS	protection with write password

ALLOCATION = *NO / *YES
Outputs the allocation block for the selected files, i.e. all file attributes that affect the allocation of storage space. The allocation block contains the following attributes:

Header line with the word “ALLOCATION”

DEVICE-TYPE	device type for volume
EXTENTS	total number of extents for the file
HIGH-US-PA	highest used PAM page
NUM-OF-EXT	Number of extents
S-ALLOC	secondary allocation for file extension
SUPPORT	type of volume
VOLUME	volume serial number

BACKUP = *NO / *YES

Outputs the backup block for the selected files, i.e. all file attributes that affect backups. The backup block contains the following attributes:

Header line with the word “BACKUP”

BACK-CLASS	backup level for ARCHIVE or HSMS
MIGRATE	indicates whether the file may be migrated
SAVED-PAG	indicates whether the file must always be fully saved
VERSION	internal attribute for incremental saving with the ARCHIVE or HSMS utility routine.

HISTORY = *NO / *YES

Outputs the history block for the selected files, i.e. all file attributes related to the file history. The history block contains the following attributes:

Header line with the word “HISTORY”

ACC-COUNT	access counter
ACC-DATE	date of last access
ACC-TIME	time of last access
CHANG-DATE	date the file was last changed (written to)
CHANG-TIME	time the file was last changed (written to)
CRE-DATE	creation date
CRE-TIME	time of creation
S-ALLO-NUM	number of secondary allocations

ORGANIZATION = *NO / *YES

Outputs the organization block for the selected files, i.e. all file attributes related to the structure of the file. The organization block contains the following attributes:

AVAIL	availability
BLK-CONTR	Block control information
BUF-LEN	block type (standard or nonstandard block)
COD-CH-SET	coded character set (CCS) with XHCS support
FILE-STRUC	access method when the file was created
KEY-LEN	Length of ISAM key
KEY-POS	position of ISAM key
LOG-FL-LEN	length of logical ISAM flag
PROPA-VAL	propagation of ISAM flag
REC-FORM	Record format
REC-SIZE	Record length
VAL-FL-LEN	length of ISAM value flag

The information block for file generation groups begins with the header line `GENERATION-INFO` and contains the following attributes:

BASE-NUM	base value for relative generation numbers
DEVICE-TYPE	device type for volume
FIRST-GEN	most recent or last cataloged file generation

LAST-GEN	oldest existing file generation
MAXIMUM	maximum number of simultaneously cataloged generations
OVERFL-OPT	overflow option when maximum number is reached
VOLUME	volume serial number

SECURITY = <u>*NO</u> / *YES

Outputs the security block for the selected files, i.e. all file attributes related to the file security. The security block contains the following attributes:

Header line with the word “SECURITY”

ACCESS	type of access (standard access control)
ACL	file protection with ACL (only supported for reasons of compatibility)
AUDIT	file monitoring
DESTROY	data automatically destroyed on deletion
ENCRYPTION	encryption method
EXEC-PASS	protection with execution password
EXPIR-DATE	date on which the file may be changed again
EXPIR-TIME	time relative to *EXPIR-DATE*
FREE-DEL-D	free-for-deletion date
FREE-DEL-T	time relative to free-for-deletion date
GUARD-READ	read protection using a guard
GUARD-WRIT	write protection using a guard
GUARD-EXEC	protection with execute guard
GROUP	access rights of user class “group” (BASIC-ACL)
OTHERS	access rights of user class “others” (BASIC-ACL)
OWNER	access rights of file owner (BASIC-ACL)
READ-PASS	protection with read password
SP-REL-LOCK	protection against release of storage space
USER-ACC	shareability attribute (standard access control)
WRITE-PASS	protection with write password

INFORMATION = *STATISTICS

Returns information on storage space allocation for all selected files. The storage space allocations for files and file generation groups are output separately and include additional information for each type of volume (public disks, private disks, and tapes). Files for which no entry has been made in the volume list are indicated in the output for public disks (NO VSN).

INFORMATION = *MINIMUM

Outputs a line of information with the most important file attributes in summarized form for each selected file. This line contains brief information on the FILE-STRUCTURE, passwords, highest activated access protection level, the number of reserved PAM pages, and the file name. The brief information is indicated as follows:

– the access method with which the file was created: indicated by the first letter in the value of FILE-STRUCTURE;

– information for read, write, and execute passwords: shown in each case with “Y” or “N” to indicate whether or not the corresponding password was assigned;

- information on the highest activated access control method: see the PROTECTION-ACTIVE operand.
  - When standard access control is the highest protection:
    - USER-ACCESS with “Y” for ALL-USERS or SPECIAL, or with “N” for OWNER-ONLY
    - ACCESS with “W” for write access or “R” for read access only.
  - When the basic access control list (BASIC-ACL) is the highest protection: the access rights assigned for each of the user groups OWNER, GROUP and OTHERS are shown (“RWX” or “-” for an access right that has not been set).
  - When access control using GUARDS is the highest protection: the value “GUARDS” and the protected access mode (“RWX” or “-” for an access mode not granted) are shown.

OUTPUT = <u>*SYSOUT</u> / *NONE / *SYSLST(...) / *PRINTER(...) / <filename 1..54 without-gen-vers>(...)
Defines the destination to which the information is output.

OUTPUT = *NONE
Suppresses the output of information. This specification is typically useful when only the command return code is to be evaluated in a procedure.

OUTPUT = <u>*SYSOUT</u>
The information will be presented in tabular form on SYSOUT.

OUTPUT = *SYSLST(...)
The information will be presented in tabular form on SYSLST. SYSLST will automatically be output on a printer at the end of the job. The format of the output will be determined as follows:

> **FORM-NAME = <u>*STD</u> / *FILE-NAME**
> The default value is *STD, i.e. the information will be presented in tabular form. If *FILE-NAME is specified, then a simple list of file names will be output.

OUTPUT = *PRINTER(...)
The information will be presented in tabular format and output to printer (as an immediate SPOOL job, which will be output with CONTROL-CHARACTERS= *EBCDIC).

> **FORM-NAME = <u>*STD</u> / *FILE-NAME**
> The default value is *STD, i.e. the information will be presented in tabular form. If *FILE-NAME is specified, then a simple list of file names will be output.

OUTPUT = <filename 1..54 without-gen-vers>(...)
The information will be written to the specified file; the format of the output can be determined as follows.

FORM-NAME = *STD / *FILE-NAME
The default value is *STD, i.e. the information will be presented in tabular form. If *FILE-NAME is specified, then a simple list of file names will be output.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed
2	0	DMS051E	Inconsistency in private disk pool or operand error detected
2	0	DMS054A	Insufficient disk space or access to disk not possible
2	0	DMS05B6	Invalid time conversion from UTC to LT
	1	CMD0202	Syntax or semantic error in command
	32	DMS0584	A state that does not allow the function to continue was reported during processing
	32	DMS05C7	Unexpected internal error in DMS
	64	CMD0216	Privilege errors
	64	DMS0501	Requested catalog not available
	64	DMS0512	Requested catalog not found
	64	DMS051B	Requested user ID not in pubset Guaranteed message: DMS051B
	64	DMS051C	User not authorized to access pubset Guaranteed message: DMS051C
	64	DMS0533	Requested file not cataloged in pubset Guaranteed message: DMS0533
	64	DMS0535	Specified file not shareable
	64	DMS053E	File already cataloged on private volume
	64	DMS0555	Specified file name already cataloged
	64	DMS055C	Catalog entry not found on the specified private disk or Net-Storage volume
	64	DMS0585	Error detected when processing catalog or multiprocessor system
	64	DMS0586	It is not possible to access or reserve a volume at present
	64	DMS0587	Use of the specified command has been restricted by the system administrator
	64	DMS05F8	DMS error reported
	64	DMS05FC	Specified user ID not in HOME pubset
	64	DMS05FE	Requested file(s) not found
	64	DMS06C4	File generation group not yet cataloged
	64	DMS06CC	No file corresponding to specified operands
	64	DMS06D0	Specified file generation does not exist
	64	DMS06FF	BCAM connection severed
	128	DMS0506	Function not executed due to change in master
	130	DMS0524	System address space exhausted

(Part 1 of 2)

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	130	DMS053C	Insufficient space for catalog file on pubset
	130	DMS0582	File is currently locked or being used and cannot be processed
	130	DMS0585	Error detected when processing catalog or multiprocessor system Guaranteed message: DMS053C
	130	DMS0586	It is not possible to access or reserve a volume at present
	130	DMS0594	Not enough virtual memory available
	130	DMS05C8	Maximum permitted number of files reached

(Part 2 of 2)

Special return codes and messages for SUPPORT=*DISK

The following codes (SYSOUT) or messages (SYSLST) can be output :

Code on SYSOUT	Message on SYSLST	Meaning
0	FILE DID NOT EXIST	There was no file with the same name and a new catalog entry has been created.
1	FILE HAS BEEN ERASED	A file of the same name already existed and has been overwritten.
2	FILE EXISTS / REPLACE-OLD-FILES =*NO	A file of the same name existed but was not overwritten; the REPLACE-OLD-FILES operand had the value *NO.
3	FILE IS PROTECTED (ERASE ERROR OR FILE IS IN USE)	A file of the same name existed and could not be deleted because of active protection functions (ACCESS=READ, password protection etc.) or the file is locked because it is being processed.
4	ERROR ON CATALOG ACCESS	System error during catalog access.
5	FILE ALREADY ON PRIVATE	The file is already cataloged and is stored on the volume specified in the VOLUME operand.
6	ERROR ON VTOC ACCESS	System error when accessing the F1 label of the private disk or the catalog of the Net-Storage volume.
7	GENERATION OUT OF RANGE	Invalid attempts to import a file generation: the absolute generation number of the generation to be imported conflicts with the limits defined in the group entry.
8	C.E. HAS BEEN REPLACED	A catalog entry already existed for the specified disk and has been replaced.

Table 55: IMPORT-FILE: Special return codes and messages (Part 1 of 2)

Code on SYSOUT	Message on SYSLST	Meaning
A	INVALID FILENAME	The path name for the file to be imported (including pvsid and userid) is more than 54 characters long. Pathname = :catid:$.userid:filename It should be noted that the pathname is made up from details taken from two *different* operands in the command. The file name is specified in the FILE-NAME operand, the catalog ID in the PUBLIC-VOLUME-SET operand. For a 1-character catalog ID, the file name must not be longer than 51 characters. For a 4-character catalog ID, the file name may be up to 48 characters long.
T	ERROR ON NET-STORAGE ACCESS	System error during Net-Storage access.
C	LARGE FILE NOT ALLOWED	The file to be imported is larger than 32 GB, but the pubset specified does not permit files larger than 32 GB.
D	RENAME OF NODE-FILES NOT ALLOWED	The importing of node files to a foreign user ID is not permitted.

Table 55: IMPORT-FILE: Special return codes and messages (Part 2 of 2)

Examples

Example 1: Importing files from private disks

```
/imp-file sup=disk(vol=work01,dev-type=d3435,file-name=*all,
                   output=*sysout)              ——————————————————————————————————   (1)
%  0 :2OS2:$USER1.MAX.LIB
%  0 :2OS2:$USER1.MAX.PRIV-FILE.1
%  0 :2OS2:$USER1.MAX.PRIV-FILE.2
%  0 :2OS2:$USER1.MAX.PRIV-FILE.3
/show-file-attr max.,select=(support=*priv)     ——————————————————————————————————   (2)
%       210*:2OS2:$USER1.MAX.LIB
%         3*:2OS2:$USER1.MAX.PRIV-FILE.1
%         3*:2OS2:$USER1.MAX.PRIV-FILE.2
%         3*:2OS2:$USER1.MAX.PRIV-FILE.3
%:2OS2: PRDISC:      4 FILES RES=       219 FRE=        29 REL=        21 PAGES
```

(1) All files of the user *USER1* on the private disk *WORK01* (device type *D3435*) are to be imported. The messages output during the process are to be sent to SYSOUT.

(2) The output indicates that four files were successfully imported (as is evident from the return code *0* in each case). The command causes the catalog entries for files which have names beginning with *MAX.* and which are stored on private disk to be shown (the files which were previously imported).

Example 2: Exporting and importing tape files

```
/show-file-attr support=*tape,inf=(alloc=*yes,org=*yes)     ——————————————————————   (1)
%           :2OS2:$USER1.MAX.TAPE-FILE.1
%  ------------------------------- ORGANIZATION -------------------------------
%  FILE-STRUC = SAM         BUF-LEN    = STD(1)      BLK-CONTR  = PAMKEY
%  REC-FORM   = (V,N)       REC-SIZE   = 2044
%  CODE       = EBCDIC      LABEL      = (STD,1)     FILE-SEQ   = 1
%  BLK-OFFSET = 4
%  ------------------------------- ALLOCATION   -------------------------------
%  SUPPORT    = PVT                                  BLK-COUNT =  1
%  EXTENTS     VOLUME     DEVICE-TYPE      EXTENTS     VOLUME     DEVICE-TYPE
%              D2326K      TAPE-C4
%         :2OS2:$USER1.MAX.TAPE-FILE.2
%  ------------------------------- ORGANIZATION -------------------------------
%  FILE-STRUC = SAM         BUF-LEN    = STD(1)      BLK-CONTR  = PAMKEY
%  REC-FORM   = (V,N)       REC-SIZE   = 2044
%  CODE       = EBCDIC      LABEL      = (STD,1)     FILE-SEQ   = 2
%  BLK-OFFSET = 4
%  ------------------------------- ALLOCATION   -------------------------------
%  SUPPORT    = PVT                                  BLK-COUNT =  1
%  EXTENTS     VOLUME     DEVICE-TYPE      EXTENTS     VOLUME     DEVICE-TYPE
%              D2326K      TAPE-C4
%:2OS2: TAPE  :      2 FILES
/export-file vol=d2326k,file=name(*all),output=*sysout   —————————————————————————   (2)
%  DMS0516 DELETE FILE(S) ':2OS2:$USER1.' ? REPLY (Y=YES; N=NO; T=TERMINATE COM
MAND; ?=EXPLAIN ADDITIONAL OPTIONS)?y                      ———————————————————————   (3)
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.TAPE-FILE.1' DELETED
%  DMS0800 SPECIFIED FILE ':2OS2:$USER1.MAX.TAPE-FILE.2' DELETED
/show-file-attr support=tape                               ———————————————————————   (4)
%  DMS06CC NO FILE CORRESPONDING TO SPECIFIED OPERANDS
/import-file sup=*tape(vol=d2326k,dev-type=tape-c4,
                       file-name=max.tape-file.1)          ———————————————————————   (5)
/import-file sup=*tape(vol=d2326k,dev-type=tape-c4,
                       file-name=max.tape-file.2)          ———————————————————————   (6)
/show-file-attr support=*tape,inf=(org=*yes,alloc=*yes)      —————————————————————   (7)
```

```
%          :2OS2:$USER1.MAX.TAPE-FILE.1
%  -------------------------------- ORGANIZATION --------------------------------
%  FILE-STRUC = NONE        BUF-LEN    = NONE        BLK-CONTR  = NONE
%  REC-FORM   = NONE        REC-SIZE   = 0
%  CODE       = NONE        LABEL      = NONE        FILE-SEQ   = NONE
%  BLK-OFFSET = 0
%  -------------------------------- ALLOCATION   --------------------------------
%  SUPPORT    = PVT                                  BLK-COUNT =  0
%  EXTENTS     VOLUME     DEVICE-TYPE      EXTENTS     VOLUME     DEVICE-TYPE
%  (           D2326K      TAPE-C4 )
%        :2OS2:$USER1.MAX.TAPE-FILE.2
%  -------------------------------- ORGANIZATION --------------------------------
%  FILE-STRUC = NONE        BUF-LEN    = NONE        BLK-CONTR  = NONE
%  REC-FORM   = NONE        REC-SIZE   = 0
%  CODE       = NONE        LABEL      = NONE        FILE-SEQ   = NONE
%  BLK-OFFSET = 0
%  -------------------------------- ALLOCATION   --------------------------------
%  SUPPORT    = PVT                                  BLK-COUNT =  0
%  EXTENTS     VOLUME     DEVICE-TYPE      EXTENTS     VOLUME     DEVICE-TYPE
%  (           D2326K      TAPE-C4 )
%:2OS2: TAPE   :      2 FILES

/copy-file from=max.tape-file.1,to=#copy.tape-1          -------------------------  (8)
/show-file-attr max.tape-file.1,inf=(org=*yes)            ------------------------  (9)
%          :2OS2:$USER1.MAX.TAPE-FILE.1
%  -------------------------------- ORGANIZATION --------------------------------
%  FILE-STRUC = SAM         BUF-LEN    = STD(1)      BLK-CONTR  = PAMKEY
%  REC-FORM   = (V,N)       REC-SIZE   = 2044
%  CODE       = EBCDIC      LABEL      = (STD,1)     FILE-SEQ   = 1
%  BLK-OFFSET = 4
%:2OS2: TAPE   :      1 FILE
/add-file-link link=dmcopy11,file-name=max.tape-file.2,sup=*tape(f-seq=2)  ------- (10)
/copy-file from=max.tape-file.2,to=#copy.tape-2          ------------------------- (11)
/show-file-attr max.tape-file.2,inf=(org=*yes)            ------------------------ (12)
%          :2OS2:$USER1.MAX.TAPE-FILE.2
%  -------------------------------- ORGANIZATION --------------------------------
%  FILE-STRUC = SAM         BUF-LEN    = STD(1)      BLK-CONTR  = PAMKEY
%  REC-FORM   = (V,N)       REC-SIZE   = 2044
%  CODE       = EBCDIC      LABEL      = (STD,1)     FILE-SEQ   = 2
%  BLK-OFFSET = 4
%:2OS2: TAPE   :      1 FILE
```

(1) Output of all catalog entries for tape files of the user ID *USER1* showing information on the file organization and volume allocation. Two tape files exist on the magnetic tape cartridge *D2326K*. The file *MAX.TAPE-FILE.2* follows the file *MAX.TAPE-FILE.1* (see the *FILE-SEQ* output fields).

(2) The tape files on the tape cartridge *D2326K* are exported. Messages are output to SYSOUT.

(3) Since more than one file is involved here, a prompt is issued to request confirmation. When this prompt is confirmed with *Y*, the catalog entries for the tape files *MAX.TAPE-FILE.1* and *MAX.TAPE-FILE.2* are deleted.

(4) No catalog entries for tape files exist at this point.

(5) Imports the tape file *MAX.TAPE-FILE.1* from the tape cartridge *D2326K*.

(6) Imports the tape file *MAX.TAPE-FILE.2* from the tape cartridge *D2326K*.

(7) Output of the catalog entries for the tape files showing information on the file organization and volumes. Only the catalog entries have been created so far (as if with CREATE-FILE). The file attributes have not yet been entered (value NONE or 0). The volume *D2326K* has only been registered (the value is still in parentheses). See also the output under step .

(8) The tape file *MAX.TAPE-FILE.1* is now accessed with the COPY-FILE command. The file is read and its contents are copied into the temporary file *#COPY.TAPE-1*.

(9) The catalog entry for the tape file *MAX.TAPE-FILE.1*

(10) A TFT entry with the default link name *DMCOPY11* is created for the tape file *MAX.TAPE-FILE.2*. This file is then opened by COPY-FILE as the input file. Since *MAX.TAPE-FILE.2* is the second file on the tape cartridge *D2326K*, FILE-SEQUENCE=2 was defined in the TFT entry. Without this specification, FILE-SEQUENCE=1 would have been assumed as in the case of *MAX.TAPE-FILE.1*.

(11) The tape file *MAX.TAPE-FILE.2* is now accessed with the COPY-FILE command. The file is read and its contents are copied into the temporary file *#COPY.TAPE-2*.

(12) The catalog entry for the tape file *MAX.TAPE-FILE.2*

Example 3: Updating catalog entries for files on private disk

```
/show-file-attr select=(sup=*priv,gen=*yes),inf=*all           --------------------   (1)
%0000000000*:20S2:$USER1.MAX.GROUP.1 (FGG)
%  ------------------------------- HISTORY      -------------------------------
%  CRE-DATE   = 2012-01-18  ACC-DATE   = NONE         CHANG-DATE = 2012-01-18
%  CRE-TIME   =   00:00:00  ACC-TIME   = NONE         CHANG-TIME =   00:00:00
%  ACC-COUNT  = 0           S-ALLO-NUM = 0
%  ------------------------------- SECURITY     -------------------------------
%  READ-PASS  = NONE        WRITE-PASS = NONE         EXEC-PASS  = NONE
%  USER-ACC   = OWNER-ONLY  ACCESS     = WRITE        ACL        = NO
%  AUDIT      = NONE        FREE-DEL-D = *NONE        EXPIR-DATE = 2012-01-18
%  DESTROY    = NO          FREE-DEL-T = *NONE        EXPIR-TIME =   00:00:00
%  SP-REL-LOCK = NO           ENCRYPTION = *NONE
%  ------------------------------- BACKUP       -------------------------------
%  BACK-CLASS = A           SAVED-PAG  = COMPL-FILE  VERSION    = 0
%  MIGRATE    = ALLOWED
%  ------------------------------- GENERATION-INFO -----------------------------
%  MAXIMUM    = 3           BASE-NUM   = 0           OVERFL-OPT = DELETE-ALL
%  FIRST-GEN  = 4           LAST-GEN   = 4
%  EXTENTS     VOLUME     DEVICE-TYPE
%              WORK01       D3435
%0000000003*:20S2:$USER1.MAX.GROUP.1(*0004)
%  ------------------------------- HISTORY      -------------------------------
%  CRE-DATE   = NONE        ACC-DATE   = NONE        CHANG-DATE = NONE
%  CRE-TIME   = NONE        ACC-TIME   = NONE        CHANG-TIME = NONE
%  ACC-COUNT  = 0           S-ALLO-NUM = 0
%  ------------------------------- SECURITY     -------------------------------
%  READ-PASS  = NONE        WRITE-PASS = NONE        EXEC-PASS  = NONE
%  USER-ACC   = OWNER-ONLY  ACCESS     = WRITE       ACL        = NO
%  AUDIT      = NONE        FREE-DEL-D = *NONE       EXPIR-DATE = NONE
%  DESTROY    = NO          FREE-DEL-T = *NONE       EXPIR-TIME = NONE
%  SP-REL-LOCK= NO           ENCRYPTION = *NONE
%  ------------------------------- BACKUP       -------------------------------
%  BACK-CLASS = A           SAVED-PAG  = COMPL-FILE  VERSION    = 0
%  MIGRATE    = ALLOWED
%  ------------------------------- ORGANIZATION -------------------------------
%  FILE-STRUC = NONE        BUF-LEN    = NONE        BLK-CONTR  = NONE
%  IO(USAGE)  = READ-WRITE  IO(PERF)   = STD         DISK-WRITE = IMMEDIATE
%  REC-FORM   = NONE        REC-SIZE   = 0
%  AVAIL      = *STD
%  ------------------------------- ALLOCATION   -------------------------------
%  SUPPORT    = PVT         S-ALLOC    = 9           HIGH-US-PA = 0
%  EXTENTS     VOLUME     DEVICE-TYPE      EXTENTS     VOLUME     DEVICE-TYPE
%      1       WORK01       D3435
%  NUM-OF-EXT = 1
%:2OS2: PRDISC:      2 FILES RES=         3 FRE=          3 REL=         0 PAGES
/write-text '*** Datum: &(DATE(FORMAT=*GERMAN)) ***'          --------------------   (2)
*** Datum: 18.03.2012 ***
/import-file sup=*by-file-name(file-name=max.group(*4)),inf=*all  ------------------   (3)
%  DMS0533 REQUESTED FILE NOT CATALOGED IN PUBSET '2OS2'. COMMAND TERMINATED
/import-file sup=*by-file-name(file-name=max.group.1),inf=*all    ------------------   (4)
%  DMS0533 REQUESTED FILE NOT CATALOGED IN PUBSET '2OS2'. COMMAND TERMINATED
/show-file-attr max.group*,select=(sup=*priv)               -----------------------   (5)
%  DMS06CC NO FILE CORRESPONDING TO SPECIFIED OPERANDS
```

(1) Returns information on all catalog entries for files on private disk. The private disk *WORK01* contains the file generation group *MAX.GROUP.1* with generation *0004. The group entry was created on 18.01.2012. Generation *0004 is merely cataloged. It has not yet been opened with OPEN=*OUTPUT.

(2) Output of the current date using WRITE-TEXT to replace an expression in the output text of the builtin DATE( ) function (a component of the software product SDF-P) in order to indicate the date in German. The date is 18.03.2012.

(3) The private disk *WORK01* was made available to the user by the system administration only for the month of January. At the start of February, all the data on this disk was deleted in an VOLIN run, but the entry in the file catalog was not updated. The entry in the catalog file was not updated in the process. The IMPORT-FILE command is now used in order to update the entry for generation *0004 of the file generation group *MAX.FILE.GROUP.1* from the VTOC of the private disk. Since the VTOC of the private disk *WORK01* does not contain an entry for this generation, the catalog entry is also deleted, and message *DMS0533* is output.

(4) The IMPORT-FILE command updates the group entry of the file generation group *MAX.FILE.GROUP.1* from the VTOC of the private disk: Since the VTOC of the private disk *WORK01* does not contain an entry for this generation, the catalog entry is also deleted, and message *DMS0533* is output.

(5) The output shows that there are no catalog entries for files whose name starts with *MAX.GROUP.*

IMPORT-NODE-FILE

Create catalog entry for node files (import node files)

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION TSOS
Routing code:	$ (bei NBCONOPI=N) bzw. E (bei NBCONOPI=Y)

Function

The IMPORT-NODE-FILE command creates the catalog entries for node files (files of the type NODE-FILE) which are located on a Net-Storage volume (import). DMS creates the catalog entry for a node file in TSOSCAT and in the file catalog of the Net-Storage volume on the basis of the Inode attributes on the NFS server.

Optionally the catalog entries of already imported node files can be updated on the basis of the Inode attributes on the NFS server (REPLACE=*NODE-FILE-UPDATE operand).

Catalog entries of node files can be deleted again with the EXPORT-NODE-FILE command (export).

Privileged functions

System support staff may export files belonging to any user ID. It may ignore the passwords of protected files and also branch to guided dialog whenever the user ID is changed.

By default, systems support (TSOS privilege) is a co-owner of all the files (and can therefore export files under any user ID). When SECOS is used, this co-ownership can be restricted.

Systems support can generate files under any user ID (TSOS privilege).In conjunction with the SECOS software product a user can allow other user IDs to act as co-owners. Co-owners of a user ID are then also allowed to export a file.

Format

IMPORT-NODE-FILE
VOLUME = <vsn 6..6>
,**F**ILE-**NAME** = <filename 1..54 without-cat-gen with-wild(80)>
,**FILE-STRUC**TURE = ***STD** / ***PAM**
,**REPL**ACE = ***NO** / *YES / ***NODE-FILE-UPDATE**
,**IGNORE-PROT**ECTION = ***NO** / *YES
,**REPORTING** = ***ERROR** / ***FULL**
,**OUTPUT** = list-poss(2): ***NONE** / ***SYSLST** / ***SYSOUT**
,**PUBSET** = ***STD** / <cat-id 1..4>

VOLUME = <vsn 6..6>
Volume serial number (VSN) of the Net-Storage volume on which the node files to be imported are stored.

FILE-NAME = <filename 1..54 without-cat-gen with-wild(80)>
Specifies the node files which are to be cataloged.
For privileged users, see "Privileged functions" on page 3-442.

FILE-STRUCTURE = *STD / *PAM
Specifies the FILE-STRUCTURE attribute of the node file which is entered in the file catalog in accordance with the REPLACE operand.

FILE-STRUCTURE = *STD
The following applies when REPLACE=*NO/*YES: A node file is imported into BS2000 as a PAM file if the file size on the NFS file system is not equal to zero. If the file size on the NFS file system is equal to zero, the imported file is assigned the default attributes of a file generated with CREATE-FILE.
The following applies when REPLACE=*NODE-FILE-UPDATE: The catalog entries are updated on the basis of the Inode attributes.

FILE-STRUCTURE = *PAM
The following applies when REPLACE=*NO/*YES: A node file is imported into BS2000 as a PAM file irrespective of the file size on the NFS file system.
The following applies when REPLACE=*NODE-FILE-UPDATE: The catalog entries are updated on the basis of the Inode attributes.

REPLACE = *NO / *YES / *NODE-FILE-UPDATE
Specifies whether files which already exist in BS2000 are replaced or whether only the catalog entry is updated on the basis of the Inode attributes on the NFS server.

REPLACE = *NO
Files which already exist are not replaced, nor are their catalog entries updated.

REPLACE = *YES
Files which already exist on the pubset are replaced by the specified node files. Files on public space and on Net-Storage are deleted and files are exported to private disk. When the node files are imported, the entries in TSOSCAT and in the file catalog of the Net-Storage volume are created again.

REPLACE = *NODE-FILE-UPDATE
In the case of files which already exist, the entries in TSOSCAT and in the file catalog of the Net-Storage volume are updated on the basis of the Inode attributes on the NFS server.

IGNORE-PROTECTION = *NO / *YES
*The operand value *YES is reserved for privileged users.* Systems support can determine whether the protection attributes should be ignored for existing files when REPLACE=*YES is specified.

REPORTING = *ERROR / *FULL
Determines the scope of the processing log which can be requested in the OUTPUT operand.

REPORTING = *ERROR
Only files which could not be imported are listed. The reason is displayed by means of a message code.

REPORTING = *FULL
All files are listed. In the case of files which could not be imported, the reason is displayed with a message code.

OUTPUT = *NONE / list-poss(2): *SYSOUT / *SYSLST
Specifies whether a processing log is to be output.

OUTPUT = *NONE
No processing log is output.

OUTPUT = *SYSOUT
A processing log is output to SYSOUT.

OUTPUT = *SYSLST
A processing log is output to SYSLST.

PUBSET = *STD / <cat-id 1..4>
Specifies the catalog ID of the pubset in which the files are to be cataloged. The Net-Storage volume specified in the VOLUME operand must be assigned to the pubset specified here.
The default setting is *STD, i.e. the catalog entries are set up under the file catalog of the default pubset of the user ID (see the output field `DEFAULT-PUBSET` in the SHOW-USER-ATTRIBUTES command).

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
	1	CMD05EE	Path name too long after completion
	1	CMD0624	File name invalid
	32	DMS0578	Internal error in file protection check
	32	DMS05C7	DMS internal error
	64	CMD0102	Interrupted by K2 key
	64	CMD0216	Privilege errors
	64	DMS0512	Requested catalog not found
	64	DMS051B	Requested user ID not in pubset
	64	DMS051C	User not authorized to access pubset
	64	DMS0535	No access authorization on the catalog entry of the file
	64	DMS0594	Not enough virtual memory available
	64	DMS05FC	Specified user ID not in home pubset
	64	DMS0610	Function execution supplied a return code for at least one of the selected file names
	64	DMS0640	Access to Net-Storage is rejected by the ONETSTOR subsystem because of communication problems with the net client
	64	DMS0642	Large files not permitted on pubset
	64	DMS0643	Net client reports access error
	64	DMS0644	Net client reports internal error
	64	DMS0645	File does not exist on Net-Storage
	64	DMS0649	Net server reports POSIX-ACL error
	64	DMS064A	Net client reports that access to files is forbidden on the Net-Storage volume
	64	DMS064B	Access to node files not supported by the net client
	64	DMS064C	Directory of the specified user ID does not exist on Net-Server
	64	DMS064D	File is not a node file
	64	DMS064E	Node file is not located on the specified Net-Storage volume
	64	DMS064F	FCB type of the file and specified file structure do not match
	64	DMS0650	No node file found to import
	64	DMS0651	File exists, import not possible
	64	DMS06CC	No file matches selection criteria

IMPORT-PUBSET

Import pubset

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Pubset and MRSCAT management
Domain:	MULTI-CATALOG-AND-PUBSET-MGMT
Privileges:	TSOS OPERATING
Routing code:	R

Function

This command creates a separate task under the control of the calling task.This task performs IMPORT processing asynchronously to the calling task. This task requests all the resources.
Each volume set of an SM pubset is imported separately.

The F5 labels are read in and reconstructed if necessary. The mnemonic names of the disks belonging to the pubset are entered or updated in the SVL of the system disk (pubres). The name of the system disk is stored in the pubset's MRSCAT entry. The user catalog is opened and the specified pubset is set to “accessible”. The user tables that contain the current user information are set up in the memory. Access to this pubset is then permitted. Depending on the specifications made by system support staff in the ADD-MASTER-CATALOG-ENTRY command, automatic start of SPEEDCAT with or without task switching is supported. SPOOL is notified and the spoolout jobs are transferred to TYPE5/KP or TYPE4. User jobs in the wait state HELD-BY-CALENDAR or HELD-BY-PUBSET because the imported pubset was previously unavailable are released.
When a pubset is imported with ACTUAL-JOIN=*FIRST all files and job variables of the user ID TSOS are retained. Files and job variables of all other users are deleted.

The change in the availability of a pubset is notified to all active processors in the network.

A number of different pubsets can be imported to a processor.

Format

```
IMPORT-PUBSET

 PUBSET = <cat-id 1..4>
,ACTUAL-JOIN = *STD / *ZIP / *FIRST / *NONE
,MONJV = *NONE / <filename 1..54 without-gen-vers>
,JV-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> / <integer -2147483639..2147483639>
,RESIDENT-BUFFERS = *STD / *NO / *YES
,NUMBER-OF-BUFFERS = *STD / <integer 1..255>
,USE = *STD / *SHARE / *EXCLUSIVE(...) / *FROM-REMOTE(...)
   *EXCLUSIVE(...)
     |   CONVERT-VOLUME-SET = *NO / *YES
   *FROM-REMOTE(...)
     |   HOST-NAME = *NONE / <alphanum-name 1..8>
,SHARER-TYPE = *STD / *SLAVE / *MASTER(...)
   *MASTER(...)
     |   MASTER-CHANGE = *NO / *YES
,SESSION-CHECK-MSG = *YES / *NO
,RECONSTRUCT-USERCAT = *NO / *RESET / *BY-BACKUP(...)
   *BY-BACKUP(...)
     |   SCOPE = *ALL / *BACKUP / *TSOSCAT
,RECONSTRUCT-F5-LABEL = *NO / *YES
,DEFECT-VOLUME-SET = *NONE / list-poss(256): <cat-id 1..4>
,IN-HOLD-VOLUME-SET = *NONE / list-poss(256): <cat-id 1..4>
,REPAIR-TSOSCAT = *NO / *YES
,EXTRA-LARGE-CAT-CONV = *NO / *YES
,CHECK-PUBSET-MIRRORS = *NO / *YES
```

Operands

PUBSET = <cat-id 1..4>
Catalog ID of the pubset to be imported.

ACTUAL-JOIN = *STD / *ZIP / *FIRST / *NONE
Specifies handling of the user catalog during import.

ACTUAL-JOIN = *STD
Opens the existing user catalog. The effect of this operand is governed by the RECONSTRUCT-USERCAT operand. With RECONSTRUCT-USERCAT=*NO, the existing user catalog ($TSOS.SYSSRPM) is opened. The effect of the *RESET and *BACKUP operand values is described under the RECONSTRUCT-USERCAT operand.

ACTUAL-JOIN = *ZIP
This operand value may only be specified in the event of disk storage space bottlenecks to prevent a log file and the .BACKUP file for the user catalog being generated by user administration. Apart from that, this operand value is treated in exactly the same way as *STD, in other words processing is governed by the value of the RECONSTRUCT-USERCAT operand.

ACTUAL-JOIN = *FIRST
Creates a new user catalog containing only the system user IDs. The TSOS user IDs remain intact. This operand value may only be specified when a pubset is imported for the first time after generation. The RECONSTRUCT-USERCAT operand is ignored.

ACTUAL-JOIN = *NONE
This operand value is only used internally by specific utilities. This operand value may only be specified when a pubset is imported for the first time after generation. It is evaluated by the SIR utility, for instance. The RECONSTRUCT-USERCAT operand is ignored.

MONJV = *NONE / <filename 1..54 without-gen-vers>
Specifies whether a monitoring job variable is set. This variable is set to the following values during import:

$I	at the start of import
$R	at the end of import if the entire pubset has been successfully imported
$A	import was terminated due to an error
$W	if a shared pubset has been imported and the availability of the master processor has not yet been confirmed

This operand should only be specified if the JV software product is used.

Notes

- The specified job variable only becomes the monitoring job variable if the pubset has not been imported.
- The job variable must already be cataloged, otherwise it is not set. However, IMPORT processing continues even if the job variable is not defined.

JV-PASSWORD = _*NONE_ / <c-string 1..4> / <x-string 1..8> / <integer -2147483639..2147483639>

Password of the job variable if this is write-protected.
The operand JV-PASSWORD is defined as “secret”:

- The password entered is not logged.
- The input field is automatically blanked out in the guided dialog.
- In unguided dialog and foreground procedures, the entry *SECRET or ^, SDF provides a blanked out input field for inputting the password .

RESIDENT-BUFFERS = _*STD_ / *NO / *YES

Specifies whether resident or nonresident buffers are to be created (also see “Notes” on page 3-454).

RESIDENT-BUFFERS = _*STD_

The value specified in the MRSCAT entry applies.

RESIDENT-BUFFERS = *NO

A nonresident buffer is created.

RESIDENT-BUFFERS = *YES

A resident buffer is created.

NUMBER-OF-BUFFERS = _*STD_ / <integer 1..255>

Defines the number of buffers.

NUMBER-OF-BUFFERS = _*STD_

The value specified in the MRSCAT entry applies.

NUMBER-OF-BUFFERS = <integer 1..255>

The specified number of buffers (minimum 6; see “Notes” on page 3-454) is created.

USE = _*STD_ / *SHARE / *EXCLUSIVE(...) / *FROM-REMOTE(...)

Defines the mode of access to the imported pubset. Please observe the relevant conditions and requirements.

USE = _*STD_

The value specified in the MRSCAT entry applies.

USE = *SHARE

Specifying this operand only makes sense if the MRSCAT is configured for shareability. The pubset is to be imported as a shared pubset.

USE = *EXCLUSIVE(...)
The pubset is to be imported as an exclusive pubset. A volume set can also be imported in order to perform recovery.

CONVERT-VOLUME-SET = *NO / *YES
Specifies whether a normal import is to be performed or whether the volume set of a destroyed SM pubset is to be converted into an SF pubset.

CONVERT-VOLUME-SET = *NO
A normal import is to be performed.

CONVERT-VOLUME-SET = *YES
A volume set that is to be converted into an SF pubset was specified in the PUBSET operand. This function can be used to re-establish the accessibility of data in the volume sets of an SM pubset whose control volume set has become inoperative. Please take account of the notes on page 3-456.

USE = *FROM-REMOTE(...)
Makes the catalog of a “remote-imported” pubset available if MRSCAT entries were defined after HIPLEX MSCF network startup..

HOST-NAME = *NONE / <alphanum-name 1..8>
BCAM name of the partner host owning the pubset. There has to be an MSCF link to the computer.

SHARER-TYPE = *STD / *SLAVE / *MASTER(...)
*Specifying this operand only makes sense if the MRSCAT is configured for shareability or if the USE operand is set to *SHARE.* Defines the ownership of the pubset. Please observe the relevant conditions and requirements.

SHARER-TYPE = *STD
The system selects the system automatically on the basis of the pubset’s attributes:

- If the shared pubset already has a master or another system is explicitly predefined as the master (with /SET-PUBSET-ATTRIBUTES), the system selects SHARER-TYPE=*SLAVE.
- In all other cases it selects SHARER-TYPE=*MASTER.

SHARER-TYPE = *SLAVE
The home system is to be a slave sharer.

SHARER-TYPE = *MASTER(...)
The home system is to take over the as yet unassigned ownership of the pubset being imported.

MASTER-CHANGE = *NO
The home system is to take over the as yet unassigned ownership of the pubset being imported. The command is rejected if the pubset has already been imported.

MASTER-CHANGE = *YES
Systems support can initiate an explicit master change from a slave system following the failure of the master or after the EXPORT-PUBSET command was issued with MASTER-CHANGE=*YES from the master.
An explicit master change is not possible in the case of BACKUP-MASTER=*NONE and ALTERNATE-BACKUP=*NONE (see the SET-PUBSET-ATTRIBUTES command).
An explicit master change can be used in the following situations:

- The master change initiated automatically by watchdog at a specified backup master was rejected with an error.
- An automatic master change is hindered by BACKUP-MASTER=*NONE and ALTERNATE-BACKUP=*BY-OPERATOR.

SESSION-CHECK-MSG = <u>*YES</u> / *NO
Specifies whether a message is to be output if, during import processing, a check reveals that the pubset involved has already been imported on another system or that the last session was terminated abnormally.

SESSION-CHECK-MSG = <u>*YES</u>
If the check reveals that the pubset has already been imported on another system or that the last session was terminated abnormally, message `DMS038C` is output.

SESSION-CHECK-MSG = *NO
The check is deactivated during an import job for a data pubset; output of message `DMS038C` is suppressed. System support staff should only use this option, however, if the system was loaded with automatic startup.

RECONSTRUCT-USERCAT = <u>*NO</u> / *RESET / *BY-BACKUP(...)
This operand enables the caller to reconstruct the user catalog, either in full on the basis of a backup copy or from scratch on the basis of the user IDs included in TSOSCAT. The operand is evaluated only in conjunction with the settings ACTUAL-JOIN=*STD/*ZIP; it is ignored with ACTUAL-JOIN=*FIRST.

RECONSTRUCT-USERCAT = <u>*NO</u>
No reconstruction is performed.

RECONSTRUCT-USERCAT = *RESET
The user catalog is reconstructed. The user IDs are taken from the current TSOSCAT. The entries are reset.

RECONSTRUCT-USERCAT = *BY-BACKUP(...)
The user catalog is reconstructed on the basis of a backup user catalog ($TSOS.SYSSRPM.BACKUP) saved earlier with ARCHIVE and restored to the system.

SCOPE = <u>*ALL</u> / BACKUP / *TSOSCAT
These specifications define how to deal with user IDs which are included only in the backup or only in TSOSCAT.

SCOPE = <u>*ALL</u>
All the entries are copied from the backup. Any additional entries included in TSOSCAT are recreated.

SCOPE = BACKUP
The entries in the backup determine which user IDs are set up. Files belonging to user IDs not included in the backup are deleted.

SCOPE = *TSOSCAT
The entries in TSOSCAT determine which user IDs are set up. Only the user IDs which are included in it are copied from the backup or recreated.

RECONSTRUCT-F5-LABEL = <u>*NO</u> / *YES
Specifies whether reconstruction of the F5 label is to be initiated explicitly.

RECONSTRUCT-F5-LABEL = <u>*NO</u>
Reconstruction of the F5 label is not initiated explicitly. However, the system can initiate a reconstruction internally, regardless of this (e.g. following a system crash). A reconstruction will also take place if a check of the TSOSCAT user chain is requested (operand REPAIR-TSOSCAT=*YES).

RECONSTRUCT-F5-LABEL = *YES
Reconstruction of the F5 label is performed in any case. This covers all volume set belonging to an SM pubset.

DEFECT-VOLUME-SET = <u>*NONE</u> / list-poss(256): <cat-id 1..4>
This operand is only evaluated for SM (System-Managed) pubsets which are imported exclusively or as master. The operand defines which volume sets are defective and are be removed from the SM pubset. These volume sets are also to be removed from the pubset configuration file and from MRSCAT. The control volume set cannot be removed.

DEFECT-VOLUME-SET = <u>*NONE</u>
All the volume sets listed in the pubset configuration file are imported. If one or more volume sets are found to be defective, the import process aborts. The catalog name of each volume set found to be defective is displayed. The import process must then be repeated, naming all the defective volume sets.

DEFECT-VOLUME-SET = list-poss(256): <cat-id 1..4>
Specifies the catalog names of those volume sets which are no longer to be part of the SM pubset and are to be removed from the pubset configuration file and from MRSCAT. All the files in a defective volume set are removed from the file index of the SM pubset in question

and added to a file named $TSOS.SYS.PUBSET.DEFECT.<volset-id>.<date>.<time>, where <date> is in yyyy-mm-dd format and <time> is in hhmmss format. The file names added to this file form the basis for subsequent reconstruction using the HSMS utility.

IN-HOLD-VOLUME-SET = *NONE / list-poss(256): <cat-id 1..4>
This operand is only evaluated for SM (System-Managed) pubsets which are imported exclusively or as master. The operand defines which volume sets are to be placed in the IN-HOLD status when they are imported, i.e. these volume sets are temporarily flagged as not operable. They can be activated again with the MODIFY-PUBSET-RESTRICTIONS command. A volume set that is exported while IN-HOLD, is implicitly reactivated when imported if it was not named explicitly in the IN-HOLD-VOLUME-SET operand.

IN-HOLD-VOLUME-SET = *NONE
None of the volume sets is put IN-HOLD.

IN-HOLD-VOLUME-SET = list-poss(256): <cat-id 1..4>
Specifies the volume sets that are put IN-HOLD while being imported.

REPAIR-TSOSCAT = *NO / *YES
Specifies whether the TSOSCAT user chains are repaired during the CMS start phase of the import.

REPAIR-TSOSCAT = *NO
TSOSCAT user chains are not repaired.

REPAIR-TSOSCAT = *YES
This specification should only be entered if the import was not possible because of a defective TSOSCAT. Catalog entries can be lost when the repair process takes place.
TSOSCAT user chains are analyzed in the CMS start phase of the import and they are repaired, if necessary. Defective blocks are removed. If a file’s catalog entry was on a defective block, that file is no longer accessible. For further information on the reconstruction of file catalogs, see the “Introduction to System Administration” [14]. An F5 label reconstruction for the entire pubset is performed.

EXTRA-LARGE-CAT-CONV = *NO / *YES
This operand defines whether the catalog of an SF pubset or all catalogs of an SM pubset are to be converted to the “extra large” format. For details on catalog formats, see the “Introduction to System Administration” [14].

Note

> Pubsets with catalogs in “extra large” format cannot be imported into BS2000/OSD-BC ≤ V6.0B!

EXTRA-LARGE-CAT-CONV = *NO
No conversion to “extra large” catalog format takes place.

EXTRA-LARGE-CAT-CONV = *YES
Conversion takes place to “extra large” catalog format. Conversion is performed subsequently for volume sets of an SM pubset which are in the IN-HOLD status when they are reactivated.

CHECK-PUBSET-MIRRORS = <u>*NO</u> / *YES
Determines whether the homogeneity of the pubset is to be checked. In the case of a pubset which is not mirrored a check is also made to see whether only individual disks are mirrored. A pubset is homogeneous when all volumes of the pubset have the same mirroring properties.

The homogeneity check is performed for additional mirror units (SRDF and TimeFinder/Mirror functions on Symmetrix disk storage systems) and for clones (EC-Clone functions on ETERNUS DX, TimeFinder/Clone on Symmetrix and Snapview clone on CLARiiON disk storage systems).

CHECK-PUBSET-MIRRORS = <u>*NO</u>
No homogeneity check is performed.

CHECK-PUBSET-MIRRORS = *YES
A homogeneity check is performed.

Return codes

(SC2)	SC1	Maincode	Bedeutung
	0	CMD0001	Command executed without error
1	0	DMS0350	Pubset already available
	1	CMD0202	Syntax error
	32	DMS0352	Error while accessing MRSCAT entry
	64	DMS0360	No authorization for command
	64	DMS036B	Missing MRSCAT entry, wrong type
	64	DMS037B	Import as shared PVS not possible
	64	DMS13C9	Pubset has already been remotely imported; import mode cannot be changed
	130	DMS0351	Other import/export task active
	130	DMS035A	Maximum task number reached
	130	DMS0362	Class-4 memory error

Notes

– The home pubset cannot be imported explicitly. The home pubset is imported automatically during system initialization (startup phase).

- The IMPORT-PUBSET command generates a new task, the IMPORT task, and starts it. Actual importing is performed by this IMPORT task asynchronously with the calling task. After successful generation of the IMPORT task, the following message is output on the console:

  ```
  DMS035B IMPORT PUBSET TASK WITH TSN '(&00)' FOR PUBSET WITH PUBSET ID
  '(&01)' HAS BEEN CREATED AND STARTED.
  ```

  All messages output by the IMPORT job are sent to the console.

- Specifications made via `RESIDENT–BUFFERS` and `NUMBER–OF–BUFFERS` can indirectly influence the working set or paging rate of the system. If, for example, a large number of resident buffers are created on a small system, the cataloging operations will be faster but the paging rate for all other applications will increase. On the other hand, a class 4 memory bottleneck can occur for nonresident buffers. If no buffer specifications are made, the system's default values are used. The following 3-level hierarchy applies:

  1. Explicit parameter specifications in the IMPORT-PUBSET command.
  2. Specifications via the ADD- OR MODIFY-MASTER-CATALOG-ENTRY command.

     If only one of the parameters RESIDENT-BUFFERS or NUMBER-OF-BUFFERS is specified, the default value applies to the other one (RESIDENT-BUFFERS= *NO, NUMBER-OF-BUFFERS=32).
  3. Specification according to system parameters CATBUFR and BMTNUM.

- For reasons of performance and reliability, a minimum number of CMS buffers is defined by the system. If a smaller number is specified explicitly in the NUMBER-OF-BUFFERS operand, the minimum number defined by the system is set.

- If the pubset to be imported is still locked due to a previous system crash, the operator can cancel this lock by means of the UNLOCK-DISK command. However, the operator must first ensure that the pubset is not being used by another system which is currently active.

- The MRSCAT of the pubset must contain a static definition of how import processing is to react in the event of an error during a new reservation or reconnection to a cache area (see also the MODIFY-PUBSET-CACHE-ATTRIBUTES command). Whenever a new MRSCAT entry is added, the default value, i.e. cancelation in the event of an error, applies. The following error situations may occur:

  New reservation of a cache area: The cache area cannot be made available in the required size. Possible reactions:

  1. Import processing continues using the remaining buffer available.
  2. Import processing is aborted.

  Reconnection of a cache area: The cache area with the cache ID stored in the SVL cannot be connected. Possible reactions:

1. Import processing continues without using the old cache buffer. The data on the pubset is then inconsistent. For all files that may be inconsistent, a corresponding indicator is set in TSOSCAT to prevent these files from being reopened. Only system support staff can reset this indicator; however, the file can be deleted by its owner.
2. Import processing is aborted.

Notes on importing a volume set with conversion to an SF pubset

The function “import and convert volume set” (see the USE=*EXCLUSIVE(CONVERT-VOLUME-SET=*YES) operand on page 3-450) is a basic function for the recovery of SM pubsets whose control volume set has failed: converting the remaining intact volume sets into SF pubsets makes the data present on them available again and it is then possible to recover the SM pubset using SMPGEN (see also the “System Managed Storage” User Guide [45]).
The conversion of volume sets results in operable SF pubsets. Consequently, this function should not be used for anything other than the recovery of SM pubsets without very careful consideration. The specific problem of the maximum length of file names will be examined in the following. In all cases, it is necessary to remember that in an SM pubset, the files processed by an application are not all located in the same volume set. Following the subdivision of an SM pubset into SF pubsets by means of the import/convert function it is therefore no longer possible to address the files belonging to an application uniquely via defcat or a **single** catalog ID. The corresponding metadata is also no longer available in full.

The procedure described in the chapter “SMPGEN” of the “Utility Routines” User Guide [9] largely applies to the preparation of an SMPGEN run performed in order to recover an SM pubset. However, the FDDRL backup of the volumes in question should be performed within a recovery scenario **before** the “conversion import” step in order to ensure that this step is also protected by a physical backup.

When performing conversion, the following points relating to the control volume set, the length of file names, file generations and DAB caches should be considered:

Control volume set

It is not possible to convert control volume sets, and such an operation would also have no point, since the function “import and convert” is used for recovery purposes on the failure of a control volume set. The metadata stored on the failed control volume set must be recovered from a backup or other source following restoration of the SM pubset by SMPGEN.

File name length

On conversion, the catalog ID of the volume set is taken over as the catalog ID of the resulting SF pubset. If the length of this catalog ID is greater than the length of the (previously valid) catalog ID of the SM pubset then it is necessary to ensure that the maximum possible file name length (54 characters) is not exceeded. To this end, when conversion is performed, any files whose names would exceed the maximum permitted length are renamed. A message, which requests operator confirmation, is issued at the console in order to query whether renaming should be performed. The names of the renamed files have the following structure:

```
:<catid>:$<userid>.S.RENAME.<timestamp>.<tempfile_indicator>.<counter>
```

The name sections have the following meanings:

<catid>	Catalog ID of the SF pubset
<userid>	user ID
<timestamp>	Time stamp in ISO4 format
<tempfile_indicator>	Displays whether a temporary file is concerned: T temporary file N "normal" file

Any file renaming operations that are performed are logged in an ISAM file on the home pubset. The name of the file is:

```
:<catid_home>:$TSOS.TSOSCAT.CONV.<catid>.<timestamp>
```

where <catid_home> indicates the catalog ID of the home pubset, <catid> the catalog ID of the SF pubset created as a result of the conversion and <timestamp> the time stamp in ISO4 format.

The log records indicate the allocation of the file names generated during conversion to the original file names. It is particularly important to ensure that file generations which have excessively long file names are also renamed. When they are renamed, they are also assigned the default name as described above and lose the “file generation” property.

Example

The SM pubset K contains a volume set F64K. Following an error in the control volume set, the remaining volume sets, including F64K, are initially converted into SF pubsets. This requires certain files to be renamed. The log file is located on the home pubset A:

```
:A:$TSOS.TSOSCAT.CONV.F64K.2011-10-05.112950
```

Contents of the log file:

```
:F64K:$USER1234.S.RENAME.2011-12-06-090300.N.000002
,:F64K:$USER1234.FILEA901234567890123456789012345678901234
:F64K:$USER1234.S.RENAME.2011-12-06-090300.N.000004
,:F64K:$USER1234.FILEGROUP3456789012345678901234567(*0001)
:F64K:$USER1234.S.RENAME.2011-12-06-090300.N.000005
,:F64K:$USER1234.FILEGROUP3456789012345678901234567(*0003)
:F64K:$USER1234.S.RENAME.2011-12-06-090300.T.000003
,:F64K:$USER1234.FILEXXX1234567890123456789012345678901234
:F64K:$USER1234.S.RENAME.2011-12-06-090300.N.000001
,:F64K:$USER1234.FILE8901234567890123456789012345678901234
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣
,:F64K:$USER1234.FGG1(*0002)
```

Following restoration of the SM pubset, the renamed files can be given their original names again with the help of the logging file. How to proceed in the case of file generation groups is described in the next paragraph ("File generation groups"). **Once recovery has been performed successfully and the original file names have been restored it is essential to delete the log files**, after backing them up if necessary. If this is not done, name conflicts may occur when importing and converting volume sets with the same name and, in some cases, may result in faulty operation.

File generation groups

In the case of SM pubsets, the file group's catalog entry is located on the control volume set while the catalog entries for the individual file generations can be distributed across the volume sets as required. Consequently, after the volume sets have been converted into SF pubsets, the individual file generations and associated catalog entries are present whereas group entries are not. The question of renaming file generations with "excessively long" file names during the conversion process has already been addressed.The names of the remaining file generations are logged without being renamed. When this is done, the log file contains spaces in place of the new file name (see example, page 3-458).

New catalog entries must be created for the file generation groups before these can be accessed. The following procedure is recommended:

A group entry is created for each corresponding individual generation:

```
/CREATE-FILE-GROUP  <filename>,*GENERATION-PARAMETER(MAXIMUM=1,
                    FIRST-GENERATION=n, LAST-GENERATION=n)
```

The content of the file generation is then copied to a file for which any required file name can be specified. The name of the file generation and the target file are logged. This file generation group is then deleted.

Once the SM pubset has been restored, it is possible to recover the file generation groups: The group entry is recreated. After this, the files that were created during conversion or manually from file generations are again added to the group as file generations by means of the MODIFY-FILE-ATTRIBUTES command. It should be noted that file generations that were stored on the control volume set of the original SM pubset are lost. This may result in discontinuities in the version numbers of the generation. This consideration should be borne in mind when performing the recovery.

DAB caches

When caching is performed with DAB (see also the "DAB" User Guide [5]) it is necessary to distinguish between ADM-PFA caching and (user) PFA caching. In both cases, synchronization with any existing cache data is performed as part of pubset import involving volume set conversion.

1. ADM-PFA caching
   ADM-PFA caching can be set for entire SM or SF pubsets, volume sets or individual files as required. When this is done, DAB uses the volumes in question and the names of the buffered files as the metadata. It is necessary to distinguish between the following three scenarios when an SM pubset's control volume set fails:

   a) An export of the intact volume set is possible and is performed
      In this case, the data from the volume sets is written back. Only control volume set data may be present in the cache. In order to prepare for pubset recovery it is necessary to terminate DAB caching for the objects in question using /STOP-DAB-

CACHING and also, if the cache contains data belonging to the defective control volume set, /FORCE-STOP-DAB-CACHING. When this is done, a list of the “pinned” files is created.

b) Import with conversion and cache present in the current session
This is the situation that occurs if the cache is not cleared in case a). However, in this situation the cache contains no data. Caching of the data ranges in question is prohibited due to the import/conversion of the volume set. Due to the fact that after import and conversion, the file names administered by DAB no longer correspond to the data present in the pubset, it is essential to clear the cache areas in question with /STOP-DAB-CACHING.

c) Defective control volume set after a crash with existing GS crash areas
When the volume sets are imported for conversion, all the GS cache areas are initially recovered by DAB. In this case, any data present in the cache is written back to disk. Caching is then prohibited as described above. Any remaining cache should be cleared as soon as possible (with STOP-DAB-CACHING/FORCE-STOP-DAB-CACHING).

2. User PFA
In the case of user PFA, a cache area is always assigned to one volume set (for SM pubsets). It will probably not be possible to clear the cache area for the defective control volume set and this will have to be terminated with the /FORCE-DESTROY-CACHE command. For the other volume sets, it is necessary to differentiate between the following two scenarios:

a) Export and import with conversion in the current session
In the case of all other volume sets, the assigned cache area is cleared correctly following an export due to a failed control volume set. No new cache area is set up when the volume sets are subsequently imported and converted.

b) Import and conversion following a crash with existing GS cache areas
If volume sets are imported and converted while GS cache areas are still present in a follow-up session, then the cache area is automatically restored on import, the cache data is written back and the cache area is then cleared again before conversion is initiated. This ensures that the cache data is consistent. This also applies for volume sets previously used as shared pubsets with cache areas in the global GS. When shared pubsets and local GS cache areas are used, restrictions exist with respect to volume set conversion: As a rule conversion goes hand in hand with the loss of cache data (only the cache area of the current system, not all the others, can be collected when conversion takes place). Consequently, a global GS should be chosen for shared SM pubsets using GS caching with PFA (this is also generally the case, see the “DAB” User Guide [5]).

Using the "Import with homogeneity check" function

The homogeneity check with regard to SRDF and/or BCV mirroring for a pubset which is to be imported is executed by specifying the CHECK-PUBSET-MIRRORS = *YES operand.

If in the course of an import a volume is detected which has different mirroring properties from volumes which have already been processed, the answerable message `DMS1369` is output on the console. Depending on the operator’s answer, one of the following procedures is selected:

- Pubset import is aborted.
- Import processing is continued despite the fact that the volume of the pubset which has just been processed has been recognized as inhomogeneous. In this case the message `DMS136B` is issued on the console for every further volume which has different mirroring properties.

INCLUDE-DEVICE-CONNECTION

Attach virtual connections

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Device management
Domain:	DEVICE
Privileges:	OPERATING
Routing code:	G

Function

This command enables the operator to attach virtual connections between the hardware units (CHN, CTL, DVC), i.e. permit the operating system to use these connections.

Effect of the INCLUDE-DEVICE-CONNECTION command:

1. If the connections specified were in the state “removed explicitly”, they assume the state “included”. The connections can be used again.
2. If the outer units associated with this connection were in the state “detached implicitly”, they assume the state “attached”. These hardware units can be used again.
3. In the outer hardware units which have assumed the state “attached” as per point 2, the outward connections are checked: if the connections were in the state “removed implicitly”, they assume the state “included”. These connections can be used again.
4. In the case of disk and tape devices which are capable of forming path groups, the newly attached paths are included in the path group.

Format

```
INCLUDE-DEVICE-CONNECTION

 FROM = *CHANNEL(...) / *CONTROLLER(...) / list-poss(8): <alphanum-name 2..2> / <x-text 4..4>
    *CHANNEL(...)
      |  CHANNEL-PATH-ID = list-poss(8): <x-text 2..2>
    *CONTROLLER(...)
      |  CONTROLLER-UNIT = list-poss(8): <alphanum-name 2..2> / <x-text 4..4>
,TO = *CHANNEL(...) / *CONTROLLER(...) / list-poss(8): <alphanum-name 2..2> / <x-text 4..4>
    *CHANNEL(...)
      |  CHANNEL-PATH-ID = list-poss(8): <x-text 2..2>
    *CONTROLLER(...)
      |  CONTROLLER-UNIT = list-poss(8): <alphanum-name 2..2> / <x-text 4..4>
,SCOPE = *OWN-SYSTEM-ONLY / *VM2000-GLOBAL
```

Operands

FROM =
Defines the virtual connection to be attached in terms of one of the delimiting hardware units. The direction in which the virtual connection is attached is not prescribed. FROM therefore does not have to be the inner and TO the outer unit.

FROM = *CHANNEL(...)
Defines the virtual connection to be attached in terms of a specific channel.

> **CHANNEL-PATH-ID = list-poss(8): <x-text 2..2>**
> Specifies the channel path ID of the channel that delimits the virtual connection. A maximum of 8 channels can be specified.

FROM = *CONTROLLER(...)
Defines the virtual connection to be attached in terms of a specific controller.

> **CONTROLLER-UNIT = list-poss(8): <alphanum-name 2..2> / <x-text 4..4>**
> Specifies the mnemonic device code (MN) of the controller that delimits the virtual connection. A maximum of 8 controllers can be specified.

FROM = list-poss(8): <alphanum-name 2..2> / <x-text 4..4>
Defines the virtual connection to be attached in terms of a specific device. A maximum of 8 devices (mnemonic device codes) can be specified.

TO = *CHANNEL(...) / *CONTROLLER(...) / list-poss(8): <alphanum-name 2..2> / <x-text 4..4>
Defines the virtual connection to be attached in terms of the other delimiting hardware unit. This unit represents the second element of the pair and is specified according to the FROM operand. A maximum of 8 units can be specified.

SCOPE =
Specifies how the command is to be executed under VM2000.

SCOPE = *OWN-SYSTEM-ONLY
The command is only executed in the local system.

SCOPE = *VM2000-GLOBAL
If entered at the VM Monitor System (VM1) the command is executed at all the guest systems running BS2000/OSD ≥ V5.0.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
4	64	NKR0...	Path already attached
12	64	NKR0...	Internal check negative
16	64	NKR0...	Caller error
20	64	NKR0...	Software error

Note

If there is an error in command termination, the maincode contains the message code of the message output during command processing.

Example

Attach the connection between device L1 and channel 12:

```
/INCLUDE-DEVICE-CONNECTION FROM=L1, TO=*CHANNEL(12)
```

or

```
/INCL-DEV-CONN FROM=L1,TO=*CH(12)
```

INFORM-ALL-JOBS

Send message to all active user tasks

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Operator terminal control
Domain:	JOB
Privileges:	OPERATING TSOS
Routing code:	E

Function

The INFORM-ALL-JOBS command enables systems support to send a message to all interactive users currently connected to the system. The system adds the date and time of day to the message.

Note

To send a message to a specific interactive user, systems support must use the INFORM-JOB command.

Format

INFORM**-ALL-JOBS**
MSG = <c-string 1..72 with-low>

Operands

MSG = <c-string 1..72 with-low>
A text string up to 72 characters long is sent to all active user tasks.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
2	0	NBR0725	Warning: The message could not be sent to all interactive tasks
2	32	NBR0727	Internal error: Error in $PSTMS call (incorrect UNIT, FUNCTION or VERSION)
	64	CMD0216	User does not have the required privilege
	130	EXC0061	Class 4 memory request failed; command processing aborted

Example

Systems support input:

```
/inform-all-jobs msg = ???'Attention SHUTDOWN in 15 minutes.'
```

Output on terminals of interactive users:

```
% BCST Attention SHUTDOWN in 15 minutes. :14:53:38 :12-01-26026
```

INFORM-JOB

Send message to specific user task

Description status: BS2000 OSD/BC V10.0A

Functional area: Operator terminal control

Domain: JOB

Privileges: OPERATING
TSOS

Routing code: E

Function

The INFORM-JOB command enables systems support to send a message to a specific batch or interactive job. The system adds the date and time of day to the message. The message is written to SYSOUT.

If the job has not yet been started (queue type 1) or has already terminated, the INFORM-JOB command will not be executed and the command submitter will be informed.

Note

The precise output format of the message is controlled by the NBMESLG system parameter.

Format

```
INFORM-JOB

 MSG = <c-string 1..151 with-low>
,JOB-IDENTIFICATION = *TSN(...) / *TERMINAL(...)
   *TSN(...)
     |  TSN = <alphanum-name 1..4>
   *TERMINAL(...)
     |  PROCESSOR = <name 1..8>
     | ,STATION = <name 1..8>
```

Operands

MSG = <c-string 1..151 with-low>
This operand represents the message to be transmitted.
Its maximum length is 151 characters. This length limitation does not apply to the date and time of day, which are added by the system.

JOB-IDENTIFICATION = *TSN(...) / *TERMINAL(...)
Specifies where the message is to be sent. The message can be sent either to a job identified by its task sequence number (TSN) or to an interactive task identified by processor and station name.

JOB-IDENTIFICATION = *TSN(...)
The message is to be sent to a job identified by its TSN.

TSN = <alphanum-name 1..4>
TSN (up to 4 alphanumeric characters) of the job to which the message is to be sent.

JOB-IDENTIFICATION = *TERMINAL(...)
The message is to be sent to an interactive task on a terminal.

PROCESSOR = <name 1..8>
BCAM name of the processor to which the terminal is connected.

STATION = <name 1..8>
BCAM name of the terminal to which the message is to be sent.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
2	0	EXC0064	Warning. Specified task has temporarily suppressed message function
1	1	EXC0240	Syntax error
2	64	EXC0080	Specified task does not exist
3	64	EXC0062	Specified task cannot receive message
4	64	EXC0081	Invalid task type
5	64	EXC0109	Semantic error; command ignored
	130	EXC0061	Not enough class 4 memory; command aborted

Example

Systems support input:

```
/inform-job msg = '*** Please release reserved tape devices?***' ,
           tsn = 0FC3
```

Output on terminal of interactive task 0FC3:

```
%MESS *** Please release reserved tape devices‘ *** :14:53:38 :12-01-26026
```

INFORM-OPERATOR

Send message to operator terminal

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing
Domain:	JOB PROGRAM
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The INFORM-OPERATOR command sends a message to the operator terminal.
The message is by default sent to the recipient defined by the MSGDEST system parameter.

If the message text starts with the “<“ character, the next character is checked. If this character is a valid authorization code (no distinction is made between uppercase/lowercase) then the message is sent to all the terminals to which this authorization code is assigned. If the next character is not defined as an authorization code, the message is also sent in accordance with the MSGDEST system parameter.

The user can also request a response from the operator (WAIT-RESPONSE operand). The task is put on hold until the response arrives.

Format

INFORM**-OPER**ATOR
MSG = <c-string 1..230 with-low> ,**WAIT-RESP**ONSE = ***NO** / ***Y**ES

Operands

MSG = <c-string 1..230 with-low>
Text of the message which is to be sent to the operator terminal.
If the first character in the message is the character “<”, the next character is interpreted as a routing code to which the message is to be sent.

WAIT-RESPONSE = *NO / *YES
Specifies whether the task is to wait for a response from the operator.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
	64	EXC0090	Out of space

INFORM-PROGRAM

Send message to program (STXIT routine)

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job processing Program control
Domain:	JOB PROGRAM
Privileges:	STD-PROCESSING OPERATING TSOS HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The INFORM-PROGRAM command enables a nonprivileged user to send a message to an interrupted program in the user’s own interactive task.

The interrupted program is continued at the address defined in the STXIT macro for the event: “message to the program”. If the program does not contain a STXIT routine for this event, the INFORM-PROGRAM command is rejected and an error message is issued. The occurrence of the “message to the program” event is independent of a message that the user can additionally specify in the MSG operand.
A message specified in the MSG operand will, if specified in the STXIT macro, be transferred to an area in the program defined expressly for that purpose.

The STXIT mechanism is described in the “Executive Macros” manual [22].

Privileged functions

Systems support (TSOS or OPERATING privilege) can also use the command to send a message to a loaded program in a noninteractive task.
If the program does not have an STXIT routine for this event, the command is accepted but has no effect. If there is no program loaded in the task, the command is rejected.

Using this privileged function, systems support can among other things terminate specific database applications gracefully before system shutdown by informing them that the “message to the program” event has occurred. If the database has been programmed to process the event, it can takes steps to shut itself down.

Format

```
INFORM-PROGRAM                                              Alias: IFPG

 MSG = *NO / <c-string 1..64 with-low>
,JOB-IDENTIFICATION = *OWN / *TSN(...) / *MONJV(...)
   *TSN(...)
     |  TSN = <alphanum-name 1..4>
   *MONJV(...)
     |  MONJV = <filename 1..54 without-gen>
```

Operands

MSG = *NO / <c-string 1..64 with-low>
Text of the message to be sent to the program.
The default is *NO, which means that no message will be sent.
The length of the message is limited to 64 characters. Shorter messages have a null byte appended to the right to indicate end-of-text in the receive area. If no area for message transfer was defined in the STXIT macro, the specified message is ignored. If no message is specified although an area has been defined, a null byte is transferred.

JOB-IDENTIFICATION = *OWN / *TSN(...) / *MONJV(...)
Indicates the form of identification used for the task in which the target program is running. An interrupted program in the user’s interactive task (*OWN) or in some other noninteractive task (*TSN or *MONJV) is to be informed that the “message to the program” event has occurred. Additionally, the user can send a message comprising up to 64 characters to the program (see MSG operand).

JOB-IDENTIFICATION = *OWN
The user informs an interrupted program running in his own interactive task.

JOB-IDENTIFICATION = *TSN(...)
Only privileged users (OPERATING or TSOS privilege) are allowed to select this value.
The task in which the target program is running is identified by its TSN. The target must not be an interactive task.

> **TSN = <alphanum-name 1..4>**
> Task sequence number.

JOB-IDENTIFICATION = *MONJV(...)

Only privileged users (OPERATING or TSOS privilege) are allowed to select this value.

The task in which the target program is running is identified by its monitoring JV. The target must not be an interactive task.

MONJV = <filename 1..54>

Name of the job variable monitoring the job.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No errors
	64	EXC0920	Specified job ID invalid
	64	EXC0921	Command not permitted for interactive tasks
	64	EXC0922	No program loaded
	64	EXC0923	No STXIT interface defined for this event in specified program
	64	EXC0924	No STXIT routine defined for this event in specified program
	64	EXC0925	STXIT routine already activated
	64	EXC0090	Insufficient storage space

LIST-CATALOG-ENTRY

Write catalog entry to file

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Pubset and MRSCAT management
Domain:	MULTI-CATALOG-AND-PUBSET-MANAGEMENT
Privileges:	TSOS

Function

The LIST-CATALOG-ENTRY command is designed to facilitate diagnosis of errors in a catalog entry. Systems support can have the entire catalog entry, including the extents, written to file for a particular file or job variable.

The command can be entered from any system in the network for shared pubsets.

Format

LIST-CATALOG-ENTRY
NAME = <filename 1..54> ,**OBJECT** = ***FILE** / ***JV** ,**OUTPUT** = <filename 1..54>

Operands

NAME = <filename 1..54>
Name of the file or job variable whose catalog entry is to be output. The name must contain the catalog and user IDs.

OBJECT = *FILE / *JV
Specifies whether the command is to be executed for the catalog entry of a file or job variable.

OUTPUT = <filename 1..54>
Name of the file to which the catalog entry is to be written.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without error
	32	CMD0221	Internal error
	64	DMS0501	Catalog is not available in the system
	64	DMS0505	Error in computer communication
	64	DMS0512	Catalog is not known in the system
	64	DMS1343	A change of master is currently running for the pubset

Format of the output

A header contains the fully qualified file name, a flag for the file or job variable, the pubset type and the catalog name (important for SM pubset).
Subsequently the catalog entry is output in hexadecimal, printable form together with its extents, contiguous parts being output contiguously.

LIST-FILE-FROM-SNAPSET

Provide information about files on a Snapset

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE SNAPSET
Privileges:	STD-PROCESSING TSOS

Function

The LIST-FILE-FROM-SNAPSET command enables the user to obtain information about files which were saved on a Snapset when a pubset was backed up. The information relates to whether files can be restored (using the RESTORE-FILE-FROM-SNAPSET command). The associated pubset must be imported.

Nonprivileged users can obtain information about all files which they can access (as with SHOW-FILE-ATTRIBUTES, which supplies information from the current file catalog).

The Snapsets are temporarily not available if the SHC-OSD subsystem was not active when the pubset was imported. In this case the command is aborted with DMS0622. As soon as SHC-OSD is active, the Snapsets are subsequently activated when the SHOW-SNAPSET-CONFIGURATION command is called.

Privileged functions

Systems support (TSOS privilege) can obtain information on the files of all user IDs. Wildcards are not permitted in the user ID here.

Format

```
LIST-FILE-FROM-SNAPSET

 FILE-NAME = <filename 1..54 with-wild(80)>

,SNAPSET = *LATEST / *ALL / <name 1..1 with-low> / <integer -52..-1> / *INTERVAL(...)

   *INTERVAL(...)
      |   OLDEST = -52 / <integer -52..-1>
      |  ,NEWEST = -1 / <integer -52..-1>

,INFORMATION = *NAME-AND-SPACE / *ALL

,OUTPUT = list-poss(2): *SYSOUT / *SYSLST
```

Operands

FILE-NAME = <filename 1..54 with-wild(80)>
Selects the files which are to be listed. The files must satisfy the following requirements:
- They must be cataloged when the Snapset is created.
- The pubset on which they are cataloged must be imported locally.

Aliases may be specified. Individual file generations can be specified. When a file generation group is specified, the file generations are also output.

SNAPSET = *LATEST / *ALL / <name 1..1 with-low> / <integer -52..-1> / *INTERVAL(...)
Specifies the Snapset from which the file information is to be output.
Information about all existing Snapsets for a pubset can be obtained using the SHOW-SNAPSET-CONFIGURATION command.

SNAPSET = *LATEST
The information is to be output from the latest Snapset (i.e. from the most up-to-date pubset backup).

SNAPSET = *ALL
The information from all existing Snapsets is output.

SNAPSET = <name 1..1 with-low>
Specifies the Snapset explicitly by means of the Snapset ID. The maximum of 52 pubsets are distinguished by means of Snapset IDs specified which comprise letters from the 26 lowercase letters a to z and the 26 uppercase letters A to Z.

SNAPSET = <integer -52..-1>
Specifies the Snapset explicitly by means of the relative age. The value -1 specifies the latest Snapset.

SNAPSET = *INTERVAL(...)
Information on all Snapsets which lie in the specified age range is output:

> **OLDEST = -52 / <integer -52..-1>**
> Specifies the oldest Snapset; the range begins with this Snapset.
>
> **NEWEST = -1 / <integer -52..-1>**
> Specifies the newest Snapset; the range ends with this Snapset.

INFORMATION = *NAME-AND-SPACE / *ALL
Determines the type and scope of the information to be output.

INFORMATION = *NAME-AND-SPACE
For each cataloged file name one line is output which contains the file size in PAM pages and the file name. Work files (WORK-FILE=*YES) and migrated files as well as files on Net-Storage or on private disk are also flagged. Tape files are listed without file size or identification.

INFORMATION = *ALL
Two lines are output for each cataloged file name. The first line has the same format as with INFORMATION=*NAME-AND-SPACE. The second line contains the creation date, date of modification and the file status. The file status indicates whether the file was open in write mode when the backup took place (STATE=OPENED or CLOSED). In the case of files which cannot be restored (e.g. files with the backup frequency BACKUP-CLASS=E or special system files, work files, files on Net-Storage or on private disk), STATE=NOREST is displayed.

OUTPUT = list-poss(2): *SYSOUT / *SYSLST
Specifies whether the information is to be directed to SYSOUT (default) and/or SYSLST.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed without errors
1	0	CMD0001	Interrupt the output with the K2 key
2	0	DMS05B6	Incorrect conversion of UTC to LT
2	0	DMS0616	Volume set in SM pubset might not be available
	1	CMD0202	Too many operands, syntax error Guaranteed message: DMS0598
	1	CMD0202	Syntax error in Snapset specification Guaranteed message: DMS06F7
	1	CMD0202	Path name too long Guaranteed message: DMS05EE
	32	DMS05C7	Unexpected internal error in DMS
	64	DMS0501	Requested catalog not available
	64	DMS0512	Requested catalog not found
	64	DMS051B	Specified user ID not in home pubset Guaranteed message: DMS051B
	64	DMS051C	User does not have permission to access pubset Guaranteed message: DMS051C
	64	DMS051D	Logon and pubset passwords different
	64	DMS0535	Catalog entry not accessible
	64	DMS0585	Error detected when processing catalog or multiprocessor system
	64	DMS05FC	Specified user ID not in home pubset
	64	DMS0616	Volume set not accessible in SM pubset
	64	DMS0622	Snapset not available
	64	DMS0684	File does not exist
	64	DMS06CC	No file name corresponds to specified wildcard string
	130	DMS0585	Error detected when processing catalog or multiprocessor system
	130	DMS0594	Not enough virtual memory available

Examples

```
/list-file-from-snapset :kh:sysa*
----------------------------SNAPSET a----------------------------------------
         4 :KH:$TSOS.SYSACF.EDT.170
         4 :KH:$TSOS.SYSACF.LMS.034
         4 :KH:$TSOS.SYSACF.PMLOG.035
        12 :KH:$TSOS.SYSACF.SORT.079
        20 :KH:$TSOS.SYSAFR
```

```
/list-file-from-snapset :kh:sys.serslog*,inf=all
----------------------------SNAPSET a----------------------------------------
        56 :KH:$TSOS.SYS.SERSLOG.2011-08-25.001.01
  CRE-DATE=2011-08-25 11:56:55  CHANG-DATE=2011-08-25 12:04:52  STATE=CLOSED
        56 :KH:$TSOS.SYS.SERSLOG.2011-08-25.002.01
  CRE-DATE=2011-08-25 12:17:51  CHANG-DATE=2011-08-25 12:33:32  STATE=CLOSED
        56 :KH:$TSOS.SYS.SERSLOG.2011-08-25.003.01
  CRE-DATE=2011-08-25 13:15:38  CHANG-DATE=2011-08-25 14:01:39  STATE=CLOSED
        56 :KH:$TSOS.SYS.SERSLOG.2011-08-28.004.01
  CRE-DATE=2011-08-28 08:14:34  CHANG-DATE=2011-08-28 08:37:57  STATE=CLOSED
        64 :KH:$TSOS.SYS.SERSLOG.2011-08-28.005.01
  CRE-DATE=2011-08-28 08:44:31  CHANG-DATE=2011-08-28 08:44:31  STATE=OPENED
        56 :KH:$TSOS.SYS.SERSLOG.2011-08-28.006.01
  CRE-DATE=2011-08-28 09:49:23  CHANG-DATE=2011-08-28 10:17:44  STATE=CLOSED
```

```
/list-file-from-snapset :kh:ca*,snap=-2,inf=*all
----------------------------SNAPSET b----------------------------------------
         8 :KH:$TSOS.CAP
  CRE-DATE=2011-08-25 10:58:40  CHANG-DATE=2011-08-25 10:58:40  STATE=CLOSED
         4 :KH:$TSOS.CAT.SHC-OSD.060
  CRE-DATE=2011-08-25 13:47:36  CHANG-DATE=2011-08-25 13:47:36  STATE=CLOSED
         4 :KH:$TSOS.CAT.SYMAPI-D.062
  CRE-DATE=2011-08-25 13:26:29  CHANG-DATE=2011-08-25 13:26:29  STATE=CLOSED
```

```
/list-file-from-snapset :x:s*,inf=all
---------------------------SNAPSET a-----------------------------------------
       33 :X:$TSOS.SYSCAT.SNAPSET
 CRE-DATE=2011-09-14 08:32:23  CHANG-DATE=2011-09-14 08:32:24  STATE=OPENED
       33 :X:$TSOS.SYSDAT.APUB-PUBX00
 CRE-DATE=NONE                 CHANG-DATE=NONE                 STATE=NOREST
       24 :X:$TSOS.SYSSRPM
 CRE-DATE=2011-09-14 08:31:23  CHANG-DATE=2011-09-14 08:31:23  STATE=NOREST
       30 :X:$TSOS.SYSSRPM.BACKUP
 CRE-DATE=2011-09-14 08:32:25  CHANG-DATE=2011-09-14 08:32:25  STATE=CLOSED
```

```
/list-file-from-snapset :kh:$ulr.,snapset=*all
----------------------------SNAPSET c----------------------------------------
         0 :KH:$ULR.FGG (FGG)
         8 :KH:$ULR.FGG(*0001)
         8 :KH:$ULR.FGG(*0002)
```

```
        12 :KH:$ULR.SAM.EDT
        64 :KH:$ULR.SAM.WORK
----------------------------SNAPSET b----------------------------------------
         0 :KH:$ULR.FGG (FGG)
         8 :KH:$ULR.FGG(*0001)
         8 :KH:$ULR.FGG(*0002)
         8 :KH:$ULR.T.1
         8 :KH:$ULR.T.2

/list-file-from-snapset :kh:t.,snapset=*all
----------------------------SNAPSET c----------------------------------------
%  DMS06CC NO FILE CORRESPONDING TO SPECIFIED OPERANDS
----------------------------SNAPSET b----------------------------------------
         8 :KH:$ULR.T.1
         8 :KH:$ULR.T.2
```

LIST-JV-FROM-SNAPSET

Provide information about job variables on a Snapset

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Job variables
Domain:	JOB-VARIABLES SNAPSET
Privileges:	STD-PROCESSING TSOS

Function

The LIST-JV-FROM-SNAPSET command enables the user to obtain information about job variables which were saved on a Snapset when a pubset was backed up. The information relates to whether job variables can be restored (using the RESTORE-JV-FROM-SNAPSET command). The associated pubset must be imported.

Nonprivileged users can obtain information about all job variables which they can access (as with SHOW-JV-ATTRIBUTES, which supplies information from the current file catalog).

The Snapsets are temporarily not available if the SHC-OSD subsystem was not active when the pubset was imported. In this case the command is aborted with DMS0622. As soon as SHC-OSD is active, the Snapsets are subsequently activated when the SHOW-SNAPSET-CONFIGURATION command is called.

Privileged functions

Systems support (TSOS privilege) can obtain information on the job variables of all user IDs. Wildcards are not permitted in the user ID here.

Format

LIST-JV-FROM-SNAPSET
JV-NAME = <filename 1..54 without-gen-vers with-wild(80)>
,**SNAPSET** = ***LATEST** / ***ALL** / <name 1..1 with-low> / <integer -52..-1> / ***INTERVAL**(...)
***INTERVAL**(...)
│ **OLD**EST = **-52** / <integer -52..-1>
│ ,**NEW**EST = **-1** / <integer -52..-1>
,**INF**ORMATION = ***NAME-AND-SIZE** / ***ALL**
,**OUTPUT** = list-poss(2): ***SYSOUT** / ***SYSLST**

Operands

JV-NAME = <filename 1..54 without-gen-vers with-wild(80)>
Selects the job variables which are to be listed. The job variables must satisfy the following requirements:

- They must be cataloged when the Snapset is created.
- The pubset on which they are cataloged must be imported locally.

Aliases may be specified.

SNAPSET = *LATEST / *ALL / <name 1..1 with-low> / <integer -52..-1> / *INTERVAL(...)
Specifies the Snapset from which the file information is to be output.
Information about all existing Snapsets for a pubset can be obtained using the SHOW-SNAPSET-CONFIGURATION command.

SNAPSET = *LATEST
The information is to be output from the latest Snapset (i.e. from the most up-to-date pubset backup).

SNAPSET = *ALL
The information from all existing Snapsets is output.

SNAPSET = <name 1..1 with-low>
Specifies the Snapset explicitly by means of the Snapset ID. The maximum of 52 Snapsets for a pubset are distinguished by means of Snapset IDs specified which comprise letters from the 26 lowercase letters a to z and the 26 uppercase letters A to Z.

SNAPSET = <integer -52..-1>
Specifies the Snapset explicitly by means of the relative age. The value -1 specifies the latest Snapset.

SNAPSET = *INTERVAL(...)
Information on all Snapsets which lie in the specified age range is output:

> **OLDEST = -52 / <integer -52..-1>**
> Specifies the oldest Snapset; the range begins with this Snapset.
>
> **NEWEST = -1 / <integer -52..-1>**
> Specifies the newest Snapset; the range ends with this Snapset.

INFORMATION = *NAME-AND-SIZE / *ALL
Determines the type and scope of the information to be output.

INFORMATION = *NAME-AND-SIZE
For each cataloged job variable one line is output which contains the length of its value and its name.

INFORMATION = *ALL

Two lines are output for each cataloged job variable. The first line has the same format as with INFORMATION=*NAME-AND-SPACE. The second line contains the creation date and release date.

OUTPUT = list-poss(2): *SYSOUT / *SYSLST

Specifies whether the information is to be directed to SYSOUT (default) and/or SYSLST.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed without errors
1	0	CMD0001	Interrupt the output with the K2 key
2	0	DMS05B6	Incorrect conversion of UTC to LT
2	0	DMS0616	Volume set in SM pubset might not be available
	1	CMD0202	Syntax or semantic error Guaranteed messages: DMS0598, DMS06F7, DMS05DE, DMS05EE
	32	DMS05C7	Unexpected internal error in DMS
	64	DMS0501	Requested catalog not available
	64	DMS0512	Requested catalog not found
	64	DMS051B	Specified user ID not in home pubset Guaranteed message: DMS051B
	64	DMS051C	User does not have permission to access pubset
	64	DMS051D	Logon and pubset passwords different
	64	DMS0535	Catalog entry not accessible
	64	DMS0585	Error detected when processing catalog or multiprocessor system
	64	DMS05FC	User ID does not exist
	64	DMS0616	Volume set not accessible in SM pubset
	64	DMS0622	Snapset not available
	64	DMS0682	Error while accessing job variable (e.g. it does not exist); for more detailed information on the cause of the error, see the insert of the SYSOUT message Guaranteed message: DMS0682
	130	DMS0585	Error detected when processing catalog or multiprocessor system
	130	DMS0594	Not enough virtual memory available

Example

```
/list-jv-from-snapset :x:,inf=*all
----------------------------SNAPSET b--------------------------
         0 :X:$TSOS.JV.#AAA
  CRE-DATE=2011-09-14 09:23:40  EXPIR-DATE=2011-09-14 00:00:00
         0 :X:$TSOS.JV.TTT
  CRE-DATE=2011-09-14 09:23:35  EXPIR-DATE=2011-09-14 00:00:00
         0 :X:$TSOS.JV.1
  CRE-DATE=2011-09-14 09:23:24  EXPIR-DATE=2011-09-14 00:00:00
         0 :X:$TSOS.JV.2
  CRE-DATE=2011-09-14 09:23:28  EXPIR-DATE=2011-09-14 00:00:00
```

LIST-NET-DIRECTORIES

Display directories released for Net-Storage

Description status:	BS2000 OSD/BC V10.0A
Functional area:	Net-Storage administration
Domain:	DEVICE STORAGE-MANAGEMENT
Privileges:	TSOS OPERATING
Routing code:	G

Function

The LIST-NET-DIRECTORIES command provides information on all directories of a net server which are released for use as Net-Storage. The listed directories or also the directories beneath them can be connected to BS2000 as Net-Storage (see the MOUNT-NET-STORAGE command).

Fundamental information on the use of Net-Storage in BS2000 is provided in the “Introduction to System Administration” [14]. How to work with files on Net-Storage is described in the "Introductory Guide to DMS" [13].

Format

```
LIST-NET-DIRECTORIES

 SERVER = <composed-name 1..256 with-under> / <c-string 1..256 with-low> / *IP-ADDRESS(...)

    *IP-ADDRESS(...)
      |   IP-ADDRESS = <composed-name 7..15> / <c-string 2..39>

,CLIENT = <composed-name 1..8 with-under> / <c-string 1..8> / *DNS(...) / *IP-ADDRESS(...)

    *DNS(...)
      |   DNS-NAME = <c-string 1..256 with-low>

    *IP-ADDRESS(...)
      |   IP-ADDRESS = <composed-name 7..15> / <c-string 2..39>

,SELECT = *ALL / *MOUNTED / *NOT-MOUNTED
```

Operands

SERVER =
Determines the Net-Server whose released directories are to be listed.

SERVER = <composed-name 1..256 with-under> / <c-string 1..256 with-low>
Host name or fully qualified domain name of a net server.

SERVER = *IP-ADDRESS(...)
IP address of the net server.

> **IP-ADDRESS = <composed-name 7..15> / <c-string 2..39>**
> Specifies the IP address of the net server in IPv4 or IPv6 format.

CLIENT =
Specifies the net client for which directories are to be displayed. The Net-Storage (more precisely: the released directory) is mounted on the net client.

CLIENT = <composed-name 1..8 with-under> / <c-string 1..8>
Internal BCAM name of the net client.

CLIENT = *DNS(...)
Domain name of the net client.

> **DNS-NAME = <c-string 1..256 with-low>**
> Specifies the fully qualified domain name of the net client.

CLIENT = *IP-ADDRESS(...)
IP address of the net client.

IP-ADDRESS = <composed-name 7..15> / <c-string 2..39>
Specifies the IP address of the net client in IPv4 or IPv6 format.

SELECT = *ALL / *MOUNTED / *NOT-MOUNTED
Determines which of the released directories are displayed.

SELECT = *ALL
All directories are displayed which are released for the specified net client and are thus available for use as Net-Storage.

SELECT = *MOUNTED
All directories are displayed which are already mounted on the specified net client and connected to BS2000.

SELECT = *NOT-MOUNTED
All directories are displayed which have been released for the specified net client but have not yet been mounted and connected to BS2000.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed without error
	1	NKAN003	Syntax error in the input
	32	NKAN004	Error in command execution
	64	NKAN015	No relevant information available

Output format

Sample output

```
/list-net-directories server=*ip(172.17.65.103)),client=*ip(192.168.138.12)
-   NET-SERVER:  IPV4: 172.17.65.103 ──────────────────────────────────  (1)
-   NET-CLIENT:  IPV4: 192.168.138.12 ─────────────────────────────────  (2)
************************************************************************
- NET-DIRECTORY: /nas/100/fgqa/SQ200/test    (MOUNT-ID = 1) ───────────  (3)
************************************************************************
- NET-DIRECTORY: /nas/100/fgqa/HNC42    (MOUNT-ID = 2)
************************************************************************
- NET-DIRECTORY: /export/nas/100/fgqa    (NOT MOUNTED)
************************************************************************
- NET-DIRECTORY: /nas/100/fgqa    (NOT MOUNTED)
************************************************************************
- NET-DIRECTORY: /export    (NOT MOUNTED)
************************************************************************
```

(1) NET–SERVER displays the name or IP address of the net server as specified in the command. For an IP address, the IPV4 or IPV6 format is also displayed.

(2) NET–CLIENT displays the name or IP address of the net client as specified in the command. For an IP address, the IPV4 or IPV6 format is also displayed.

The following information is output in a separate information block for each released directory:

(3) NET–DIRECTORY displays the path name of the released directory and the status. The status can assume the following values:

- (NOT MOUNTED) The directory is not connected.
- (MOUNT–ID = <n>) If the directory is connected, the mount ID is displayed.

Output in S variable

Output information	Name of the S variable	T	Contents	Condition
Directory name of the Net-Storage released on the net server	var(*LIST).DIR(*LIST).NAME	S	<name 1..64> ”	
Mount ID of the Net-Storage (0: volume not connected)	var(*LIST).DIR(*LIST).MOUNT-ID	I	<integer>	
Name of the net server	var(*LIST).DIR(*LIST).SERVER.NAME [1]	S	<name 1..256> ”	
IP address of the net server in IPv4 format	var(*LIST).DIR(*LIST).SERVER.IPV4 [1]	S	<name 7..15> ”	
IP address of the net server in IPv6 format	var(*LIST).DIR(*LIST).SERVER.IPV6 [1]	S	<name 15..39> ”	
Name of the net client	var(*LIST).DIR(*LIST).CLIENT.NAME [1]	S	<name 1..256> ”	
DNS name of the net client	var(*LIST).DIR(*LIST).CLIENT.DNS [1]	S	<name 1..256> ”	
IP address of the net client in IPv4 format	var(*LIST).DIR(*LIST).CLIENT.IPV4 [1]	S	<name 7..15> ”	
IP address of the net client in IPv6 format	var(*LIST).DIR(*LIST).CLIENT.IPV6 [1]	S	<name 15..39> ”	

[1] Statusabhängige Ausgabe: Leerstring, wenn die Information nicht zur Verfügung steht.

LIST-NODE-FILES

List node files on Net-Storage volume

Description status:	BS2000 OSD/BC V10.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING HARDWARE-MAINTENANCE SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION
Routing code:	@

Function

The LIST-NODE-FILES command lists node files (files of the type NODE-FILE) which are located on a Net-Storage volume and can be imported with the IMPORT-NODE-FILE command. This comprises all node files with a BS2000-compliant name which are stored directly in the user-specific directory on the Net-Storage volume. Directories and files located below it are not taken into account.

The command supports structured output in S variables (see “Output in S variable” on page 3-493).

Privileged functions

Systems support (TSOS privilege) can have node files from all user-specific directories listed.

Format

LIST-NODE-FILES
VOLUME = <vsn 6..6>
,USER-DIRECTORY = *OWN / *ALL / <name 1..8>
,NODE-FILE-NAME = <filename 1..54 without-cat-gen-vers with-wild(80)>
,INFORMATION = *STD / *SUMMARY
,OUTPUT = *SYSOUT / list-poss(2): *SYSOUT / *SYSLST

Operands

VOLUME = <vsn 6..6>
VSN of the Net-Storage volumes whose node files are to be listed.

USER-DIRECTORY = *OWN / *ALL / <name 1..8 >
Directory of the user ID from which the node files are listed. The directory name is the same as the user ID. Only those files are listed which reside directly below this directory on the Net-Storage volume. Directories contained in the directory are ignored. Nonprivileged users can only obtain information on node files for their own user ID.

USER-DIRECTORY = *OWN
Lists node files from the directory of the user's own user ID.

USER-DIRECTORY = *ALL
This operand value is reserved for privileged users. Lists node files from all user directories of the Net-Storage volume.

USER-DIRECTORY = <name 1..8 >
Lists node files from the specified user directory. The name of the directory corresponds to the user ID. Nonprivileged users can only specify their own user ID.

NODE-FILE-NAME = <filename 1..54 without-cat-gen-vers with-wild(80)>
File names of the node files which are to be listed. Only files with BS2000-compliant file names are listed, i.e. node files which can be imported.

INFORMATION = *STD / *SUMMARY
Controls the scope of the information to be output.

INFORMATION = *STD
Outputs a line of information with the most important file attributes in summarized form for each node file. The line of information contains brief information about the file access rights, file size, change date and file name. Up to 74 characters are displayed.

INFORMATION = *SUMMARY
For each specified user directory, outputs the number of node files and the total number of bytes which these occupy in this directory.

OUTPUT =
Specifies where the information is to be output.

OUTPUT = *SYSOUT
The output is directed to SYSOUT.

OUTPUT = *SYSLST
Output is written to SYSLST.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
	64	CMD0102	Interrupted by K2 key
	64	CMD0216	Privilege errors
	64	DMS051B	Requested user ID not in pubset
	64	DMS0594	Not enough virtual memory available
		DMS0586	It is not possible to access or reserve a volume at present
	130	DMS0586	It is not possible to access or reserve a volume at present
		DMS05C7	DMS internal error
		DMS0607	Error when generating S variables
	64	DMS0640	ONETSTOR reports communication problem with net client
	64	DMS0643	Net client reports access error
	64	DMS0644	Net client reports internal error
	64	DMS06CC	No file matches selection criteria

Output format

Sample output

```
/list-node-files volume=ck68@0,user-dir=*own,node-file-name=*
 CK68@0/USER1/ ────────────────────────────────────────────────────────  (1)
 -rw-------   10k Sep 12 14:27 NNDF1 ──────────────────────────────────  (2)
 -rw-------    2M May 21 01:10 NNDF2
 -rw-------    1M May 21 01:15 NNDF3
 -rw-------    2M May 21 01:20 NNDF4
```

Explanation of the output

The following information is output for each specified user-specific directory:

(1) Path name of the user-specific directory containing the displayed node files.

(2) The following file attributes are displayed for each file of the user-specific directory:

– The read, write and execute access rights for the file owner, the user's group and the other users in UNIX format: r=read permission, w=write permission, x=execute permission and "-" when no authorization is available.

– The file size in bytes (without suffix), in KB (suffix "k"), in MB (suffix "M") or in GB (suffix "G").

– Date and time of the last change.

– Name of the file.

Output in S variable

The INFORMATION operand identifies the S variables which are to be created.

Output information	Name of the S variable	T	Contents	Condition
Name of the file	var(*LIST).F-NAME	S	” <filename>	INF=*STD
File size, depending on size in bytes (without suffix), in KB (suffix "k"), in MB (suffix "M") or in GB (suffix "G")	var(*LIST).F-SIZE	I	” <integer 0..1023> <integer 0..1023>k <integer 0..1023>M <integer 0..1023>G	INF=*STD
Execute permission for GROUP	var(*LIST).GROUP.EXEC	S	” NO / YES	INF=*STD
Read permission for GROUP	var(*LIST).GROUP.READ	S	” NO / YES	INF=*STD
Write permission for GROUP	var(*LIST).GROUP.WRITE	S	” NO / YES	INF=*STD
Date of last file modification	var(*LIST).LAST-CHA-DATE	S	” <yyyy-mm-dd>	INF=*STD
Time of last file modification	var(*LIST).LAST-CHA-TIME	S	” <hh:mm>	INF=*STD
Number of node files in the directory	var(*LIST).NUM-OF-F	I	” <integer 0..1023>	INF=*SUM
Execute permission for OTHERS	var(*LIST).OTHERS.EXEC	S	” NO / YES	INF=*STD
Read permission for OTHERS	var(*LIST).OTHERS.READ	S	” NO / YES	INF=*STD
Write permission for OTHERS	var(*LIST).OTHERS.WRITE	S	” NO / YES	INF=*STD
Execute permission for OWNER	var(*LIST).OWNER.EXEC	S	” NO / YES	INF=*STD
Read permission for OWNER	var(*LIST).OWNER.READ	S	” NO / YES	INF=*STD
Write permission for OWNER	var(*LIST).OWNER.WRITE	S	” NO / YES	INF=*STD
UNIX file type	var(*LIST).UNIX-F-TYPE	S	” FILE SOFTLINK	INF=*STD
UNIX path name: directory of the node file	var(*LIST).UNIX-PATHNAME	S	” <filename>	INF=*STD/ *SUM

(Part 1 of 2)

Output information	Name of the S variable	T	Contents	Condition
Sum of the storage space of the node files in the directory	var(*LIST).USED-BYTES	I	’’ <integer 0..1023> <integer 0..1023>k <integer 0..1023>M <integer 0..1023>G	INF=*SUM
user ID	var(*LIST).USER-ID	S	’’ <userid>	INF=*STD

(Part 2 of 2)

LOAD-ALIAS-CATALOG

Load entries from file into alias catalog

Description status:	ACS V19.0A
Functional area:	File processing
Domain:	FILE
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The LOAD-ALIAS-CATALOG command enables the user to load an alias catalog file (also called an AC file) into the task-local alias catalog. If no alias catalog exists for the task, an alias catalog is created for it automatically the first time an AC file is loaded.

The AC file contains the assignments of aliases to actual file or job variable names. When the alias substitution function is active, ACS substitutes the actual file or job variable name for the aliases that are specified by the user in commands. The alias substitution function has priority over the prefix insertion function and prevents the insertion of prefixes.

The ACS administrator can provide the user with one or more AC system files containing predefined alias definitions for specific software configurations (e.g. compilers, utilities, etc.). The names of the available AC system files can be displayed by means of the SHOW-ACS-SYSTEM-FILES command. The name of the default AC system file is identified by the character “>” in the output of this command.

If desired, the user may also load a personally created AC file containing user-specific alias definitions (see also the STORE-ALIAS-CATALOG command).

The alias catalog that is local to the task can be extended by loading additional AC files (see the MODE=*MERGE operand) or by adding individual alias definitions to it with the ADD-ALIAS-CATALOG-ENTRY command.

Entries in the alias catalog that is local to the task can be displayed by the user with the SHOW-ALIAS-CATALOG-ENTRY command.

The commands HOLD-ALIAS-CATALOG and RESUME-ALIAS-CATALOG can be used to halt and resume alias substitution, respectively.

The user can save the current alias catalog in an AC file by using the STORE-ALIAS-CATALOG command (only the user entries are saved), or delete it with the PURGE-ALIAS-CATALOG command.

The alias catalog is automatically deleted at the end of the task.

Loading errors

Syntactically incorrect or invalid AC entries cannot be loaded into the alias catalog and are not logged, but message ACS0003 is output. **Five** such errors will cause the load operation to abort.
The existing alias catalog, if any, will be restored to its original state if loading is aborted.

Format

```
LOAD-ALIAS-CATALOG                                                    Alias: LDAC

ALIAS-CATALOG-ID = *STD / <composed-name 1..20> / *OWN(...)
  *OWN(...)
    | FROM-FILE = <filename 1..54 without-gen-vers>
,MODE = *MERGE / *REPLACE
,SELECT = *ALL-ENTRIES / *BY-ALIAS-NAME(...)
  *BY-ALIAS-NAME(...)
    | USER-IDENTIFICATION = *ANY / *OWN / *NONE / *DEFAULT-USERID
,SUCCESS-MSG = *STD / *YES / *NO
```

Operands

ALIAS-CATALOG-ID =*STD / <composed-name 1..20> / *OWN(...)
Names the AC file from which the alias catalog is to be loaded. The user must have execution rights for the AC file.

ALIAS-CATALOG-ID = *STD
The alias catalog is loaded from the default AC system file. This file is identified in the output of the SHOW-AC-SYSTEM-FILES command by the character “>”.

ALIAS-CATALOG-ID = <composed-name 1..20>
Identification of an AC system file. The available AC system files can be determined by the user with the SHOW-AC-SYSTEM-FILES command.

ALIAS-CATALOG-ID = *OWN(...)
The alias catalog is loaded from the AC file that is specified by the user.

> **FROM-FILE = <filename 1..54 without-gen-vers>**
> File name of the AC file.

MODE = *MERGE / *REPLACE
Specifies whether the alias catalog to be generated is to overwrite or extend an existing alias catalog.

MODE = <u>*MERGE</u>
The existing alias catalog, if any, is retained, and entries from the specified AC system file or the user-specific AC file are appended to it. Existing entries with the same alias are overwritten.

MODE = *REPLACE
Previous entries in the alias catalog are deleted. The new alias catalog will only have the entries contained in the AC system file or AC file.

SELECT = <u>*ALL-ENTRIES</u> / *BY-ALIAS-NAME(...)
Determines whether all entries in the specified AC system file or AC file are to be loaded into the task-specific alias catalog.

SELECT = <u>*ALL-ENTRIES</u>
All entries are loaded into the alias catalog of the task.

SELECT = *BY-ALIAS-NAME(...)
Only the entries specified in the selection which follows are loaded into the task-local alias catalog. The selection criterion used is the name of the alias.

USER-IDENTIFICATION =
The entries to be loaded are selected on the basis of the user ID that was specified in the alias.

USER-IDENTIFICATION = <u>*ANY</u>
The user ID does not serve as a selection criterion (equivalent to SELECT=*ALL-ENTRIES).

USER-IDENTIFICATION = *OWN
Only the entries which have the user's own user ID in the alias are loaded.

USER-IDENTIFICATION = *NONE
Only the entries without a user ID in the alias are loaded.

USER-IDENTIFICATION = *DEFAULT-USERID
Only the entries with the system default ID in the alias are loaded. The alias *$file* is equivalent to *$.file* if *file* does not contain a period.

SUCCESS-MSG = <u>*STD</u> / *YES / *NO
Specifies whether a message is to be output after the AC system file or AC file is successfully loaded.

SUCCESS-MSG = <u>*STD</u>
The success message depends on the global settings or the setting which are local to the task (see the MODIFY-ACS-OPTIONS command). The currently applicable settings can be determined from the output of the SHOW-ACS-OPTIONS command.

SUCCESS-MSG = *YES
Message ACS0001 is output after successful loading.

SUCCESS-MSG = *NO
Message ACS0001 is suppressed.

Return codes

(SC2)	SC1	Maincode	Meaning/Guaranteed messages
	0	CMD0001	Command executed normally Guaranteed message: ACS0001 (if logging is enabled; see SUCCESS-MSG operand)
1	0	CMD0001	No action (e.g. no default ACSF)
2	0	ACS0003	Warning: invalid record in ACF ignored
	32	CMD0221	Internal error
	64	ACS0004	Aborted due to invalid records in ACF
	64	ACS0011	Error when accessing file
	64	ACS0012	ACSF not found
	64	ACS0029	Command not permitted
	128	ACS0018	ACS is not available
	130	ACS0036	Resource bottleneck

Example

Creating the alias catalog and loading entries from the AC file

```
/add-alias-cat alias=bsp,file=max.file.3                      ──────────────────────────────   (1)
%  ACS0020 ALIAS CATALOG ACTIVATED
/add-alias-cat alias=max,file=max.file.3                      ──────────────────────────────   (2)
/add-alias-cat alias=sf.sdf,file=$tsos.sys.sdf.system.syntax,range=*file    ───────   (3)
/show-alias                                                   ──────────────────────────────   (4)
%       ALIAS FILE NAME          ->     FILE NAME
%UB : BSP                        -> MAX.FILE.3
%UB : MAX                        -> MAX.FILE.3
%UF : SF.SDF                     -> $TSOS.SYS.SDF.SYSTEM.SYNTAX
%  ACS0037 NUMBER OF ALIAS CATALOG ENTRIES: 3 (FOR SYSTEM: 0, FOR USER: 3)
/load-alias-cat alias-cat=*own(acs.a1),mode=*replace          ────────────────────────   (5)
%  ACS0001 ALIAS CATALOG 'USER1AC' LOADED (DESCRIPTOR='V0001A', DATE=2011-02-11, NUMBER
OF ENTRIES=3)
/show-alias                                                   ──────────────────────────────   (6)
%       ALIAS FILE NAME          ->     FILE NAME
%UF : ASS                        -> $.ASSEMBH
%UB : BSP                        -> LST.BSP.2
%UF : SF.SDF                     -> $TSOS.SYS.SDF.SYSTEM.SYNTAX
%  ACS0037 NUMBER OF ALIAS CATALOG ENTRIES: 3 (FOR SYSTEM: 0, FOR USER: 3)

/rem-alias alias=ass                                          ──────────────────────────────   (7)
/add-alias alias=max,file=max.file.3                          ──────────────────────────────   (8)
/show-alias                                                   ──────────────────────────────   (9)
%       ALIAS FILE NAME          ->     FILE NAME
%UB : BSP                        -> LST.BSP.2
%UB : MAX                        -> MAX.FILE.3
%UF : SF.SDF                     -> $TSOS.SYS.SDF.SYSTEM.SYNTAX
%  ACS0037 NUMBER OF ALIAS CATALOG ENTRIES: 3 (FOR SYSTEM: 0, FOR USER: 3)
/load-alias-cat alias=*own(acs.a1),mode=*merge                 ─────────────────────────────  (10)
%  ACS0001 ALIAS CATALOG 'USER1AC' LOADED (DESCRIPTOR='V0001A', DATE=2011-02-11, NUMBER
OF ENTRIES=3)
```

```
/show-alias                                         ─────────────────────────────── (11)
%       ALIAS FILE NAME         ->     FILE NAME
%UB : ASS                       -> $.ASSEMBH
%UB : BSP                       -> LST.BSP.2
%UB : MAX                       -> MAX.FILE.3
%UF : SF.SDF                    -> $TSOS.SYS.SDF.SYSTEM.SYNTAX
%  ACS0037 NUMBER OF ALIAS CATALOG ENTRIES: 4 (FOR SYSTEM: 0, FOR USER: 4)
/purge-alias-cat                                    ─────────────────────────────── (12)
/show-alias                                         ─────────────────────────────── (13)
%  ACS0017 ALIAS CATALOG NOT AVAILABLE. COMMAND REJECTED
```

(1) The command ADD-ALIAS-CATALOG-ENTRY creates an entry in the task-local alias catalog to define the alias *EXP* for the file *MAX.FILE.3*. Since no alias catalog had been activated for the task, the alias catalog in question is activated in the process.

(2) The command ADD-ALIAS-CATALOG-ENTRY creates an entry in the task-local alias catalog to define the alias *MAX* for the file *MAX.FILE.3* (thus defining a second alias).

(3) The command ADD-ALIAS-CATALOG-ENTRY creates an entry in the task-local alias catalog to define the alias *SF.SDF* for the file *$TSOS.SYS.SDF.SYSTEM.SYNTAX* (because RANGE=*FILE the entry here applies only for files).

(4) Output of the entries contained in the alias catalog.

(5) The entries of the AC file *ACS.A1* are loaded into the alias catalog, and all previously existing entries are removed (`MODE=REPLACE`) in the process.

(6) The output of the SHOW-ALIAS-CATALOG-ENTRY command shows the three user entries that were saved in the AC file *ACS.A1* by means of the STORE-ALIAS-CATALOG-ENTRY command. All entries made earlier were removed or replaced by new entries for the same alias. For example, the alias *EXP* now addresses the file *LST.EXP.2*.

(7) Deletes the entry with the alias *ASS*.

(8) Adds an entry with the alias *MAX* for the file *MAX.FILE.3*.

(9) Output of the current entries in the alias catalog.

(10) The entries of the AC file *ACS.A1* are loaded into the alias catalog. All previously made entries are to be retained in this case (`MODE=MERGE`), but entries with the same alias are to be overwritten.
The same result could be obtained by the user with the ADD-ALIAS-CATALOG-ENTRY commands for all entries of the alias catalog.

(11) Output of the current entries in the alias catalog.

(12) The task-local alias catalog is deleted by using the PURGE-ALIAS-CATALOG command. This also ends the ACS substitution function.

(13) The command SHOW-ALIAS-CATALOG-ENTY indicates that no task-local alias catalog is loaded.

For further examples, see the ADD-ALIAS-CATALOG-ENTRY and HOLD-ALIAS-SUBSTITUTION commands.

LOAD-EXECUTABLE-PROGRAM

Load executable program (LLM, load, object module)

Description status:	BLSSERV V2.8A
Functional area:	Program control System control and optimization
Domain:	PROGRAM
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

Function

The LOAD-EXECUTABLE-PROGRAM command has two functions:

– It calls the dynamic binder loader DBL and defines the primary input. DBL links LLMs or OMs into a load unit, which it then loads into main memory. The DBL loads an LLM which has already been linked by the BINDER into a load unit and stored in a PAM file (PAM-LLM) only in main memory.

– It calls the static loader ELDE. A program (load module) previously linked by the TSOSLNK linkage editor is loaded by ELDE into main memory.

The loaded load unit or program is not started until a RESUME-PROGRAM command is executed.

If the user wants to load *and* start the load unit or program, the START-EXECUTABLE-PROGRAM command can be used instead of the LOAD-EXECUTABLE-PROGRAM and RESUME-PROGRAM commands.

In the case of operands which have *DBL-PARAMETERS as their default value, the MODIFY-DBL-DEFAULTS command can be used to modify the default DBL parameters which apply when the task is started. The parameters can be configured separately for calls to the LOAD- and START-EXECUTABLE-PROGRAM commands on the one hand and for the BIND macro call on the other. The SHOW-DBL-DEFAULTS command lists the values which are currently set. The RESET-DBL-DEFAULTS command can be used to return the settings to the DBL defaults.

i Structure-implicit notation is guaranteed for the PROGRAM-MODE, REP-FILE and ALTERNATE-LIBRARIES operands (see section “Abbreviation of names” on page 1-54).

Restrictions

Users with SAT-FILE-EVALUATION, SAT-FILE-MANAGEMENT and SECURITY-ADMINISTRATION privileges can only use the command in procedures.

Format

(Part 1 of 2)

```
LOAD-EXECUTABLE-PROGRAM                                                      Alias: LDX

 FROM-FILE = <filename 1..54 without-gen> / *OMF / [*LIBRARY-ELEMENT](...)

   [*LIBRARY-ELEMENT](...)

      |  LIBRARY = *DBL-DEFAULT / *BLSLIB / <filename 1..54 without-gen> / *LINK(...)
      |
      |     *LINK(...)
      |     |  LINK-NAME = <structured-name 1..8>
      |
      | ,ELEMENT-OR-SYMBOL = <composed-name 1..64 with-under>(...) / <c-string 1..32 with-low symbol>
      |     <composed-name 1..64 with-under>(...)
      |     |  VERSION = *HIGHEST-EXISTING / <composed-name 1..24 with-under>
      |     | ,NAME-SCOPE = *ELEMENT / *STD / *SYMBOL
      |
      | ,TYPE = (L,C,R) / list-poss(3): L / C / R

,PROGRAM-PARAMETERS = *NONE / <c-string 1..1800 with-low>

,DBL-PARAMETERS = *STD / [*PARAMETERS](...)

   [*PARAMETERS](...)

      |  LOADING = [*PARAMETERS](...)
      |     [*PARAMETERS](...)
      |     |  PROGRAM-MODE = *DBL-DEFAULT / *ANY / *24
      |     |
      |     | ,LOAD-INFORMATION = *DBL-DEFAULT / *DEFINITIONS / *MAP / *NONE /
      |     |                     *REFERENCES
      |     |
      |     | ,REP-FILE = *DBL-DEFAULT / *NONE / <filename 1..54 without-gen>
      |     |
      |     | ,IGNORE-ATTRIBUTES = *DBL-DEFAULT / *NONE / *READ-ONLY
      |     |
      |     | ,AMODE-CHECK = *DBL-DEFAULT / *STD / *ADVANCED
```

(Part 2 of 2)

```
   | ,RESOLUTION = [*PARAMETERS](...)
   |    [*PARAMETERS](...)
   |     |  SHARE-SCOPE = *DBL-DEFAULT / *SYSTEM-MEMORY / *NONE / *ALL /
   |     |                              *MEMORY-POOL(...)
   |     |     *MEMORY-POOL(...)
   |     |        |  SCOPE = *ALL / *USER-ID / *USER-GROUP / *HOST-SYSTEM
   |     |
   |     | ,PROGRAM-VERSION = *DBL-DEFAULT / *STD / *BLANK /
   |     |                        <composed-name 1..24 with-under>
   |     | ,ALTERNATE-LIBRARIES = *DBL-DEFAULT / *NONE / list-poss(2): *TASKLIB / *BLSLIB##
   |     |
   |     | ,AUTOLINK = *DBL-DEFAULT / *YES / *NO / *ALTERNATE-LIBRARIES
   |
   | ,ERROR-PROCESSING = [*PARAMETERS](...)
   |    [*PARAMETERS](...)
   |     |  NAME-COLLISION = *DBL-DEFAULT / *STD / *ABORT
   |     |
   |     | ,UNRESOLVED-EXTRNS = *DBL-DEFAULT / *STD / *DELAY / *ABORT
   |     |
   |     | ,ERROR-EXIT = *DBL-DEFAULT / *NONE / <x-string 1..8>
   |
   | ,REPORTING = [*PARAMETERS](...)
   |    [*PARAMETERS](...)
   |     |  MESSAGE-CONTROL = *DBL-DEFAULT / *INFORMATION / *WARNING / *ERROR /
   |     |                        *NONE
   |     |
   |     | ,PROGRAM-MAP = *DBL-DEFAULT / *NONE / *SYSLST(...) / *SYSOUT / *BOTH(...)
   |     |     *SYSLST(...)
   |     |        |  SYSLST-NUMBER = *STD / <integer 1..99>
   |     |     *BOTH(...)
   |     |        |  SYSLST-NUMBER = *STD / <integer 1..99>
,MONJV = *NONE / <filename 1..54 without-gen-vers>

,CPU-LIMIT = *JOB-REST / <integer 1..32767 seconds>

,TEST-OPTIONS = *DBL-DEFAULT / *NONE / *AID

,RESIDENT-PAGES = [*PARAMETERS] (...)

   [*PARAMETERS](...)
   |  MINIMUM = *STD / <integer 0..32767 4Kbyte>
   | ,MAXIMUM = *STD / <integer 0..32767 4Kbyte>

,VIRTUAL-PAGES = *STD / <integer 0..32767 4Kbyte>
```

Operands

FROM-FILE =

Determines the input source from which the program is to be loaded.

FROM-FILE = <filename 1..54 without-gen>
Name of the PAM file that contains a load module or PAM-LLM.

i If the file/library name is specified without a catalog/user ID and if it is not cataloged in the user ID, the system tries to access a file or library of the same name in the system default ID. (For information on this “secondary read” function see the “Introductory Guide to DMS” [13].)

FROM-FILE = *OMF
The input source is the EAM object module file. This contains object modules only.

FROM-FILE = *LIBRARY-ELEMENT(...)
The input source is a program library from which modules are retrieved. Either the dynamic binder loader DBL or the static loader ELDE is called depending on the element type.

LIBRARY =
Specifies the input source from which modules (LLMs, load modules or object modules) are to be retrieved.
The input source must generally be a program library. If the input source is an object module library (OML) then the following conditions must be fulfilled:

- Only object modules (OM, TYPE=R) may be loaded
- A value other than the default value ELEMENT must be specified for the NAME-SCOPE operand.

Whether or not other libraries are searched depends on the specification in the ALTERNATE-LIBRARIES operand (see page 3-511).

LIBRARY = *DBL-DEFAULT
The setting in the last MODIFY-DBL-DEFAULTS command applies. *BLSLIB is set by default on task start.

LIBRARY = *BLSLIB
The input source is the **standard library** with the file link name BLSLIB.

LIBRARY = <filename 1..54 without-gen>
File name of a library serving as input source.

LIBRARY = *LINK(...)
File link name of the library serving as input source.

LINK-NAME = <structured-name 1..8>
File link name of the library.

ELEMENT-OR-SYMBOL =
Specifies the modules that are to be retrieved from the specified input source.

In the case of an object module, the start address is dependent on the specifications in the END record as follows:

– If a start address is specified in the END record then this address is used.
– If no start address is specified in the END record then the address of the first CSECT in the module is used.
– If an external reference (EXTRN or WXTRN) is specified in the END record then the module is started at the address of the symbol that resolves this reference. If no symbol that resolves the reference is found then
  – in the case of an unresolved EXTRN, loading is aborted,
  – in the case of an unresolved WXTRN, the module is started at the address of the first CSECT.

An LLM is started at the address that was taken over by the BINDER when the LLM was stored.

ELEMENT-OR-SYMBOL = <composed-name 1..64 with-under>(...)
Name of the object module (element type R), load module (element type C) or LLM (element type L). The possible element types are specified in the TYPE operand. Depending on the specification in the NAME-SCOPE operand, a CSECT name, ENTRY name or element name from a module may be specified.

VERSION =
Specifies the element version.

VERSION = <u>*HIGHEST-EXISTING</u>
The default value for the highest element version in program libraries is taken over (see the “LMS” manual [21]).

VERSION = <composed-name 1..24 with-under>
Explicit specification of the element version.

NAME-SCOPE =
Specifies whether the name entered in ELEMENT-OR-SYMBOL can be an element name or also a CSECT or ENTRY name.

NAME-SCOPE = <u>*ELEMENT</u>
Only elements in PLAM libraries are searched for.

NAME-SCOPE = *SYMBOL
Only CSECT or ENTRY names are searched for. If the specified name is longer than 32 characters, the command is rejected.
For the search order, see section “Linkage process” in the “BLSSERV” User Guide [3].

NAME-SCOPE = *STD
Both element names (with the exception of type C) and CSECT/ENTRY names are searched for. If the specified name is longer than 32 characters, the command is rejected.
For the search order, see section “Linkage process” in the “BLSSERV” User Guide [3].

ELEMENT-OR-SYMBOL = <c-string 1..32 with-low *symbol*>
Specifies a symbol as a C string (pay attention to uppercase/lowercase). For the search order, see section “Linkage process” in the “BLSSERV” User Guide [3].

TYPE = (L,C,R) / list-poss(3): L / C / R
Specifies the element type. When a list is specified, the order of the specified element types also determines the search order. The default setting (L,C,R) specifies that the element is first searched for in the LLMs, then in the load modules and finally in the OMs. In the case of NAME-SCOPE=*SYMBOL/*STD, the specification of element type C is ignored.
The following applies for element names > 41 characters:

- If element type C is specified as an individual value, the command is rejected.
- If multiple element types are specified, element type C is ignored.

The operand is only evaluated for the input source (LIBRARY), not for alternate libraries for which the order L,R applies.

PROGRAM-PARAMETERS = *NONE / <c-string 1..1800 with-low>
Program parameters which read in the called program from the S variable SYSPARAM or via the C function *getopt*.
The default value is *NONE, i.e. no parameters are passed to the program. In such cases, an existing SYSPARAM is deleted.

DBL-PARAMETERS = *STD / *PARAMETERS(...)
This operand is not evaluated for load modules (element type C).
Designates all the parameters that are set by default in DBL or whose default setting can be modified using the MODIFY-DBL-DEFAULTS command. *STD is set by default, i.e. the command is called with the current DBL parameters. It is possible to declare explicit values for the command call by specifying *PARAMETERS(...).

DBL-PARAMETERS = *PARAMETERS(...)
Explicit specification of the DBL parameters for this command call.

LOADING = *PARAMETERS(...)
Specifies the parameters for the load operation.

PROGRAM-MODE=
Specifies in which part of the address space (above or below the 16-Mbyte boundary) the modules of the load unit are to be loaded. For the addressing mode, see the “BLSSERV” User Guide [3].

PROGRAM-MODE = *DBL-DEFAULT
The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is 24.

PROGRAM-MODE = *ANY
The modules of the load unit may be loaded above or below the 16-Mbyte boundary.

PROGRAM-MODE = *24
The entire load unit is loaded below the 16-Mbyte boundary.

LOAD-INFORMATION =
Specifies the loading information for the load unit.

LOAD-INFORMATION = *DBL-DEFAULT
The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *DEFINITIONS.

LOAD-INFORMATION = *DEFINITIONS
Loads an ESD containing the program definitions of all the modules in the load unit. Program definitions include program sections (CSECTs), entry points (ENTRYs), COMMON areas and external dummy sections (XDSECs-D).

LOAD-INFORMATION = *REFERENCES
Loads an ESD containing not only the program definitions, but also the satisfied references of all the modules in the load unit.
References include external references (EXTRNs), V-type constants, conditional external references (WXTRNs) and external dummy sections (XDSECs-R).

LOAD-INFORMATION = *MAP
Only one ESD required for the generation of the DBL list is *temporarily* loaded. The ESD is unloaded as soon as the DBL list has been generated.

LOAD-INFORMATION = *NONE
The external symbols dictionary (ESD) is not loaded with LLMs and is only loaded temporarily with OMs.

- With LLMs, the ESD is not read at all and consequently is also not used to resolve references within the generated load unit.
- With OMs, the ESD is read and is temporarily available during the loading process.

If external references need to be resolved between LLMs and OMs of the generated load unit, the minimum specification is *DEFINITIONS or *MAP.

REP-FILE = *DBL-DEFAULT / *NONE / <filename 1..54 without-gen>
Specifies whether REP records are to be applied to the modules of a load unit.
Default: The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *NONE, i.e. no REP records are used.
If the user specifies a file name, the DBL attempts to process REP records from this file. The REP records must have the standard format that is processed by the RMS utility routine (see the “Utility Routines” manual [9]). Should an error occur during REP record processing, a message (warning) is output, the invalid REP record is skipped and REP processing is then continued.

IGNORE-ATTRIBUTES =
Specifies which CSECT attributes are ignored on loading.

IGNORE-ATTRIBUTES = *DBL-DEFAULT
The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *NONE.

IGNORE-ATTRIBUTES = *NONE
No CSECT attributes are ignored on loading.

IGNORE-ATTRIBUTES = *READ-ONLY
The CSECT attribute READ-ONLY is ignored on loading. The CSECT is loaded into a readable/writable main memory page. This enables breakpoints to be set when testing with AID, for example.

AMODE-CHECK = *DBL-DEFAULT / *STD / *ADVANCED
Determines whether additional checks of the addressing mode should take place during loading.

AMODE-CHECK = *DBL-DEFAULT
The setting of the last MODIFY-DBL-DEFAULTS command applies. When the task begins, *STD is the default.

AMODE-CHECK = *STD
Only the checks compatible with BLSSERV < V2.5 are executed.

AMODE-CHECK = *ADVANCED
The same checks as for AMODE-CHECK = *STD are performed.
During loading a check is also performed to see whether inconsistencies can occur while resolving external references because of the load unit’s addressing mode.

RESOLUTION = *PARAMETERS(...)
Specifies how external references are to be handled.

SHARE-SCOPE =
*This specification is ignored for a PAM-LLM (SHARE-SCOPE=*NONE applies implicitly).*
Specifies whether, and if so, which portion of the shared code (see section "Sharable programs (shared code)" in the "BLSSERV" User Guide [3]) is included in the search for modules of the load unit and for unresolved external references.
On the system level, system administration can make available frequently used programs or modules to users in class 4/5 memory (as nonprivileged subsystems). The nonprivileged user can also make available programs or modules to users in class 6 memory (in a memory pool).
If a user uses modules from the available shared code, these modules need not be created in the task's class 6 memory. The references to modules already loaded as shared code suffice.
The shared code is searched in the following order:

1. Search of the shared code in class 6 memory (Common Memory Pools). If DBL finds the symbol it is searching for, it passes the load address, connects the task to the memory pool and terminates the load operation.
2. Search of the system's shared code, i.e. search of all nonprivileged subsystems that have been loaded into class 3/4 and class 5 memory. If DBL finds the symbol it is searching for, it passes the load address, connects the task to the memory pool and terminates the load operation.

SHARE-SCOPE = *DBL-DEFAULT
The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *SYSTEM-MEMORY.

SHARE-SCOPE = *SYSTEM-MEMORY
Only the shared code made available in class 3/4 and class 5 memory is included in the search.

SHARE-SCOPE = *NONE
Shared code is not included in the search.

SHARE-SCOPE = *ALL
Both the shared code made available in class 3/4 and class 5 memory (system memory) and that made available in class 6 memory (memory pools) is included in the search.

SHARE-SCOPE = *MEMORY-POOL(...)
The user shared code available in class 6 memory is included in the search, i.e. memory pools with the specified scope are included.

SCOPE =
Scope of the memory pools to be included.

SCOPE = *ALL
All memory pools which the task can access are included.

SCOPE = *USER-ID
Only memory pools which were set up for the user ID of the calling task are included, i.e. memory pools which are shareable for all tasks under this user ID.

SCOPE = *USER-GROUP
Only memory pools which were set up for the user group to which the calling task belongs are included, i.e. memory pools which are shareable for all tasks of this user group.

SCOPE = *HOST-SYSTEM
Only memory pools which were set up globally are included, i.e. memory pools which can be used by all tasks.

PROGRAM-VERSION =
Specifies the program version that DBL is to use. With regard to DBL, a program version is the version of a load unit which is about to be or has already been loaded.

PROGRAM-VERSION = *DBL-DEFAULT
The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *STD.

PROGRAM-VERSION = *STD
The load unit produced by the load call is assigned the version of the loaded library element as its program version.
If the symbol specified in the load call has already been loaded, the command looks for the program version defined in the SELECT-PROGRAM-VERSION command.
If no program version has yet been defined, DBL uses the first symbol it finds.

PROGRAM-VERSION = *BLANK
The program version is disregarded. If the load unit specified in the call is not yet loaded, the load unit being loaded is not assigned a version.

PROGRAM-VERSION = <composed-name 1..24 with-under>
Version of the program looked for among the programs which are already loaded. If this program version has not yet been loaded, the load unit being loaded is assigned this version.

ALTERNATE-LIBRARIES = *DBL-DEFAULT / *NONE / list-poss(2) *TASKLIB / *BLSLIB##

*This specification is ignored for a PAM-LLM (ALTERNATE-LIBRARIES=*NONE applies implicitly).*

Specifies whether alternate libraries are to be searched for symbols. They are also used for the DBL autolink function (see also the AUTOLINK operand).

If a list is specified then the libraries are searched in the specified order.

When the alternate libraries are searched, then the fixed type order L,R applies irrespective of the TYPE operand. In this case, only symbols are searched for, not element names.

ALTERNATE-LIBRARIES = *DBL-DEFAULT

The setting in the last MODIFY-DBL-DEFAULTS command applies. *NONE is set by default at task start.

ALTERNATE-LIBRARIES = *NONE

No alternate libraries are searched.

ALTERNATE-LIBRARIES = *TASKLIB

The system and/or user Tasklibs are searched in the following order:

1. The library that was assigned using the SET-TASKLIB command
2. The library $userid.TASKLIB
   or, if this does not exist
   The TASKLIB library under the system default ID (DEFLUID ID)

ALTERNATE-LIBRARIES = *BLSLIB##

The libraries assigned with the file link name BLSLIBnn or $BLSLBnn (00≤nn≤99) are searched.

Files with the file link name $BLSLBnn are alternate system libraries that are internally assigned by the system depending on the components that are required by the runtime system.

AUTOLINK =

*This specification is ignored for a PAM-LLM (AUTOLINK=*NO applies implicitly).*

Specifies whether the DBL autolink function is to be used.

AUTOLINK = *DBL-DEFAULT

The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *YES.

AUTOLINK = *YES

The DBL autolink function is used and both the input source and any alternate libraries are searched.

AUTOLINK = *NO

The DBL autolink function is not used.

AUTOLINK = *ALTERNATE-LIBRARIES
The DBL autolink function is used for alternate libraries only if alternate library searching has been enabled (ALTERNATE-LIBRARIES=*TASKLIB/*BLSLIB##).

ERROR-PROCESSING = *PARAMETERS(...)
Specifies how any errors that occur are to be handled.

NAME-COLLISION =
Specifies how name collisions between symbols of the same name are to be handled. Name collisions are discovered only if the symbols are *not* masked.

NAME-COLLISION = *DBL-DEFAULT
The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *STD.

NAME-COLLISION = *STD
Name collisions between nonmasked symbols are indicated by warning messages. The module containing the symbol with the same name is loaded. The new occurrence of the symbol is masked, i.e. it is no longer used to satisfy external references.

NAME-COLLISION = *ABORT
Loading of the current load unit is aborted if a name collision between nonmasked symbols is discovered.

UNRESOLVED-EXTRNS =
*This specification is ignored for a PAM-LLM (UNRESOLVED-EXTRNS=*STD applies implicitly).*
Specifies how unresolved external references are to be handled. All unresolved external references are output to the symbol file SYSOUT, with unresolved external dummy sections (XDSECs-R) listed separately.

UNRESOLVED-EXTRNS = *DBL-DEFAULT
The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *STD.

UNRESOLVED-EXTRNS = *STD
Unresolved external references are assigned an address specified in the ERROR-EXIT operand.

UNRESOLVED-EXTRNS = *DELAY
Unresolved external references are satisfied at a later time. This operand is permissible only when LOAD-INFORMATION=*REFERENCES.
The DBL stores the unresolved external references in the link context. After the next load unit is loaded into the context, the DBL attempts to satisfy the stored external references with CSECTs and ENTRYs from this new load unit. This process is repeated whenever further load units are loaded for as long as the context continues to exist. External dummy sections (XDSECs-R) cannot be stored.

When stored in the context, the unresolved external references are assigned a (preliminary) address, which is specified in the ERROR-EXIT operand.

UNRESOLVED-EXTRNS = *ABORT
All external references must be resolved. If not, the current load unit is aborted.

ERROR-EXIT = *DBL-DEFAULT / *NONE / <x-string 1..8>
Defines the address that unresolved external references are to be assigned if the operands UNRESOLVED-EXTRNS=*STD / *DELAY are specified.
Default: The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *NONE.
If *NONE is specified, external references are left unresolved (equivalent to specifying X'FFFFFFFF').

REPORTING = *PARAMETERS(...)
Specifies how the load operation is to be logged.

MESSAGE-CONTROL =
Specifies the lowest message class at which messages are to be output.

MESSAGE-CONTROL = *DBL-DEFAULT
The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *INFORMATION.

MESSAGE-CONTROL = *INFORMATION
Outputs messages of all message classes.

MESSAGE-CONTROL = *WARNING
Outputs messages of the WARNING and ERROR message classes only; messages of the INFORMATION message class are not output.

MESSAGE-CONTROL = *ERROR
Outputs messages of the ERROR message class only.

MESSAGE-CONTROL = *NONE
No messages are to be output.

PROGRAM-MAP =
Specifies whether or not a DBL list is to be output and defines the output destination for the DBL list.

PROGRAM-MAP = *DBL-DEFAULT
The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *NO.

PROGRAM-MAP = *NONE
No DBL list is to be output.

PROGRAM-MAP = *SYSLST(...)
The output destination is the SYSLST system file.

SYSLST-NUMBER =
Specifies the number of the SYSLST system file from the set SYSLST00 to SYSLST99.

SYSLST-NUMBER = <u>*STD</u>
The output destination is the system file SYSLST[00]

SYSLST-NUMBER = <integer 1..99>
The output destination is a SYSLST system file from the set SYSLST01 to SYSLST99.

PROGRAM-MAP = *SYSOUT
The output destination is the SYSOUT system file.

PROGRAM-MAP = *BOTH(...)
The system files SYSOUT and SYSLST are both output destinations.

SYSLST-NUMBER =
Specifies the number of the SYSLST system file from the set SYSLST00 to SYSLST99.

SYSLST-NUMBER = <u>*STD</u>
The output destination is the system file SYSLST[00]

SYSLST-NUMBER = <integer 1..99>
The output destination is a SYSLST system file from the set SYSLST01 to SYSLST99.

MONJV = <u>*NONE</u> / <filename 1..54 without-gen-vers>
Name of the job variable that is to monitor the program.
While the program is executing, the system sets the JV to one of the following values:

$R	program executing
$T	program terminated
$A	program abnormally terminated

This operand is available only to users who have the JV software product (see also the “Job Variables” manual [20]).

CPU-LIMIT = <u>*JOB-REST</u> / <integer 1..32767 *seconds*>
Maximum CPU time, in seconds, allowed for execution of the program.
If execution of the program exceeds the specified time, in interactive mode the program is interrupted and message EXC0075 is output. The user can request a dump, abort the program or continue the program. If a STXIT routine is defined in the program for the event of reaching the CPU time limit, this routine is executed and the program terminated.
In batch mode the program is terminated. See also section “Time limits in BS2000” on page 1-103.

CPU-LIMIT = *JOB-REST
If the job was started with a time limit, the remaining CPU time will be used as the time limit for the program. Otherwise, the program executes without a time limit.

TEST-OPTIONS =
Specifies whether symbolic addresses in the program may be used in testing (with AID). Symbolic addresses can only be used to debug programs for which debugging and diagnostic information (LSD/list for symbolic debugging) was generated at compile time. This entails setting certain compiler options when compiling the source program (see the appropriate compiler user guide).

TEST-OPTIONS = *DBL-DEFAULT
The setting from the last MODIFY-DBL-DEFAULTS command applies. At the start of a task, the preset value is *NONE.

TEST-OPTIONS = *NONE
The local symbolic address table/the internal symbolic address table is not used.

TEST-OPTIONS = *AID
This operand is required for testing with AID.
This specification permits the use of symbolic addresses when testing the program with AID commands. For this, a local symbolic address table or an internal symbolic address table must have been generated when the program was compiled.
This specification is only possible for OMs (element type R) and LLMs (element type L) if LOAD-INFORMATION=*DEFINITIONS/*REFERENCES is also specified.

RESIDENT-PAGES = *PARAMETERS(...)
Number of resident memory pages required for the program run.
This operand must be specified if, in the program, pages are to be made resident with a CSTAT macro (see the “Executive Macros” manual [22]). The permissible number of resident memory pages can be controlled by the operator.
If the operand is omitted (MIN=*STD,MAX=*STD), the memory requirements are taken from the start record of the program, which requires the file to be opened.

MINIMUM = *STD / <integer 0..32767 *4Kbyte*>
Minimum number of resident memory pages required.

MAXIMUM = *STD / <integer 0..32767 *4Kbyte*>
Maximum number of resident memory pages required.

VIRTUAL-PAGES = *STD / <integer 0..32767 *4Kbyte*>
The operand is evaluated only if a load module (element type C) is loaded.
Number of pageable memory pages which must be available for the program run. If the specified number exceeds the number of pages available on the computer, the command is rejected.

VIRTUAL-PAGES = *STD
The memory requirements are taken from the start record of the program.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
2	0	BLS0150	Warning on loading of the program (see the message on SYSOUT)
	32	BLS0152	Loading of the program aborted due to system error
	64	CMD0216	The user is not authorized to use the command
	64	BLS0153	Loading of the program aborted due to an unrecoverable error (see the SYSOUT message for the reason)

Notes

- Further information on loading programs can be found in the description of the loaders in the “BLSSERV” User Guide [3].
- DBL does not lock any libraries or library elements during the program run.
- Calling DBL with START-/LOAD-EXECUTABLE-PROGRAM always corresponds to a call with START-/LOAD-PROGRAM and RUN-MODE=*ADVANCED. The functionality of RUN-MODE=*STD is only available with the earlier commands START-/LOAD-PROGRAM.
- Some DBL functions are not available for an LLM that has already been linked into a load unit with BINDER and stored in a PAM file (PAM-LLM). Explicit specifications for the operands involved are ignored and the command is executed implicitly with the following values:
  - SHARE-SCOPE=*NONE
  - ALTERNATE-LIBRARIES=*NONE
  - AUTOLINK=*NO
  - UNRESOLVED-EXTRNS=*STD
- If multiple identical element types (TYPE operand) or alternate libraries (ALTERNATE-LIBRARIES operand) are specified, command execution is aborted with a warning.
- If NAME-SCOPE=*SYMBOL/*STD or ELEMENT-OR-SYMBOL= <c-string> is specified then:
  - if only element type C is declared (TYPE operand), command execution is aborted with an error message.
  - if element type C and at least one other element type are declared (TYPE operand) then element type C is ignored and the command is executed. If NAME-SCOPE=*STD then a warning is issued if an existing C element is skipped.

- As the specification of an element name > 32 characters in the ELEMENT-OR-SYMBOL operand cannot refer to a CSECT or ENTRY name (up to 32 characters), the following applies:
  - When NAME-SCOPE=*SYMBOL is specified the command is rejected.
  - When NAME-SCOPE=*STD is specified the search is restricted just to element names without a warning.
- As the specification of an element name > 41 characters in the ELEMENT-OR-SYMBOL operand cannot refer to a load module (element type C with up to 41 characters), the following applies:
  - When TYPE=C is specified the command is rejected.
  - If in addition to C at least one other element type is specified in the TYPE operand, element type C is ignored without a warning.
- DBL defines the following names for the context and the load unit:
  - the default name “LOCAL#DEFAULT” for the context
  - the default name “%UNIT” for the load unit
- Execution of programs cannot be nested. If a LOAD-EXECUTABLE-PROGRAM or LOAD-PROGRAM command is issued when a program is already loaded, the loaded program is unloaded and the new command is executed. No error message appears; however, an ABEND-STXIT routine defined in the unloaded program is first activated, unless the LOAD-PROGRAM command is called via the CMD macro (see the “Executive Macros” manual [22]).
- Program monitoring (see also the “Job Variables” manual [20]):
  - The status indicator in the job variable monitoring the program is set to “$R” when the program is started.
  - If a program is already loaded or a started program is interrupted by a HOLD-PROGRAM command or depression of the K2 key, the status indicator in the program-monitoring job variable is set to “$A”.
  - If the JV is not accessible at the time the command is processed, an error message is issued to SYSOUT and command processing is continued.

Examples

See the START-EXECUTABLE-PROGRAM command.

LOAD-PROGRAM

Load program (load or object module)

Description status:	BLSSERV V2.8A
Functional area:	Program control
Domain:	PROGRAM
Privileges:	STD-PROCESSING SAT-FILE-EVALUATION SAT-FILE-MANAGEMENT SECURITY-ADMINISTRATION

i As of BLSSERV V2.3, the LOAD-PROGRAM command has been replaced by LOAD-EXECUTABLE-PROGRAM.
However, LOAD-PROGRAM continues to be supported for reasons of compatibility. Despite this, the LOAD-EXECUTABLE-PROGRAM command should be used for new applications.
The section below simply contains a short functional description, the syntax format and the command return codes.

Function

The LOAD-PROGRAM command has two functions:

– It calls up the dynamic binder loader DBL and defines the primary input. The DBL then links LLMs (link and load modules) or OMs (object modules) to create a load unit, which it then loads into memory.

– It calls up the static loader ELDE. ELDE then loads a program (load module) linked by the TSOSLNK linkage editor into memory.

When calling with LOAD-PROGRAM, only the checks of the addressing mode compatible with BLSSERV < V2.5 are performed. This corresponds to specifying AMODE-CHECK=*STD in the START- or LOAD-EXECUTABLE-PROGRAM command. Execution of the loaded program or load unit does not begin until it is requested by means of a RESUME-PROGRAM command. If users wish to load *and* start the load unit or program with just one command, they may issue the START-EXECUTABLE-PROGRAM (or START-PROGRAM) command in place of the LOAD-PROGRAM and RESUME-PROGRAM commands.

In the case of operands which have *DBL-PARAMETERS as their default value, the MODIFY-DBL-DEFAULTS command can be used to modify the default DBL parameters which apply when the task is started. The parameters can be configured separately for calls to the LOAD- and START-... commands on the one hand and for the BIND macro call on the other. The SHOW-DBL-DEFAULTS command lists the values which are currently set. The RESET-DBL-DEFAULTS can be used to return the settings to the original DBL defaults.

Restrictions

Users with SECURITY-ADMINISTRATION, SAT-FILE-EVALUATION or SAT-FILE-MANAGEMENT privilege can use the command in procedures only.

Format

(Part 1 of 2)

```
LOAD-PROGRAM                                                              Alias: LDPG

FROM-FILE = <filename 1..54 without-gen> / *MODULE(...) / *PHASE(...)

  *MODULE(...)

   |  LIBRARY = *DBL-PARAMETERS / *STD / *OMF / <filename 1..54 without-gen> / *LINK(...)
   |
   |    *LINK(...)
   |
   |     |  LINK = <structured-name 1..8> / <filename 1..8 without-gen-vers>
   |
   | ,ELEMENT-OR-SYMBOL = *ALL / <composed-name 1..64 with-under>(...) /
   |                      <c-string 1..32 with-low>(...)
   |
   |    <composed-name 1..64 with-under>(...)
   |     |  VERSION = *STD / <text 1..24>
   |
   |    <c-string 1..32 with-low>(...)
   |     |  VERSION = *STD / <composed-name 1..24 with-under> / <c-string 1..24>
   |
   | ,PROGRAM-MODE = *DBL-PARAMETERS / *24 / *ANY
   |
   | ,RUN-MODE = *DBL-PARAMETERS / *STD / *ADVANCED(...)
   |
   |    *ADVANCED(...)
   |
   |     |  ALTERNATE-LIBRARIES = *DBL-PARAMETERS / *NO / *YES
   |     |
   |     | ,NAME-COLLISION = *DBL-PARAMETERS / *STD / *ABORT
   |     |
   |     | ,UNRESOLVED-EXTRNS = *DBL-PARAMETERS / *STD / *DELAY / *ABORT
   |     |
   |     | ,ERROR-EXIT = *DBL-PARAMETERS / *NONE / <x-string 1..8>
   |     |
   |     | ,MESSAGE-CONTROL = *DBL-PARAMETERS / *INFORMATION / *ERROR / *WARNING
   |     |
   |     | ,LOAD-INFORMATION = *DBL-PARAMETERS / *DEFINITIONS / *MAP / *NONE /
   |     |                     *REFERENCES
```

(Part 2 of 2)

```
            ,PROGRAM-MAP = *DBL-PARAMETERS / *NO / *SYSLST(...) / *SYSOUT / *BOTH(...)

               *SYSLST(...)
                    SYSLST-NUMBER = *STD / <integer 0..99>

               *BOTH(...)
                    SYSLST-NUMBER = *STD / <integer 0..99>
            ,SHARE-SCOPE = *DBL-PARAMETERS / *SYSTEM-MEMORY / *NONE / *ALL /
                                 *MEMORY-POOL(...)
               *MEMORY-POOL(...)
                    SCOPE = *ALL / *USER-ID / *USER-GROUP / *HOST-SYSTEM

            ,IGNORE-ATTRIBUTES = *DBL-PARAMETERS / *NONE / *READ-ONLY

            ,REP-FILE = *DBL-PARAMETERS / *NONE / <filename 1..54 without-gen>

            ,AUTOLINK = *DBL-PARAMETERS / *YES / *NO / *ALTERNATE-LIBRARIES

            ,PROGRAM-VERSION = *DBL-PARAMETERS / *STD / *BLANK /
                                     <composed-name 1..24 with-under>
   *PHASE(...)

         LIBRARY = <filename 1..54 without-gen>

        ,ELEMENT = <filename 1..41 without-gen-vers>

        ,VERSION = *STD / <text 1..24>
,CPU-LIMIT = *JOB-REST / <integer 1..32767 seconds>

,TEST-OPTIONS = *DBL-PARAMETERS / *NONE / *AID

,MONJV = *NONE / <filename 1..54 without-gen-vers>

,RESIDENT-PAGES = *PARAMETERS (...)

   *PARAMETERS(...)

         MINIMUM = *STD / <integer 0..32767 4Kbyte>

        ,MAXIMUM = *STD / <integer 0..32767 4Kbyte>

,VIRTUAL-PAGES = *STD / <integer 0..32767 4Kbyte>
```

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
2	0	BLS0150	Warning on program loading (see message to SYSOUT)
	32	BLS0152	Program loading aborted due to system error
	64	CMD0216	User has no authorization for command
	64	BLS0153	Program loading aborted due to unrecoverable error (for cause, see SYSOUT message)

LOCK-FILE-LINK

Lock file link name against deletion

Description status: BS2000 OSD/BC V10.0A

Functional area: File processing

Domain: FILE

Privileges: STD-PROCESSING
SAT-FILE-EVALUATION
SAT-FILE-MANAGEMENT

Function

The LOCK-FILE-LINK command locks an entry in the task file table (TFT), so that any subsequent REMOVE-FILE-LINK command for this TFT entry will be ineffective until an UNLOCK-FILE-LINK command is issued with the associated file link name.

If the file link name is amended by a CHANGE-FILE-LINK command, the TFT entry will remain locked, but now under the new file link name.

The ADD-FILE-LINK command can be used at any time to amend a TFT entry which has been locked by LOCK-FILE-LINK, but the lock will remain in effect.

Volumes from a TFT entry that is locked using LOCK-FILE-LINK are not released by the SECURE-RESOURCE-ALLOCATION command.

Format

LOCK-FILE-LINK
LINK-NAME = ***FIRST-BLANK** / <filename 1..8 without-gen>

Operands

LINK-NAME = *FIRST-BLANK / <filename 1..8 without-gen>
The file link name of the TFT entry which is to be locked.
If there is no TFT entry with this file link name, an entry will be created. This entry can then be allocated to a file by an ADD-FILE-LINK command or via the FCB and have its contents written into it.

LINK-NAME = *FIRST-BLANK
If *FIRST-BLANK is specified, the first TFT entry whose file link name consists of spaces (blanks) will be processed.

LINK-NAME = <filename 1..8 without-gen>
File link name of the TFT entry which is to be locked.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed
	1	CMD0202	Syntax or semantic error in command
	32	DMS05C7	Unexpected internal error in DMS
	64	DMS06FF	BCAM connection severed
	130	DMS0594	Not enough virtual memory available

Example

Locking file link names to prevent deletion

```
/add-file-link link=sortin,file-name=max.file.10          ————————————————————————   (1)
/add-file-link link=sortout,file-name=max.file.10.sort-1  ————————————————————————   (2)
/show-file-link                                           ————————————————————————   (3)
%
%-- LINK-NAME --------- FILE-NAME ---------------------------------------------
%   SORTIN              :2OS2:$USER1.MAX.FILE.10
%   SORTOUT             :2OS2:$USER1.MAX.FILE.10.SORT-1
/sort-file                                                ————————————————————————   (4)
%  BLS0523 ELEMENT 'SRT80', VERSION '079', TYPE 'L' FROM LIBRARY ':1OSH:$TSOS.SY
SLNK.SORT.079' IN PROCESS
%  BLS0524 LLM 'SRT80', VERSION '079' OF '2009-03-09 15:55:15' LOADED
%  BLS0551 COPYRIGHT (C) FUJITSU TECHNOLOGY SOLUTIONS 2009. ALL RIGHTS RESERVED
%  SRT1001  2012-04-28/17:59:44/000000.00 SORT/MERGE STARTED, VERSION 07.9C00/BS
2000V18.0
%  SRT1130  PLEASE ENTER SORT STATEMENTS
%  SRT1016  SORT/MERGE INPUT RECORDS:...........................2.924 (FROM 01)
%  SRT1030  SORT/MERGE OUTPUT RECORDS:..........................2.924
%  SRT1002  2012-04-28/17:59:45/000000.09 SORT/MERGE COMPLETED
/show-file-link                                           ————————————————————————   (5)
%
%-- LINK-NAME --------- FILE-NAME ---------------------------------------------
%   SORTOUT             :2OS2:$USER1.MAX.FILE.10.SORT-1

/add-file-link link=sortin,file-name=max.file.10          ————————————————————————   (6)

/lock-file-link link=sortin                               ————————————————————————   (7)

/show-file-link inf=(status=*yes)                         ————————————————————————   (8)
```

```
%
%-- LINK-NAME --------- FILE-NAME -------------------------------------------
%   SORTIN              :2OS2:$USER1.MAX.FILE.10
%  -------------------- STATUS ---------------------------------------------
%  STATE      = INACTIVE    ORIGIN     = FILE        LOCK-F-LI  = YES
%-- LINK-NAME --------- FILE-NAME -------------------------------------------
%   SORTOUT             :2OS2:$USER1.MAX.FILE.10.SORT-1
%  -------------------- STATUS ---------------------------------------------
%  STATE      = INACTIVE    ORIGIN     = FILE
/sort-file                                                                    (9)
%  BLS0523 ELEMENT 'SRT80', VERSION '079', TYPE 'L' FROM LIBRARY ':1OSH:$TSOS.SY
SLNK.SORT.079' IN PROCESS
%  BLS0524 LLM 'SRT80', VERSION '079' OF '2009-03-09 15:55:15' LOADED
%  BLS0551 COPYRIGHT (C) FUJITSU TECHNOLOGY SOLUTIONS 2009. ALL RIGHTS RESERVED
%  SRT1001  2012-04-28/18:01:12/000000.00 SORT/MERGE STARTED, VERSION 07.9C00/BS
2000V18.0
%  SRT1130  PLEASE ENTER SORT STATEMENTS
%  SRT1016  SORT/MERGE INPUT RECORDS:...........................2.924 (FROM 01)
%  SRT1030  SORT/MERGE OUTPUT RECORDS:..........................2.924
%  SRT1002  2012-04-28/18:01:12/000000.09 SORT/MERGE COMPLETED
/show-file-link inf=(status=*yes)                                            (10)
%
%-- LINK-NAME --------- FILE-NAME -------------------------------------------
%   SORTIN              :2OS2:$USER1.MAX.FILE.10
%  -------------------- STATUS ---------------------------------------------
%  STATE      = INACTIVE    ORIGIN     = FILE        LOCK-F-LI  = YES
%  REM-F-LINK = YES         RELE-DEV   = YES         UNL-R-TAPE = NO
%-- LINK-NAME --------- FILE-NAME -------------------------------------------
%   SORTOUT             :2OS2:$USER1.MAX.FILE.10.SORT-1
%  -------------------- STATUS ---------------------------------------------
%  STATE      = INACTIVE    ORIGIN     = FILE
/add-file-link link=sortout,file-name=max.file.10.sort-2                     (11)

/sort-file fields=(field-explicit(pos=5,length=6,sort-order=*ascend),
  field-explicit(pos=20,length=15,sort-order=*descend))                      (12)
%  BLS0523 ELEMENT 'SRT80', VERSION '079', TYPE 'L' FROM LIBRARY ':1OSH:$TSOS.SY
SLNK.SORT.079' IN PROCESS
%  BLS0524 LLM 'SRT80', VERSION '079' OF '2009-03-09 15:55:15' LOADED
%  BLS0551 COPYRIGHT (C) FUJITSU TECHNOLOGY SOLUTIONS 2009. ALL RIGHTS RESERVED
%  SRT1001  2012-04-28/18:01:42/000000.00 SORT/MERGE STARTED, VERSION 07.9C00/BS
2000V18.0
%  SRT1130  PLEASE ENTER SORT STATEMENTS
%  SRT1016  SORT/MERGE INPUT RECORDS:...........................2.924 (FROM 01)
%  SRT1030  SORT/MERGE OUTPUT RECORDS:..........................2.924
%  SRT1002 2012-04-28/18:01:42/000000.09 SORT/MERGE COMPLETED
/show-file-link inf=(status=*yes)                                            (13)
%
%-- LINK-NAME --------- FILE-NAME -------------------------------------------
%   SORTIN              :2OS2:$USER1.MAX.FILE.10
%  -------------------- STATUS ---------------------------------------------
%  STATE      = INACTIVE    ORIGIN     = FILE        LOCK-F-LI  = YES
%  REM-F-LINK = YES         RELE-DEV   = YES         UNL-R-TAPE = NO
%-- LINK-NAME --------- FILE-NAME -------------------------------------------
%   SORTOUT             :2OS2:$USER1.MAX.FILE.10.SORT-2
%  -------------------- STATUS ---------------------------------------------
%  STATE      = INACTIVE    ORIGIN     = FILE
/unlock-file-link link=sortin                                                (14)
/show-file-link                                                              (15)
%
%-- LINK-NAME --------- FILE-NAME -------------------------------------------
%   SORTOUT             :2OS2:$USER1.MAX.FILE.10.SORT-2
/remove-file-link link=sortout                                               (16)
/show-file-link                                                              (17)
%  DMS05E1 TASK FILE TABLE (TFT) NOT AVAILABLE OR SPECIFIED FILE NOT IN 'TFT'.
OPERATION NOT PROCESSED
```

(1) Creates a TFT entry for the file *MAX.FILE.10* under the link name *SORTIN*. This link name is used by the SORT utility routine as the default link name for an input file.

(2) Creates a TFT entry for the file *MAX.FILE.10.SORT-1* under the link name *SORTOUT*. This link name is used by the SORT utility routine as the default link name for an output file.

(3) Output of the TFT entries using SHOW-FILE-LINK.

(4) The SORT-FILE command is used to sort records of the file *MAX.FILE.10* into the file *MAX.FILE.SORT-10*. This command is a component of the SORT utility routine and provides specific sort functions at the command level (see the “SORT” manual [40]).
The command was called without operands, i.e. with default values, so the complete record is sorted; the input file is determined from the TFT entry with the link name *SORTIN*, and the output file from the TFT entry with the link name *SORTOUT*.

(5) The output of TFT entries shows that the entry with the link name *SORTIN* was automatically released after sorting (implicit REMOVE-FILE-LINK).

(6) A TFT entry with the link name *SORTIN* is created again for the file *MAX.FILE.10*.

(7) The TFT entry with the link name *SORTIN* is now locked with the LOCK-FILE-LINK command. This means that the TFT entry cannot be deleted so long as the lock remains in effect. The file *MAX.FILE.10* may be used as an input file more than once for sort runs.

(8) Output of the TFT entries with the status of the file link.

(9) The command SORT-FILE is called again (as in step 4).

(10) Output of the TFT entries with the status of the file link.
The TFT entry for *SORTIN* still exists. The value of *YES* in the *REM-F-LINK* output field indicates that a REMOVE-FILE-LINK was issued for this entry (in this case, the implicit REMOVE-FILE-LINK of SORT).

(11) Creates a new TFT entry for the file *MAX.FILE.10.SORT-2* with the link name *SORTOUT*. This file is to be used as the new output file for a sort run.

(12) The SORT-FILE command is called. The input file is still *MAX.FILE.10*, but the output file is now *MAX.FILE.10.SORT-2*. In this sort run, only selected parts of records are to be sorted and transferred to the output file (selective sort): 6 bytes as of the first data byte (record length field + 1 = 5) are sorted in ascending order. The second sort criterion selects 15 bytes of the record, starting with the 17th byte, and to be sorted in descending order.

(13) Output of the TFT entries with the status of the file link (as in step 10).

(14) The lock for the TFT entry with the link name *SORTIN* is released by the UNLOCK-FILE-LINK command.

(15) (implicit) REMOVE-FILE-LINK commands were executed only after the lock was removed.

(16) The TFT entry for *SORTOUT* is released explicitly with the REMOVE-FILE-LINK command.

(17) No further TFT entries exist.

LOCK-PRODUCT-VERSION

Lock product version

Description status:	IMON-GPN V3.3A
Functional area:	System control and optimization
Domain:	SYSTEM-MANAGEMENT
Privileges:	SUBSYSTEM-MANAGEMENT

Function

Systems support can use the LOCK-PRODUCT-VERSION command to lock the product version of an installation unit. The product version is then no longer available for nonprivileged users. The locked product version remains available for privileged users, (SUBSYSTEM-MANAGEMENT privilege) provided that the version is specified in the form <product-version mandatory-man-corr>.

Locking a product version may also affect the product's START command: if the START command is called with VERSION=*STD then the highest unlocked version is invoked.

The command cannot be executed unless product installation items are entered in the SCI. This requirement is satisfied in the following cases:

- The product was supplied as standard software by Fujitsu Technology Solutions using the SOLIS2 shipment procedure and was correctly installed in the system using IMON.
- The product takes the form of customer-specific software which was registered in the SCI using an in-house SYSSII file (see the “IMON” User Guide [19]).

A product version lock cannot extend beyond the current system run at the latest. It is released on the following system start. A lock is also set during product installation. The UNLOCK-PRODUCT-VERSION command can also be used to release the lock during the current system run.

Format

```
LOCK-PRODUCT-VERSION

PRODUCT-NAME = <text 1..30 without-sep>(...)

   <text 1..30 without-sep>(...)
     |   VERSION = <product-version mandatory-man-corr>
```

Operands

PRODUCT-NAME = <text 1..30 without-sep>(...)
Specifies the name of the IMON installation unit for the software product that is to be locked.

VERSION = <product-version mandatory-man-corr>
Specifies the product version. The specification must include the release version and correction status.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed successfully
	32	IMO9101	Command terminated abnormally (system error in external call; internal error in IMON-GPN)
	64	CMD0216	Privileges error
	64	IMO9100	Product or version not found, invalid product name

LOCK-USER

Lock access to system for user

Description status:	SRPMNUC V19.0A
Functional area:	User management
Domain:	USER-ADMINISTRATION
Privileges:	STD-PROCESSING USER-ADMINISTRATION
Routing code:	$

Function

This command can be used to prevent a user from accessing the system. The access lock for the user is entered in the user catalog of the specified pubset.
If this declaration is made for the home pubset in which logon validation is performed, the user is also denied access to the system.
This temporary access lock can be canceled with the UNLOCK-USER command.
If the definition applies to a data pubset, the access lock is filed there and evaluated only in the event that this pubset assumes the role of the home pubset.

The LOCK-USER command is not permitted

– for the user ID TSOS
– for the ID of the user issuing the command

Restriction

The only nonprivileged users (STD-PROCESSING privilege) authorized to issue this command are those designated as group administrators. The actions a group administrator can take are defined by the system support personnel. On setting up and managing user groups see the “SECOS” manual [35].

Format

LOCK-USER
USER-IDENTIFICATION = <name 1..8> ,**PUBSET** = ***HOME** / <cat-id 1..4>

Operands

USER-IDENTIFICATION = <name 1..8>
Identifies the user whose access is to be locked.

PUBSET = *HOME / <cat-id 1..4>
Enters the access lock for the user in the user catalog of the specified pubset.

PUBSET = *HOME
Specifies the user catalog of the home pubset, thus denying the user access to the system. Logon attempts by the user are rejected with a message.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	No error
2	0	SRM6001	Command executed with a warning
	1	SRM6010	Syntax error
	32	SRM6020	System error during command processing
	64	SRM6040	Semantic error
	130	SRM6030	Command temporarily cannot be executed

Example

```
/lock-user cognibs3
%  SRM2201 DO YOU WANT TO LOCK USER ID 'COGNIbBS3' ON PUBSET '1OSH' ? REPLY (Y=YES;
 N=NO)?y
/show-user-attr cognibs3
%SHOW-USER-ATTRIBUTES --- PUBSET TK82 - USER COGNIBS3       2012-01-29 15:09:50
%------------------------------------------------------------------------------
%USER-ID                    COGNIBS3          PUBLIC-SPACE-USED               0
%GROUP-ID                 *UNIVERSAL          PUBLIC-SPACE-LIMIT          20000
%DEFAULT-PUBSET                 TK82          PUBLIC-SPACE-EXCESS           *NO
%MAX-ACCOUNT-RECORDS             100          TEMP-SPACE-USED                 0
%DEFAULT-MSG-LANGUAGE                         TEMP-SPACE-LIMIT       2147483647
%                                             FILES                           0
%PROTECTION-ATTRIBUTES...                     FILE-NUMBER-LIMIT        16777215
%LOGON-PASSWORD                 *YES          JOB-VARIABLES                   0
%PASSWORD-MGMT     *USER-CHANGE-ONLY          JV-NUMBER-LIMIT          16777215
%TAPE-ACCESS                   *READ          RESIDENT-PAGES              32767
%FILE-AUDIT                      *NO          ADDRESS-SPACE-LIMIT            16
%                                             DMS-TUNING-RESOURCES        *NONE
%TEST-OPTIONS...                              CSTMP-MACRO-ALLOWED           *NO
%READ-PRIVILEGE                    1          CODED-CHARACTER-SET      EDF03IRV
%WRITE-PRIVILEGE                   1          PHYSICAL-ALLOCATION           *NO
%MODIFICATION            *CONTROLLED          USER-LOCKED                  *YES
%                                             CRYPTO-SESSION-USED             0
%AUDIT...                                     CRYPTO-SESSION-LIMIT          128
%HARDWARE-AUDIT             *ALLOWED          NET-STORAGE-USAGE         *ALLOWED
%LINKAGE-AUDIT              *ALLOWED
%
%PROFILE-ID  *NONE
%MAIL-ADDRESS  Abteilung Z8   Raum 55.105
%EMAIL-ADDRESS alfred.holli@incognito.de,
%              joachim.vogi@incognito.de,
%              (jk)johannes.kuli@incognito.de,
%              (mr)mathias.reh@incognito.de
%
%+---------+-----------+---------+--------+------------+-------+------+------+
%!ACCOUNT-#! CPU-LIMIT !SPOOLOUT-!MAX-RUN-!MAX-ALLOWED-!NO-CPU-!START-!INHIB-!
%!         !           !  CLASS  !PRIORITY!  CATEGORY  ! LIMIT ! IMMED! DEACT!
%+---------+-----------+---------+--------+------------+-------+------+------+
%! ACC00015!      10000!    0    !  255   !    STD     !  NO   !  YES !   NO !
%+---------+-----------+---------+--------+------------+-------+------+------+
%DEFAULT-ACCOUNT-# FOR LOGON:        *NONE
%DEFAULT-ACCOUNT-# FOR REMOTE-LOGIN: *NONE
%
%DEFAULT-JOB-CLASS FOR BATCH-JOBS:   JCBSTD
%DEFAULT-JOB-CLASS FOR DIALOG-JOBS:  JCDSTD
%LIST OF JOB-CLASSES ALLOWED:
%JCBATCHF JCBSTD   JCB00050 JCB00200 JCB02000 JCB05000 JCB32000 JCDSTD
%------------------------------------------------------------------------------
%SHOW-USER-ATTRIBUTES           END OF DISPLAY FOR USER COGNIBS3 ON PUBSET TK82
/
```

User ID *COGNIBS3* is temporarily barred from accessing the system. This is indicated by the value *YES in the USER-LOCKED output field of the user entry displayed by the SHOW-USER-ATTRIBUTES command.

LOGOFF

Terminate current job

Description status: BS2000 OSD/BC V10.0A

Functional area: Job processing

Domain: JOB

Privileges: All privileges

Function

The LOGOFF command terminates the current job. Subsequently, the virtual memory pages and devices occupied by the job are released and the output system files made available for output.
If new file generations were created during the job, the system lists the names of the file generations involved, their base number as well as the names of the first and the current file generation.

> The LOGOFF command is now supported only for reasons of compatibility; the EXIT-JOB command should be used to terminate the job. The EXIT-JOB command offers the user expanded functionality.

Format

LOGOFF
KEEP-CONNECTION = ***NO** / ***Y**ES ,**SYS**TEM-**OUT**PUT = ***STDOUT** / ***PRINT** / ***DEL**ETE / ***TAPE-OUT**PUT

Operands

KEEP-CONNECTION =
Specifies whether the connection to the computer is to be kept so that a new job can be started immediately.
This operand is permitted only in interactive mode; in batch mode it is ignored.

KEEP-CONNECTION = *NO
Clears the connection.

KEEP-CONNECTION = *YES
Maintains the connection with the computer. A new job can be started immediately.

SYSTEM-OUTPUT =
Specifies whether system files are to be output and defines the output medium. SYSLST and SYSOUT are not output if they are empty.

SYSTEM-OUTPUT = *STDOUT
Depending on the setting of the system parameter SSMOUT, the system files SYSLST and SYSOUT (batch mode) are output to printer or transferred by email. Transfer by email can be requested explicitly using the EXIT-JOB command.

SYSTEM-OUTPUT = *PRINT
Outputs the system files SYSLST and SYSOUT (in batch mode) to printer.
An additional list of dialog jobs can be output if the value NO-SPOOL is set for the system parameter SSMLGOF1.

SYSTEM-OUTPUT = *DELETE
Output of the system file is suppressed.

SYSTEM-OUTPUT = *TAPE-OUTPUT
Outputs the system files on tape.
SYSLST and SYSOUT (batch mode) are written to tape in the file: *TAPE.TSNnnnn*, where *nnnn* is the task sequence number of the job ended with LOGOFF.

Return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed [1)]
	1	CMD0202	Syntax error
	32	CMD0221	System error; no message is sent to SYSOUT

[1)] Cannot be evaluated since in this case the task is terminated!

Note

If a LOGOFF command is issued while a program is loaded, any ABEND-STXIT routine that may have been defined is activated. This is not the case when the LOGOFF command is called via the CMD macro (see the “Executive Macros” manual [22]).