RELEASE GUIDE

HCC/MVS 3.0

HOST

COMMUNICATION

CONTROL

1st edition

Document number: 600232-A

IMPRINT

1st Edition

This Manual has been written with utmost care. Textual or formal errors still cannot be excluded!

Protected trademarks are not marked as such in this Manual. The fact that these trademarks are not shown does not imply that the trade names are free for use.

All rights withheld, including those arising from applications for proprietary rights. The publisher retains all rights of disposition, such as copying or distribution.

Subject to changes without notice.

Publisher: EMASS/GRAU Storage Systems GmbH,
Eschenstraße 3, D-89558 Böhmenkirch

(c) 1995 by EMASS/GRAU Storage Systems GmbH, Eschenstr. 3, D-89558 Böhmenkirch.

1st Edition in November 1995

CONTENTS

1 RELEASE GUIDE FOR HCC/MVS 3.0.0 1-1

1.1 LITERATURE 1-1

2 HACCPARM STATEMENT CHANGES 2-1

2.1 AUTOFREEVOL (CHANGED) 2-1
2.2 FPMADEF (CHANGED) 2-2
2.3 KEEPWT - TAPE DEVICES (CHANGED) 2-3
2.4 LDEV=RXX/TXX (CHANGED) 2-4
2.5 LDEV=RXX/TXX - DYN (CHANGED) 2-6
2.6 LDEV=RXX/TXX - FPMA (CHANGED) 2-8
2.7 LDEV=TXX - HWS (CHANGED) 2-10
2.8 LDEV=EXX (CHANGED) 2-11
2.9 LDEV=FXX (CHANGED) 2-13
2.10 LDEV=IXX (CHANGED) 2-15
2.11 LDEV=WXX (CHANGED) 2-17
2.12 MSG=CBR4405D (NEW) 2-18
2.13 MSG=CBR4406D (NEW) 2-19
2.14 MSG=CBR4423D (NEW) 2-20
2.15 MSG=CBR4424D (NEW) 2-21
2.16 MSG=CBR4432D (NEW) 2-22
2.17 MSG=CBR4438D (NEW) 2-23
2.18 MSG=CBR4439D (NEW) 2-24
2.19 ODSCR (NEW) 2-25
2.20 PDSNEXCN (CHANGED) 2-26
2.21 PRTY MO (CHANGED) 2-27
2.22 STORGRP (NEW) 2-28
2.23 TDSNXXXX (DELETED) 2-29
2.24 TMSEXIT (CHANGED) 2-30
2.25 UNIT TYPE=CAR (CHANGED) 2-31
2.26 UNIT TYPE=IOS (NEW) 2-33
2.27 UNIT TYPE=OAD (NEW) 2-34

3 START PROCEDURE CHANGES 3-1

3.1 START COMMAND 3-1
3.2 STARTED TASK JCL FOR SINGLE HOST COMPLEX 3-5
3.3 STARTED TASK JCL FOR MULTI HOST COMPLEX (LU 6.2) 3-6
3.4 START AFTER INSTALLATION 3-7

4 OPERATOR COMMANDS CHANGES 4-1

4.1 ACC - ARCHIVE COMPARISON USING COORDINATES (CHANGED) 4-1
4.2 ACV - ARCHIVE COMPARISON USING VOLSER (CHANGED) 4-3
4.3 ADD - ADD VOLSERS FOR ROBOT FUNCTIONS (DELETED) 4-4
4.4 ALLOC - RESERVING TAPE UNITS (CHANGED) 4-5
4.5 CANCEL LABEL - DELETING INACTIVE ENTRIES (NEW) 4-6
4.6 CL - INDIVIDUAL TAPE UNIT CLEANING (CHANGED) 4-7
4.7 CX CMD - LU 6.2 INTERCOMMUNICATION (CHANGED) 4-8
4.8 CX GETMAJOR - LU 6.2 MAJOR COMPLEX (CHANGED) 4-9
4.9 CX MSG - LU 6.2 SEND MESSAGE (CHANGED) 4-10
4.10 CX SETMAJOR - LU 6.2 SET MAJOR (CHANGED) 4-11
4.11 CX SETMINOR - LU 6.2 SET MINOR (CHANGED) 4-12
4.12 CX D - DISPLAY TOKEN RING (DELETED) 4-13
4.13 CX ECHO - TOKEN RING TEST (DELETED) 4-13
4.14 CX GETLINE - TOKEN RING COMMUNICATION PATH (DELETED) 4-13
4.15 CX SHUTDOWN - SHUTDOWN TOKEN RING (DELETED) 4-13
4.16 CX SWLINE - SWITCH TOKEN RING PATH (DELETED) 4-13
4.17 DC - DISPLAY COORDINATE (NEW) 4-14
4.18 DEL - DELETE VOLSER FOR ROBOT FUNCTIONS (DELETED) 4-16
4.19 DISPLAY - DISPLAY FUNCTIONS (CHANGED) 4-17
4.20 DLC - DOWNLOAD ARCHIVE USING COORDINATES (CHANGED) 4-26
4.21 DLV - DOWNLOAD ARCHIVE USING VOLSER (CHANGED) 4-29
4.22 DRT - DISPLAY ACTIVE ROBOT-FUNCTIONS (CHANGED) 4-30
4.23 DSC - DISPLAY SCRATCH DATA MEDIA (CHANGED) 4-31
4.24 DU - DISPLAY UNIT ACTIVITY (CHANGED) 4-33
4.25 DV - DISPLAY VOLSER INFORMATION (CHANGED) 4-36
4.26 EJ - EJECT MEDIUM (CHANGED) 4-45
4.27 FLIP - TURNOVER OPTICAL DISK (NEW) 4-47
4.28 FMA - ADD FOREIGN MOUNT CARTRIDGE ENTRY (CHANGED) 4-48
4.29 FMD - DELETE FOREIGN MOUNT ENTRY (CHANGED) 4-50
4.30 FREE - RELEASE TAPE UNITS (CHANGED) 4-51
4.31 FREEVOL - DETERMINE SCRATCH DATA MEDIA (CHANGED) 4-52
4.32 HELP - DISPLAY COMMAND OVERVIEW (CHANGED) 4-56
4.33 INC - INVENTORY USING COORDINATES (CHANGED) 4-57
4.34 INV - INVENTORY USING VOLSER (CHANGED) 4-59
4.35 K - INDIVIDUAL KEEP (CHANGED) 4-60
4.36 LCK - DEFINE LABEL CHECK (CHANGED) 4-62
4.37 LJB - LOAD JUKEBOX (NEW) 4-63
4.38 M - INDIVIDUAL MOUNT (CHANGED) 4-64
4.39 MOVEH - MOVE TO HOME COORDINATE (CHANGED) 4-66
4.40 MOVET - MOVE TO TEMPORARY COORDINATE (CHANGED) 4-67
4.41 MRO - ALLOWING MOUNTS (DELETED) 4-68
4.42 PCL - STOP AUTOM. TAPE CLEANING (CHANGED) 4-69
4.43 PLCK - STOP LABEL CHECK (CHANGED) 4-70
4.44 PRT - STOP ROBOT TEST (DELETED) 4-71
4.45 PTLI - STOP TAPE LABEL INITIALIZATION (CHANGED) 4-72
4.46 SCL - START AUTOM. TAPE UNIT CLEANING (CHANGED) 4-73
4.47 SET PRTY SET BASIC PRIORITY (CHANGED) 4-74
4.48 SETUP - CORRECT THE UNIT LOG (CHANGED) 4-77
4.49 SETUP AR - CORRECT UNIT AR/AMU (CHANGED) 4-79
4.50 SLCK - START LABEL CHECK (CHANGED) 4-80
4.51 SRT - START ROBOT TEST (DELETED) 4-81
4.52 START LABEL START OPTICAL DISK LABEL PROCESS (NEW) 4-82

4.53 STLI - START TAPE LABEL INITIALIZATION (CHANGED) 4-83
4.54 TLI - DEFINE TAPE LABEL INIT (CHANGED) 4-84
4.55 UJB - UNLOAD JUKEBOX (NEW) 4-85
4.56 ULC - UPLOAD ARCHIVE USING COORDINATES (CHANGED) 4-86
4.57 ULV - UPLOAD ARCHIVE USING VOLSER (CHANGED) 4-88
4.58 UNLOAD - UNLOAD UNIT (CHANGED) 4-90
4.59 UPC - UPDATE ARCHIVE USING COORDINATES (CHANGED) 4-91
4.60 UPV - UPDATE ARCHIVE USING VOLSER (CHANGED) 4-92
4.61 VARY - CONTROL COMMUNICATION (CHANGED) 4-93
4.62 VI - START INSERTION (CHANGED) 4-94
4.63 VI TC - START INSERTION WITH TARGET COORDINATE (CHANGED) 4-96

5 SMF RECORD SUBTYPE CHANGES 5-1

6 HCC MESSAGES CHANGES 6-1

6.1 CHANGED HCC MESSAGES 6-1
6.2 NEW HCC MESSAGES 6-5

7 HAA MENU CHANGES 7-1

7.1 NEW AND CHANGED HAA MENUS 7-1
7.1.1 OPTICAL LIBRARY 7-1
7.1.2 CARTRIDGE LIBRARY MENU 7-10
7.2 DELETED HAA MENUS 7-10

INDEX I-1

1 RELEASE GUIDE for HCC/MVS 3.0

1.1 LITERATURE

Manual	Order number	Reference
General Information Manual	600226-A	G
Installation/Customization Guide	600227-A	G
System Reference Guide	600233-A	G
Operators Guide	600231-A	G
Command Reference	600223-A	G
Conversion Notebook	600224-A	G
Installation/Customization Reference	600228-A	G
Messages and Codes	600230-A	G
ISPF User Guide	600229-A	G
Release Guide	600232-A	G

2 HACCPARM STATEMENT CHANGES

This Section describes the new and changed HACCPARM statements as compared to HCC/MVS 2.4.0. Statements deleted from HCC/MVS 2.4.0 are also listed.

2.1 AUTOFREEVOL (CHANGED)

AUTOMATIC FREEVOL COMMAND	(optional)
AUTOFREEVOL=YES/NO	

YES: Automatic **FREEVOL** command for SCRATCH tape bottlenecks
NO: No automatic **FREEVOL** command

The AUTOFREEVOL parameter can be automatically triggered to demand scratch tapes.

AUTOFREEVOL=YES triggers a **FREEVOL REP** command when:

1. the defined "Scratch Warning level" is reached or
2. all tapes designated as scratch tapes in the HCC Archive mirror are marked as NON-SCRATCH for one or more pools.

Default: `AUTOFREEVOL=NO`

Example: `AUTOFREEVOL=YES`

When the SCRATCH volsers are made available in a SEQ file instead of using TMS, always ensure that only the newest SCRATCH volsers are available in this file. This means deleting the contents of the SEQ file immediately after it has been processed with FREEVOL.

2.2 FPMADEF (CHANGED)

ALLOCATING SCRATCH GROUPS TO FPMA (optional)
FPMADEF=coor1,coor2,S=s,scratch

coor1	FMPA start coordinate
coor2	FPMA end coordinate
s	AML System (1,2)
scratch	Scratch group (PRIVAT,GREEN,...)

FPMADEF serves to assign various Scratch groups to an installed FPMA[1)].

Defining the MVS standard Scratch group "PRIVAT" automatically implies the group "SCRTCH".

This parameter prerequisites FPMALOC statements within the UNIT statement.

Default: None

Example:

```
FPMADEF=04010101,04010225,S=1,PRIVAT (=SCRTCH)
FPMADEF=04010301,04010425,S=1,GREEN
FPMADEF=05010201,05010215,S=2,WHITE
FPMADEF=05010216,05010325,PRIVAT     (=SCRTCH)
```

1) FPMA = Fixed Preferred MOUNT/KEEP Area

2.3 KEEPWT - TAPE DEVICES (CHANGED)

WAIT-TIME BEFORE SENDING THE KEEP REQUEST	(optional)
KEEPWT=nn[,rr,ddd]	

nn	Wait-time before sending the first KEEP, in seconds	nn	=00-99
rr	Number of AUTO-REPEATs after an N206 message	rr	=00-99
ddd	Wait-time before the first send repeat, in seconds	ddd	=030-999

KEEPWT controls the wait-time before sending a **KEEP** command for tape devices and the respective recovery measures.

The MVS KEEP message normally follows immediately after a **CLOSE** for the file. The rewind time for a full cartridge is approx. 45 seconds. Full tapes on 3490E units have no rewind time.

The **KEEP** command is sent immediately to AML, providing no other robot activities are pending. The robot removes the cartridge, and waits up to 60 seconds for the cartridge to be ejected when this fails. Message N206 is sent to HCC when the cartridge is not ejected.

KEEPWT statement specifications can increase this wait-time. HCC then sends the **KEEP** task after this interval has elapsed.

Further actions can be triggered when errors occur: the operand rr defines the number of automatic **KEEP** repeats and ddd defines the wait-time between the repeats.

KEEPWT can have a certain negative influence on performance but only when the cartridge does not require the complete rewind time and the respective **KEEP** request is the only task in the HCC wait queue.

The wait-time for automatic cleaning should also be reduced when **KEEPWT** is used (refer to the CLEAN statement).

Default: `KEEPWT=00`

Example: `KEEPWT=50,2,060`

Assumed rewind time 50 seconds, 2 repeats, each after 60 seconds.

KEEP commands for IBM 3995 - Operator Accessible Drives are processed without KEEP-WAIT-TIME.

2.4 LDEV=Rxx/Txx (CHANGED)

AML COMPONENTS- RACK/TOWER	(mandatory)
LDEV=Rxx,coor,seg1,seg2,row1,row2,pos1,pos2,slots[,S=sys][,R=rob][,LIB=x] LDEV=Txx,coor,seg1,seg2,row1,row2,pos1,pos2,slots[,S=sys][,R=rob][,LIB=x]	

Rxx	Component number (rack)		
Txx	Component number (tower)		
coor	Component coordinate (acc. to component number)		
seg1/seg2	First/last segment in the component		
	Tower:		(01-99)
	Rack:	(AML/2, AML/E, AML/J)	(01-01)
		(ABBA/1)	(01-99)
row1/row2	First/last row in a segment		
pos1/pos2	First/last position in a row		
slots	Number of slots in a row		(01-99)
sys	1	1. AML System	(Default)
	2	2. AML System	
rob	1	Robot 1	(Default)
	2	Robot 2	
	1,2	Robots 1 and 2	
x	C	Library type is Cartridge	(Default)
	O	Library type is Optical	

The LDEV=Rxx/Txx statement defines the rack/tower coordinates in the AML system. A separate LDEV statement is required for each AML unit installed.

The LDEV=Rxx/Txx statement for a twin robot system with 3 towers and 2 racks.

ABBA/1/AML/2

```
LDEV=T01,01,01,32,01,18,01,10,10,S=1,R=1,2  (Tower 5760 cart.)
LDEV=T02,02,01,32,01,15,01,10,10,S=1,R=1,2  (Tower 4800 cart.)
LDEV=T03,03,01,16,01,15,01,10,10,S=1,R=1,2  (Tower 2400 cart.)
```

ABBA/1

```
LDEV=R04,04,01,03,01,04,01,25,25,S=1,R=1    (Reg. 300 cart.)
LDEV=R05,05,01,06,01,04,01,25,25,S=1,R=2    (Reg. 600 cart.)
```

AML/2

```
LDEV=R04,04,01,01,01,12,01,10,10,S=1,R=1
LDEV=R05,05,01,01,01,12,01,10,10,S=1,R=1
LDEV=R06,06,01,01,01,12,01,10,10,S=1,R=1    (Reg. 360 cart.)
LDEV=R07,07,01,01,01,12,01,10,10,S=1,R=2
LDEV=R08,08,01,01,01,12,01,10,10,S=1,R=2
LDEV=R09,09,01,01,01,12,01,10,10,S=1,R=2    (Reg. 360 cart.)
```

The LDEV=Rxx/Txx statement for an AML/J system with one storage element and an E/I/F storage.

AML/J

```
LDEV=R01,01,01,01,01,03,01,30,30,S=1,R=1
LDEV=R02,02,01,01,01,02,01,45,45,S=1,R=1
LDEV=R03,03,01,01,01,05,01,45,45,S=1,R=1
```

This example does not reflect slot losses for the AMU and drives.

LDEV statement changes demand correction (generation) of the HCC Archive mirror (refer to HAA).

All towers and racks must be numbered consecutively in ascending sequence.

2.5 LDEV=Rxx/Txx - DYN (CHANGED)

AML COMPONENTS - DYNAMIC AREA	(optional)
LDEV=Rxx,coor,seg1,seg2,row1,row2,pos1,pos2,slots[,S=sys][,R=rob],DYN[type][,LIB=x] LDEV=Txx,coor,seg1,seg2,row1,row2,pos1,pos2,slots[,S=sys][,R=rob],DYN[type][,LIB=x]	

Rxx	Component number (rack)		
Txx	Component number (tower)		
coor	Component coordinate (acc. to component number)		
seg1/seg2	First/last segment in the component		
	Tower:		(01-99)
	Rack:	(AML/2, AML/E, AML/J)	(01-01)
		(ABBA/1)	(01-99)
row1/row2	First/last row in a segment		
pos1/pos2	First/last position in a row		
slots	Number of slots in a row		(01-99)
sys	1	1. AML System	(Default)
	2	2. AML System	
rob	1	Robot 1	(Default)
	2	Robot 2	
	1,2	Robots 1 and 2	
type	HOR	Horizontal storage format	
	VER	Vertical storage format	(Default)
x	C	Library type is Cartridge	(Default)

The LDEV=Rxx/Txx-DYN statement **redefines** the existing rack/tower coordinates in the AML system. Two LDEV statements are required for each dynamic (sub) area; one LDEV statement describing the hardware and a second LDEV-DYN statement describing the logical area.

Only the default value for LÌB= may be used for DYN, HWS or FPMA.

The LDEV=Rxx/Txx statement for a dynamic area in tower 1 with 640 slots (horizontal rotation).

ABBA/1/AML/2/E

```
LDEV=T01,01,01,32,01,18,01,10,10,S=1,R=1,2
LDEV=T01,01,01,32,17,18,01,10,10,S=1,R=1,2,DYN,HOR
```

The LDEV=Rxx/Txx statement for a dynamic area in 2 towers each with 5760 slots.

ABBA/1/AML/2/E

```
LDEV=T02,02,01,32,01,18,01,10,10,S=1,R=1,2
LDEV=T02,02,01,32,01,18,01,10,10,S=1,R=1,2,DYN,VER
LDEV=T03,03,01,32,01,18,01,10,10,S=1,R=1,2
LDEV=T03,03,01,32,01,18,01,10,10,S=1,R=1,2,DYN,VER
```

The LDEV=Rxx/Txx statement for a dynamic area in an AML/J system with one storage element.

AML/J

```
LDEV=R01,01,01,01,01,03,01,30,30,S=1,R=1
LDEV=R01,01,01,01,01,03,01,30,30,S=1,R=1,DYN
LDEV=R02,02,01,01,01,02,01,45,45,S=1,R=1
LDEV=R02,02,01,01,01,02,01,45,45,S=1,R=1,DYN
LDEV=R03,03,01,01,01,05,01,45,45,S=1,R=1
LDEV=R03,03,01,01,01,05,01,45,45,S=1,R=1,DYN
```

This example does not reflect slot losses for the AMU and drives.

LDEV statement changes demand correction (generation) of the HCC Archive mirror (refer to HAA).

Internally, HCC uses all redefined dynamic areas as a complete, coherent dynamic area within an AML system. When dynamic areas are used within Major/Minor complexes, they must be defined separately and must not overlap.

This is necessary because all primary HCCs represent independent systems with regard to files.

2.6 LDEV=Rxx/Txx - FPMA (CHANGED)

AML COMPONENTS - FPMA	(optional)

```
LDEV=Rxx,coor,seg1,seg2,row1,row2,pos1,pos2,slots[,S=sys][,R=rob],FPMA
LDEV=Txx,coor,seg1,seg2,row1,row2,pos1,pos2,slots[,S=sys][,R=rob],FPMA
```

Rxx	Component number (rack)		
Txx	Component number (tower)		
coor	Component coordinate (acc. to component number)		
seg1/seg2	First/last segment in the component		
	Tower:		(01-99)
	Rack:	(AML/2, AML/E, AML/J)	(01-01)
		(ABBA/1)	(01-99)
row1/row2	First/last row in a segment		
pos1/pos2	First/last position in a row		
slots	Number of slots in a row		(01-99)
sys	1	1. AML System	(Default)
	2	2. AML System	
rob	1	Robot 1	(Default)
	2	Robot 2	
	1,2	Robots 1 and 2	

The LDEV=Rxx/Txx-FPMA statement **redefines** the existing rack/tower coordinates in the AML system. Two LDEV statements are required for each FPMA; one LDEV statement describing the hardware and a second LDEV-FPMA statement describing the logical area.

The LDEV=Rxx/Txx statement for an FPMA in rack 1 with 300 slots served by robot 1.

ABBA/1:

```
LDEV=R01,01,01,03,01,04,01,25,25,S=1,R=1
LDEV=R01,01,01,03,01,04,01,25,25,S=1,R=1,FPMA
```

AML/2:

```
LDEV=R01,01,01,01,01,12,01,10,10,S=1,R=1
LDEV=R02,02,01,01,01,12,01,10,10,S=1,R=1
LDEV=R03,03,01,01,01,12,01,10,10,S=1,R=1
LDEV=R01,01,01,01,01,12,01,10,10,S=1,R=1,FPMA
LDEV=R02,02,01,01,01,12,01,10,10,S=1,R=1,FPMA
LDEV=R03,03,01,01,01,12,01,10,10,S=1,R=1,FPMA
```

LDEV statement changes demand correction (generation) of the HCC Archive mirror (refer to HAA).

A complete example is shown in the Annex. This example includes all the statements required to create FPMAs.

2.7 LDEV=Txx - HWS (CHANGED)

AML COMPONENTS - HWS	(optional)
LDEV=Txx,coor,seg1,seg2,row1,row2,pos1,pos2,slots[,S=sys][,R=rob],HWS	

Txx	Component number (tower)		
coor	Component coordinate (acc. to component number)		
seg1/seg2	First/last segment in the component		
row1/row2	First/last row in a segment		
pos1/pos2	First/last position in a row		
slots	Number of slots in a row		(01-25)
sys	1	1. AML System	(Default)
	2	2. AML System	
rob	1	Robot 1	(Default)
	2	Robot 2	
	1,2	Robots 1 and 2	

The LDEV=Rxx/Txx-HWS statement **redefines** the existing rack/tower coordinates in the AML system. Two LDEV statements are required for each FPMA; one LDEV statement describing the hardware and a second LDEV-HWS statement describing the logical area.

The LDEV=Txx statement for a HCC work area (HWS) in tower 1 with 10 slots.

```
LDEV=T01,01,01,32,01,18,01,10,10,S=1,R=1,2
LDEV=T01,01,32,32,09,09,01,10,10,S=1,R=1,2,HWS
```

LDEV statement changes demand correction (generation) of the HCC Archive mirror (refer to HAA).

A HWS can only be defined for a tower belonging to an ABBA/1 or AML/2 twin robot system.

Only one work area is possible for each AML system.

The number of HWS slots should at least equal the number of tape units installed. The number of HWS slots does not have to exceed the number of foreign mount slots defined.

2.8 LDEV=Exx (CHANGED)

AML COMPONENTS - EJECT AREA	(mandatory)
LDEV=Exx,00,02,02,row1,row2,pos1,pos2,slots[,S=sys][,R=rob]	

Exx	Component number (eject area)		
row1/row2	First/last row in a segment		
pos1/pos2	First/last position in a row		
slots	Number of slots in a row		(01-25)
sys	1	1. AML System	(Default)
	2	2. AML System	
rob	1	Robot 1	(Default)
	2	Robot 2	

The LDEV=Exx statement defines the coordinates of an eject area in an AML system. A separate LDEVB statement is mandatory for each AML unit installed.

Eject areas can overlap.

The LDEV=Exx statement for 2 twin robot systems with 4 E/I/F storage areas (standard) for AML/2.

AML/2	`LDEV=E01,00,02,02,01,11,01,10,10,S=1,R=1`	(110 slots)
	`LDEV=E11,00,02,02,11,11,01,10,10,S=1,R=1`	(10 slots)
	`LDEV=E02,00,02,02,13,23,01,10,10,S=1,R=2`	(110 slots)
	`LDEV=E12,00,02,02,23,23,01,10,10,S=1,R=2`	(10 slots)
	`LDEV=E03,00,02,02,01,11,01,10,10,S=2,R=1`	(110 slots)
	`LDEV=E13,00,02,02,11,11,01,10,10,S=2,R=1`	(10 slots)
	`LDEV=E04,00,02,02,13,23,01,10,10,S=2,R=2`	(110 slots)
	`LDEV=E14,00,02,02,23,23,01,10,10,S=2,R=2`	(10 slots)

The LDEV=Exx statement for an ABBA/1 system.

ABBA/1	`LDEV=E01,00,02,02,01,03,01,25,25,S=1,R=1`	(75 slots)
	`LDEV=E11,00,02,02,03,03,01,25,25,S=1,R=1`	(25 slots)

The LDEV=Exx statement for an AML/J system with 2 E/I/F storage areas.

AML/J	`LDEV=E01,00,02,02,01,01,01,10,10,S=1,R=1`	(10 slots)
	`LDEV=E11,00,02,02,01,01,01,01,10,S=1,R=1`	(1 slot)
	`LDEV=E02,00,02,02,02,02,01,10,10,S=1,R=1`	(10 slots)
	`LDEV=E12,00,02,02,02,02,01,01,10,S=1,R=1`	(1 slot)

LDEV statement changes demand correction (generation) of the HCC Archive mirror (refer to HAA).

This definition must match the AR/AMU[1)] definitions.

1) AR/AMU = Archive computer ABBA/1 or Archive Management Unit AML/2, AML/E, AML/J

2.9 LDEV=Fxx (CHANGED)

AML COMPONENTS - FOREIGN MOUNT AREA	(optional)
LDEV=Fxx,00,03,03,row1,row2,pos1,pos2,slots[,S=sys][,R=rob]	

Fxx	Component number (foreign mount area)		
row1/row2	First/last row in a segment		
pos1/pos2	First/last position in a row		
slots	Number of slots in a row		(01-25)
sys	1	1. AML System	(Default)
	2	2. AML System	
rob	1	Robot 1	(Default)
	2	Robot 2	

The LDEV=Fxx statement defines the coordinates of the foreign mount area in an AML system. A separate LDEV statement is mandatory for each AML unit installed.

Foreign mount areas must **not** overlap insert and eject areas.

The LDEV=Fxx statement for 2 twin robot systems with 4 E/I/F storage areas (standard) for AML/2.

AML/2	`LDEV=F01,00,03,03,12,12,01,10,10,S=1,R=1`	(10 slots)
	`LDEV=F02,00,03,03,24,24,01,10,10,S=1,R=2`	(10 slots)
	`LDEV=F03,00,03,03,12,12,01,10,10,S=2,R=1`	(10 slots)
	`LDEV=F04,00,03,03,24,24,01,10,10,S=2,R=2`	(10 slots)

The LDEV=Fxx statement for an ABBA/1 system.

ABBA/1:	`LDEV=F01,00,03,03,08,08,20,25,25,S=1,R=1`	(6 slots)

The LDEV=Fxx statement for an AML/J system.

AML/J:	`LDEV=F01,00,03,03,01,01,10,10,10,S=1,R=1`	(1 slot)

LDEV statement changes demand correction (generation) of the HCC Archive mirror (refer to HAA).

This definition must match the AR/AMU[1)] definitions.

[1)] AR/AMU = Archive computer ABBA/1 or Archive Management Unit AML/2, AML/E, AML/J

2.10 LDEV=Ixx (CHANGED)

AML COMPONENTS – INSERT AREA	(mandatory)
LDEV=Ixx,00,01,01,row1,row2,pos1,pos2,slots[,S=sys][,R=rob]	

Ixx	Component number (insert area)		
row1/row2	First/last row in a segment		
pos1/pos2	First/last position in a row		
slots	Number of slots in a row		(01-25)
sys	1	1. AML System	(Default)
	2	2. AML System	
rob	1	Robot 1	(Default)
	2	Robot 2	

The LDEV=Ixx statement defines the coordinates of the insert areas in an AML system. A separate LDEV statement is mandatory for each AML unit installed.

Insert areas can overlap.

The LDEV=Ixx statement for 2 twin robot systems with 4 E/I/F storage areas (standard) for AML/2.

AML/2	`LDEV=I01,00,01,01,01,01,01,01,10,S=1,R=1`	(1 slot)
	`LDEV=I11,00,01,01,01,01,01,10,10,S=1,R=1`	(10 slots)
	`LDEV=I91,00,01,01,01,11,01,10,10,S=1,R=1`	(110 slots)
	`LDEV=I02,00,01,01,13,13,01,01,10,S=1,R=2`	(1 slot)
	`LDEV=I12,00,01,01,13,13,01,10,10,S=1,R=2`	(10 slots)
	`LDEV=I92,00,01,01,13,23,01,10,10,S=1,R=2`	(110 slots)
	`LDEV=I03,00,01,01,01,01,01,01,10,S=2,R=1`	(1 slot)
	`LDEV=I13,00,01,01,01,01,01,10,10,S=2,R=1`	(10 slots)
	`LDEV=I93,00,01,01,01,11,01,10,10,S=2,R=1`	(110 slots)
	`LDEV=I04,00,01,01,13,13,01,01,10,S=2,R=2`	(1 slot)
	`LDEV=I14,00,01,01,13,13,01,10,10,S=2,R=2`	(10 slots)
	`LDEV=I94,00,01,01,13,23,01,10,10,S=2,R=2`	(110 slots)

The LDEV=Ixx statement for an ABBA/1 system.

ABBA/1	`LDEV=I01,00,01,01,01,04,01,25,25,S=1,R=1`	(100 slots)
	`LDEV=I11,00,01,01,04,04,01,25,25,S=1,R=1`	(25 slots)

The LDEV=Ixx statement for an AML/J system with 2 E/I/F storage areas.

AML/J	`LDEV=I01,00,01,01,01,01,01,10,10,S=1,R=1`	(10 slots)
	`LDEV=I11,00,01,01,01,01,01,01,10,S=1,R=1`	(1 slot)
	`LDEV=I02,00,01,01,02,02,01,10,10,S=1,R=1`	(10 slots)

LDEV statement changes demand correction (generation) of the HCC Archive mirror (refer to HAA).

This definition must match the AR/AMU[1)] definitions.

[1)] AR/AMU = Archive computer ABBA/1 or Archive Management Unit AML/2, AML/E, AML/J

2.11 LDEV=Wxx (CHANGED)

AML COMPONENTS - CP[1]	(mandatory)
LDEV=Wxx,00,00,00,00,00,pos1,pos2,slots[,S=sys][,R=rob]	

Wxx	Component number of the Problem box (cartridge pocket)		
slots	Number of slots in a row		(01)
sys	1	1. AML System	(Default)
	2	2. AML System	
rob	1	Robot 1	(Default)
	2	Robot 2	

The LDEV=Wxx statement defines the coordinates of the Problem box (cartridge pocket) in an AML system. A separate LDEV statement is mandatory for each AML unit installed.

Examples:

```
LDEV=W01,00,00,00,00,00,01,01,01,S=1,R=1
LDEV=W02,00,00,00,00,00,02,02,01,S=1,R=2
LDEV=W03,00,00,00,00,00,01,01,01,S=2,R=1
LDEV=W04,00,00,00,00,00,02,02,01,S=2,R=2
```

The AR[2] or AMU[2] manage the slots of the Problem box and therefore only one slot may be defined per Problem box.

LDEV statement changes demand correction (generation) of the HCC Archive mirror (refer to HAA). This definition must match the AR/AMU[1] definitions.

[1] CP = Problem box (cartridge pocket)
[2] AR/AMU = Archive computer ABBA/1 or Archive Management Unit AML/2, AML/E, AML/J

2.12 MSG=CBR4405D (NEW)

AUTOMATIC MESSAGE REPLY	(optional)
MSG=CBR4405D,AUTOREPLY MSG=CBR4405D,NOAUTOREPLY	

NOAUTOREPLY	HCC ignores a pending reply
AUTOREPLY	HCC answers a pending reply with the A-volser of the optical disk (OD) loaded.

This statement handles the automatic message processing for the reply CBR4405D.

Default: `MSG=CBR4405D,NOAUTOREPLY`

Example: `MSG=CBR4405D,AUTOREPLY`

This statement must be set to AUTOREPLY when automatic optical disk initialization under HCC control is desired.

2.13 MSG=CBR4406D (NEW)

AUTOMATIC MESSAGE REPLY	(optional)
MSG=CBR4406D,AUTOREPLY MSG=CBR4406D,NOAUTOREPLY	

NOAUTOREPLY	HCC ignores a pending reply
AUTOREPLY	HCC answers a pending reply with the Owner information.

This statement handles the automatic message processing for the reply CBR4406D.

The Owner information has the following format:

```
HACC Hx ON CPUx
```

HCC	Constant
Hx	HCC ID from Start procedure
ON	Constant
CPUx	SMF CPU ID

Example: HACC H2 ON S500

Default: `MSG=CBR4406D,NOAUTOREPLY`

Example: `MSG=CBR4406D,AUTOREPLY`

This statement must be set to AUTOREPLY when automatic optical disk initialization under HCC control is desired.

2.14 MSG=CBR4423D (NEW)

AUTOMATIC MESSAGE REPLY	(optional)
MSG=CBR4423D,AUTOREPLY MSG=CBR4423D,NOAUTOREPLY	

NOAUTOREPLY	HCC ignores a pending reply
AUTOREPLY	HCC answers a pending reply with the home coordinate of the A-volser of the optical disk (OD) loaded.

This statement handles the automatic message processing for the reply CBR4423D.

Default: `MSG=CBR4423D,NOAUTOREPLY`

Example: `MSG=CBR4423D,AUTOREPLY`

This statement must be set to AUTOREPLY when automatic optical disk initialization under HCC control is desired.

2.15 MSG=CBR4424D (NEW)

AUTOMATIC MESSAGE REPLY	(optional)
MSG=CBR4424D,AUTOREPLY MSG=CBR4424D,NOAUTOREPLY	

NOAUTOREPLY HCC ignores a pending reply
AUTOREPLY HCC answers a pending reply with U (=USE).

This statement handles the automatic message processing for the reply CBR4424D.

Default: `MSG=CBR4424D,NOAUTOREPLY`

Example: `MSG=CBR4424D,AUTOREPLY`

This statement must be set to AUTOREPLY when automatic optical disk initialization under HCC control is desired.

2.16 MSG=CBR4432D (NEW)

AUTOMATIC MESSAGE REPLY	(optional)
MSG=CBR4432D,AUTOREPLY MSG=CBR4432D,NOAUTOREPLY	

NOAUTOREPLY HCC ignores a pending reply
AUTOREPLY HCC answers a pending reply with the assigned Storage group.

This statement handles the automatic message processing for the reply CBR4423D.

Default: `MSG=CBR4432D,NOAUTOREPLY`

Example: `MSG=CBR4432D,AUTOREPLY`

This statement must be set to AUTOREPLY when automatic optical disk initialization under HCC control is desired.

Further information can be found in the description of the HACCPARM statement STORGRP.

2.17 MSG=CBR4438D (NEW)

AUTOMATIC MESSAGE REPLY	(optional)
MSG=CBR4438D,AUTOREPLY MSG=CBR4438D,NOAUTOREPLY	

NOAUTOREPLY HCC ignores a pending reply
AUTOREPLY HCC answers a pending reply with F (=FORMAT).

This statement handles the automatic message processing for the reply CBR4438D.

Default: `MSG=CBR4438D,NOAUTOREPLY`

Example: `MSG=CBR4438D,AUTOREPLY`

This statement must be set to AUTOREPLY when automatic optical disk initialization under HCC control is desired.

2.18 MSG=CBR4439D (NEW)

AUTOMATIC MESSAGE REPLY	(optional)
MSG=CBR4439D,AUTOREPLY MSG=CBR4439D,NOAUTOREPLY	

NOAUTOREPLY	HCC ignores a pending reply
AUTOREPLY	HCC answers a pending reply with the B-volser of the optical disk (OD) loaded.

This statement handles the automatic message processing for the reply CBR4439D.

Default: `MSG=CBR4439D,NOAUTOREPLY`

Example: `MSG=CBR4439D,AUTOREPLY`

This statement must be set to AUTOREPLY when automatic optical disk initialization under HCC control is desired.

2.19 ODSCR (NEW)

INFORMATION ON OPTICAL DISK	(optional)
ODSCR TYPE=SEQ,DSN=file-name	

SEQ Sequential input file
file-name Name of the input file

This parameter describes the method and input file to retrieve information on an optical disk using FREESPACE.

The volser of the A or B side can be entered.

Default: None

Example: `ODSCR TYPE=SEQ,DSN=HACC.DB2.OD#FSPC`

This parameter is only allowed for optical disks (OD).

2.20 PDSNEXCn (CHANGED)

FILE NAME FOR EXCHANGE LOG (PRODUCTION)	(optional)
PDSNEXCn=file-name[,RECNO=xxxx][,STATUS=ccccc]	

n	1-F (1, 2, ..., 9, A,...,F)		
file-name	Name of the HCC Exchange log		
xxxx	Record count		(Default = 300)
cccc	ACT =	Exchange log will be activated	(Default)
	INACT =	Exchange log will not be activated	

This parameter names the HCC Exchange log.

The HCC Exchange log is a BDAM file. Every secondary HCC must be connected to a primary HCC using SHARED DASD.

Exchange logs can be subsequently activated or deactivated using the **'VARY ACT/INACT'** command.

Refer to the HCC Work Files Section for further information.

Default: `PDSNEXCn=file-name,RECNO=300,STATUS=ACT`

Example:
```
PDSNEXC1=hlq.HACCEXC1,RECNO=500,STATUS=INACT
PDSNEXC2=hlq.HACCEXC2
```

The allocation must be checked again when RECNO in the HACCPARM is modified for an existing file.

FEX=(n,...,n) (format EXC.LOG) must be specified in the Start procedure for a HCC restart when EXANGELOG is added to the HACPARMx. The FEX=(n,...,n) parameter must then be deleted afterwards.

2.21 PRTY MO (CHANGED)

BASIC PRIORITY FOR MOUNT TASK	(optional)
PRTY MO=nn	

nn Valid range 01-80

The **PRTY MO** statement supports individual priority control for **MOUNT** tasks.

The specified priority is also used as basic priority for commands to the IBM 3995 - Jukebox (**LJB**, **UJB** and **Flip**).

This value can be modified dynamically with the **SET PRTY** command (valid until HCC Stop).

The priority for tasks in the wait queue is changed according to the HCC internal dispatching algorithms.

Default: `PRTY MO=35`

Example: `PRTY MO=40`

The MO/KE priorities must be "tuned" to attain balanced MOUNT/KEEP processing on AML systems with different HOST complexes.

Recommended:

MAJOR-HCC:
```
PRTY MO=35
PRTY KE=31
```

MINOR-HCC:
```
PRTY MO=45
PRTY KE=41
```

2.22 STORGRP (NEW)

ASSIGN VOLSER MASKS TO STORAGE GROUPS (optional)
STORGRP NAME=name,VOL=(volser1,volser2,...)

name Name of the Storage group (max. 8 chars.)
volserx Volser masks to be assigned to a Storage group. Wildcards such as % or * can be used in accordance with the usual rules. Up to 32 volser masks can be specified per statement.

The **STORGRP** statement serves to assign volser ranges to a Storage group. Assignment is mandatory for automatic optical disk initialization. Unique assignment is also mandatory to execute "scratch mounts" for a certain Storage group.

The statement sequence is relevant.

The statements with the highest qualification degree must be entered first, for example, fully qualified volsers.

Example: Existing volser ranges and desired assignment to Storage groups:

Total volsers:		OD001A/B - OD144A/B
		OD200A/B - OD344A/B
Name: STG001	Volsers:	OD001A/B - OD002A/B
Name: STG002	Volsers:	OD003A/B - OD009A/B
		OD100A/B - OD144A/B
Name: STG003	Volsers:	OD010A/B - OD049A/B
Name: STG004	Volsers:	OD050A/B - OD099A/B
Name: STG005	Volsers:	All other volsers
		OD200A/B - OD344A/B

The following **STORGRP** statements result:

```
STORGRP NAME=STG001,VOL=(OD001*,OD002*)
STORGRP NAME=STG002,VOL=(OD00*,OD1*)
STORGRP NAME=STG003,VOL=(OD01*,OD02*,OD03*,OD04*)
STORGRP NAME=STG004,VOL=(OD0*)
STORGRP NAME=STG005,VOL=(*)
```

2.23 TDSNxxxx (DELETED)

The following HCC/MVS 2.4.0 statements used for test purposes have been deleted for HCC/MVS 3.0.0:

- TDSNARCH
- TDSNEXCx
- TDSNMLGx
- TDSNULOG

2.24 TMSEXIT (CHANGED)

TMS USER EXIT	(optional)
TMSEXIT=modulname,fromvol,tovol[,nnnnn]	

modulname	Load module name that **must** be 8 characters long
fromvol	Lowest VOLSER number
tovol	Highest VOLSER number, whereby fromvol to tovol must be in ascending sequence (alphanumeric as well).
nnnnn	Maximum number of selectable scratch tapes (Default = 3000)

This parameter controls the user exit for scratch tape selection. The user must customize the user exit to the installation environment. PDSNSTMS identifies the input file for scratch tape information.

This exit is called when a HCC **FREEVOL** command is executed.

Suboperands formvol, tovol define the range for the volser to be selected and nnnnn the maximum number of scratch volsers to be selected.

TMSEXIT and TMS=xxxx statements are mutually exclusive; only one of them may be defined.

Refer to the SCRATCH TAPE PROCESSING and USER EXITS Sections in the System Reference Guide for further information as well as to examples in the hlq.ZHC....SAMP library.

Default: None

Example: `TMSEXIT=USEREXIT,AAAAAA,999999,9500`

The module names of the default exits have been changed; the module names should not start with HLQs, HAC or ZHC because these are reserved for the HCC modules.

2.25 UNIT TYPE=CAR (CHANGED)

UNIT DESCRIPTION	(mandatory)
UNIT=cccccccc,nn,s,r[,ALTCUU=cuu2][,FPMALOC=coor1,coor2][,TYPE=CAR]	

cccccccc	MVS unit identification	
	MVS unit address	(0001-FFFF)
nn	AML unit ID	(01-99)
s	AML System number	(1-2)
r	Servicing robot	(1-2)
cuu2	Alternative MVS unit address for tape units	(0001-FFFF)

FPMA[1] assignment for tape units:

coor1	Start coordinate
coor2	End coordinate

CAR	Tape unit	(Default)

The **UNIT** statement TYPE=CAR assigns IBM 3480/90 cartridge stations (and compatible units) to AML systems and the respective robots.

The specified MVS unit addresses (cuu and device) must be defined in all MVS systems involved.

AML unit IDs must
- start at 01,
- be in ascending sequence and
- contiguous.

ALTCUU supports assigning a second MVS unit addresses to another CPU of the same physical unit when the generation differs.

FPMALOC assigns a cartridge station a defined FPMA range.

[1] FPMA = Fixed Preferred MOUNT/KEEP Area

Examples:

```
UNIT=04A0,01,1,1,FPMALOC=03020101,03020215
UNIT=04A1,02,1,1
UNIT=04A2,03,1,1
UNIT=04A3,04,1,1
UNIT=0480,05,1,2,ALTCUU=0AA0,FPMALOC=03010801,03010825
UNIT=0481,06,1,2,ALTCUU=0AA1
UNIT=0920,07,2,1
UNIT=0921,08,2,1
```

This example describes two AML systems:

- 1 twin robot system
- 1 single robot system

Perform the following when UNIT statements are changed:

- HAA GENERATE COORDINATE RECORDS FROM HACPARM1
- HCC Start with SSI=YES and FMTU=Y (format UNIT-LOG)

2.26 UNIT TYPE=IOS (NEW)

UNIT DESCRIPTION	(mandatory)
UNIT=cccccccc,nn,s,r[,TYPE=IOS][,CAP=xxx]	

cccccccc	Library name for optical disks	
nn	AML unit ID	(01-99)
s	AML System number	(1-2)
r	Servicing robot	(1-2)
IOS	Input/Output Station	
xxx	Maximum unit capacity Only valid for TYPE=IOS	(Default = 144)

The **UNIT** statement TYPE=IOS assigns IBM 3995 jukeboxes to AML systems and the respective robots.

AML unit IDs must
- start at 01,
- be in ascending sequence and
- contiguous.

Example: `UNIT=ODLIB1,09,2,1,TYPE=IOS,CAP=144`

Perform the following when **UNIT** statements are changed:

- HAA GENERATE COORDINATE RECORDS FROM HACPARM1
- HCC Start with SSI=YES and FMTU=Y (format UNIT-LOG))

2.27 UNIT TYPE=OAD (NEW)

UNIT DESCRIPTION	(mandatory)
UNIT=cccccccc,nn,s,r[,TYPE=OAD]	

cccccccc	Drive name for optical disks	
nn	AML unit ID	(01-99)
s	AML System number	(1-2)
r	Servicing robot	(1-2)

OAD Operator Accessible Drive

The **UNIT** statement TYPE=OAD assigns the external drives on IBM 3995 jukeboxes to AML systems and the respective robots.

AML unit IDs must
- start at 01,
- be in ascending sequence and
- contiguous.

Example: `UNIT=ODDRV5,10,2,1,TYPE=OAD`

Perform the following when UNIT statements are changed:

- HAA GENERATE COORDINATE RECORDS FROM HACPARM1
- HCC Start with SSI=YES and FMTU=Y (format UNIT-LOG)

3 START PROCEDURE CHANGES

This Section describes the START procedure parameters for HCC/MVS 3.0.0. Parameters deleted from HCC/MVS 2.4.0 are also listed.

3.1 START COMMAND

A Started Task is activated on each HOST system involved.

The Started Task name must not match the subsystem names (IEFSSNxx) otherwise the subsystem will be started under MSTR (Master) instead of the primary subsystem (JES).

Recommended:	Started Task name:	HCC
	SSI name:	HAC0

Parameter SYS=PRI/SECn controls the selection of the primary or secondary systems, and therefore this parameter is already specified in the Start procedure in the SYS1.PARMLIB for each system.

HCC can be started automatically after IPL by entering the following command in SYS1.PARMLIB(COMMNDxx):

```
S HACC.A,HOT=N
```

Start command parameter description:

SYS	=	PRI	Primary Started Task (HACC0), default value
		SECn	Secondary Started Task with n=1-9/A-F (HACC1)
FMTU	=	Y	Format the UNIT Statistic and Recovery logs
	=	N	Default value
FEX	=	(n,...,n)	Format the Exchange logs n=1-9/A-F FEX=() when no Exchange log is to be formatted
HOT	=	Y	Warm start, recovery from last HCC start
	=	N	Cold start, ignoring information from UNIT log and RCA.
SSI	=	Y	Recreate HACCVT, SSI matrix and SSI module
	=	N	Default value
HID	=	Hn	HACC0 identifier (n=1-9)
MID	=	M1	MAJOR name
A1ID	=	An	AML System 1 identifier (n=1)
A2ID	=	Am	AML System 2 identifier (m=2)
M1	=	Hn	Definition of **one** HCC as MAJOR
SIM	=		Parameter deleted
TEST	=		Parameter deleted

Defaults:

A1ID = A1

FEX=(n), FMTU=Y only valid for a PRIMARY SYSTEM start.

START command for single host complex:

```
S HACC.A,SYS=PRI/SECn
         ,FMTU=Y/N
         ,FEX=(n,...,n)
         ,HOT=Y/N
         ,SSI=Y/N
         ,SUP=Y/N
         ,HID=Hn
         ,A1ID=An
         ,A2ID=Am
```

Entering FMTU=Y and/or FEX=(n,...,n) can lead to data losses in productive operation in certain circumstances.

START command for MULTI-HOST complex with **LU 6.2** inter-HOST-communication:

```
S HACC.A,SYS=PRI/SECn
         ,FMTU=Y/N
         ,FEX(n,...,n)
         ,HOT=Y/N
         ,SSI=Y/N
         ,HID=Hn
         ,MID=M1
         ,M1=Hn
         ,A1ID=An
         ,A2ID=Am
```

3.2 STARTED TASK JCL FOR SINGLE HOST COMPLEX

The following JCL statements serve as an example for a HCC Start procedure which must be stored in this form in the SYS1.PROCLIB or an installation-specific PROCLIB. This example is stored under HACCSTC1 in the hlq.ZHC....SAMP library.

```
//*-------------------------------------------------------------------*
//*              HACC STARTED TASK HACC REL. 3.0.0                    *
//*              SINGLE HOST COMPLEX                                  *
//*              AR/AMU VIA EXCP/VTAM LU2/VTAM LU6.2                  *
//*-------------------------------------------------------------------*
//HACC     PROC SYS=PRI,
//             FMTU=N,
//*            FEX=(1,2,...,A,...,F)   <--- ACTIVATE IF NEEDED
//             HOT=Y,
//             SSI=N,
//             SUP=N,
//             HID=H1,
//             A1ID=A1
//*            A2ID=A2                 <--- ACTIVATE FOR SECOND ABBA SYSTEM
//HACCPROC EXEC PGM=ZHC00100,
//             TIME=1440,
//             DPRTY=(14,7),           <--- HIGHER THAN BATCH 1)
//             REGION=4000K,
//             PARM=('SYS=&SYS',
//             'FMTU=&FMTU',
//*            'FEX=&FEX',             <--- ACTIVATE IF NEEDED
//             'HOT=&HOT',
//             'SSI=&SSI',
//             'SUP=&SUP',
//             'HID=&HID',
//             'A1ID=&A1ID')
//*            'A2ID=&A2ID')           <--- ACTIVATE FOR SECOND ABBA SYSTEM
//STEPLIB  DD  DSN=hlq.ZHC300.LOAD,DISP=SHR
//HACCPARM DD  DSN=parmdsn(member),DISP=SHR
```

1) HCC is mainly in WAIT status at the subsystem interface and not using system resources, and therefore it should be given the highest possible priority. This ensures that HCC does not run into a time-out when, for example, a system loop occurs.

3.3 STARTED TASK JCL FOR MULTI HOST COMPLEX (LU 6.2)

The following JCL statements serve as an example for a HCC Start procedure which must be stored in this form in the SYS1.PROCLIB or an installation-specific PROCLIB. This example is stored under HACCSTC2 in the hlq.ZHC....SAMP library.

```
//*-------------------------------------------------------------------*
//*                HACC STARTED TASK HACC RELASE 3.0.0                *
//*                MULTI HOST COMPLEX LU6.2                           *
//*                AR/AMU VIA EXCP/VTAM LU2/VTAM LU6.2                *
//*-------------------------------------------------------------------*
//HACC     PROC SYS=PRI,
//              FMTU=N,
//*             FEX=(1,2,...,A,...,F)   <--- ACTIVATE IF NEEDED
//              HOT=Y,
//              SSI=N,
//              SUP=N,
//              HID=H1,
//              A1ID=A1,
//*             A2ID=A2,                <--- ACTIVATE FOR SECOND ABBA SYSTEM
//              MID=M1,
//              M1=H1
//HACCPROC EXEC PGM=ZHC00100,
//              TIME=1440,
//              DPRTY=(14,7),           <--- HIGHER THAN BATCH 1)
//              REGION=4000K,
//              PARM=('SYS=&SYS',
//              'FMTU=&FMTU',
//*             'FEX=&FEX',             <--- ACTIVATE IF NEEDED
//              'HOT=&HOT',
//              'SSI=&SSI',
//              'SUP=&SUP',
//              'HID=&HID',
//              'A1ID=&A1ID',
//*             'A2ID=&A2ID',           <--- ACTIVATE FOR SECOND ABBA SYSTEM
//              'MID=&MID',
//              'M1=&M1')
//STEPLIB  DD  DSN=hlq.ZHC300.LOAD,DISP=SHR
//HACCPARM DD  DSN=parmdsn(member),DISP=SHR
```

1) HCC is mainly in WAIT status at the subsystem interface and not using system resources, and therefore it should be given the highest possible priority. This ensures that HCC does not run into a time-out when, for example, a system loop occurs.

3.4 START AFTER INSTALLATION

Files such as UNIT-LOG (and EXCHANGE-LOGs when defined) and the subsystem interface must be formatted during the first primary HCC start. The MESSAGE-LOGs need not be formatted.

Start command format:

```
S HACC.A,FMTU=Y,SSI=Y[,FEX=(n,...,n)]
```

The following operator message appears, together with other information, when HCC is initialized:

```
HAC000I ==================================================
HAC016I ===       HACC READY FOR COMMUNICATION         ===
HAC000I ==================================================
```

Basic HCC Documentation.

4 OPERATOR COMMANDS CHANGES

This Section describes the new and changed operator commands as compared to HCC/MVS 2.4.0. Statements deleted from HCC/MVS 2.4.0 are also listed.

4.1 ACC - ARCHIVE COMPARISON USING COORDINATES (CHANGED)

```
ACC [s,] coor1-coor2[,D][,SKIP=NO/YES]
ACC [s,] COOR=coor..[,D][,SKIP=NO/YES]
ACC [s,] STOP
```

s	= AML System	Default: s = 1
D	= Display information	Default: only for deviations
COOR=coor	= Generic coordinate 1 - 8 characters	
coor1-coor2	= Coordinate range	
SKIP	= Coordinate increment	
YES	= Process every 2nd coordinate For cartridges, process only every 2nd medium. For optical disks, generate 1 command for each medium.	
NO	= Process all coordinates For cartridges, generate 1 command for each medium. For optical disks, generate 2 commands for each medium.	(Default)
STOP	= Stop Archive comparison	Default: end of range specified

Archive information such as volser, status and coordinates are maintained on both the HOST and AML level.

This information can be compared with both the ACC and ACV functions and differences displayed.

It is recommended to start the HCC message log (refer to LOGSTART) when comparing larger archive ranges.

Inconsistencies can be displayed when drive coordinates are used (drive in use) and therefore this command should not be used with drive coordinates.

Any necessary corrections can be made with the **UPC** or **UPV** commands.

Example:

```
ACC COOR=01010101
```

Compare status and volsers of all coordinates in tower 02.

```
HAC000I
=========================================================
HAC036I COMMAND=ULC SQNR=2506
HAC036I ===== VOLSER SYS HOMECOOR TEMPCOOR STATUS
HAC036I ABBA: 001001  1  01010101         M,E,-
HAC036I HACC: 001001  1  01010101         M,B,N
HAC036I DIFF:                               *
HAC000I
=========================================================
HAC231I ARCHIVE COMPARE ENDED, LAST VOLSER/COORD=01010101
HAC232F NO.OF COMPARED VOLSER/COORDS=00001,NO.OF DIFF=00001
```

Volume status:

Position 1:	0	= Special status (AR/AMU)
	C	= Cleaning cartridge
	M	= Data cartridge
	O	= Optical disk
Position 2:	B	= In Archive
	E	= Ejected
	J	= Optical disk in the jukebox (phys. status = O)
	L	= Initial (phys. status = 0)
	M	= Loaded
	O	= ‘Other side’ of the optical disk loaded (phys status = O)
Position 3:	N	= Marked as no-scratch (cataloged)
	S	= Marked as scratch
Position 4:	I	= Volume in send wait queue
Position 5:	D	= Volume in use (for example, loaded on unit)

4.2 ACV - ARCHIVE COMPARISON USING VOLSER (CHANGED)

```
ACV [s,]        fromvol-tovol[,D]
ACV [s,]        volser[,D]
ACV [s,]        STOP
```

s	= AML System	Default: s = 1
D	= Display information	Default: only for deviations
volser	= Volser 6 characters	
fromvol-tovol	= Volser range	
STOP	= Stop Archive comparison	Default: end of range specified

A **ULV without UPDATE** is generated from the **ACV** command.

Example:

```
ACV 001001,D
```

Status and coordinate of volser 001001 are compared and displayed.

```
HAC000I ==============================================
HAC036I COMMAND=ULV SQNR=2606
HAC036I ===== VOLSER SYS HOMECOOR TEMPCOOR STATUS
HAC036I ABBA: 001001  1  01010101          M,E,-
HAC036I HACC: 001001  1  01010101          M,B,N
HAC036I DIFF:                                 *
HAC000I ==============================================
```

Volume status:

Position 1:	0	= Special status (AR/AMU)
	C	= Cleaning cartridge
	M	= Data cartridge
	O	= Optical disk
Position 2:	B	= In Archive
	E	= Ejected
	J	= Optical disk in the jukebox (phys. status = O)
	L	= Initial (phys. status = 0)
	M	= Loaded
	O	= ‘Other side’ of the optical disk loaded (phys status = O)
Position 3:	N	= Marked as no-scratch (cataloged)
	S	= Marked as scratch
Position 4:	I	= Volume in send wait queue
Position 5:	D	= Volume in use (for example, loaded on unit)

4.3 ADD - ADD VOLSERS FOR ROBOT FUNCTIONS (DELETED)

This HCC/MVS 2.4.0 operator command was deleted in HCC/MVS 3.0.0.

4.4 ALLOC - RESERVING TAPE UNITS (CHANGED)

```
ALLOC cuu
```

cuu = MVS unit address (3 or 4 digits)

ALLOC permanently reserves the tape unit specified under cuu for a HCC subtask. Up to 4 units can be assigned in parallel.

ALLOC is precondition for the following functions:

- Header check (refer to the corresponding commands **LCK, SLCK, PLCK, DRT**)
- Header initialization (refer to the corresponding commands **TLI, STLI, PTLI, TLIPW**)

Restriction: Only one of these functions can be active at any one time.

FREE releases a tape unit previously reserved with the **ALLOC** command and the corresponding subtask is deleted.

This command is not allowed for optical disks and incorrect use can cause processing errors.

Example:

```
ALLOC 481
ALLOC 0481
```

4.5 CANCEL LABEL - DELETING INACTIVE ENTRIES (NEW)

```
C[ANCEL] LABEL,VOL=volser[,I]
```

volser = A-volser of the process to be deleted
I = Immediate

This command removes inactive entries and terminates the corresponding function. Active processes can only be terminated when the 1st CBR4438D message has not been answered. The **CANCEL** command is ignored when this message has already been answered.

Option I (Immediate) supports immediate termination of active label processing.

VOL= **must** be the A-volser of the respective medium.

This command is only allowed for optical disks (OD) and incorrect use can cause processing errors.

Example:

```
CANCEL LABEL VOL=OD002A
```

Also refer to the **START LABEL** and **DISPLAY LABEL** commands.

4.6 CL - INDIVIDUAL TAPE UNIT CLEANING (CHANGED)

```
CL cuu
CL USE=nnn
```

cuu = MVS unit address (3 or 4 digits)
nnn = Counter (002 to 060)

This command serves to trigger an individual tape unit cleaning process.

Automatic cleaning supported by HCC normally makes this command unnecessary.

CL can be used for additional cleaning when the recommended cleaning interval is too long for bad tape material (refer to the HACPARM1 CLEAN parameter).

When the counter is used (USE count), individual cleaning is triggered for tape units when their counter has been reached or exceeded.

This command is not allowed for optical disks and incorrect use can cause processing errors.

Example:

```
CL 481
CL 0481
CL USE=020
```

4.7 CX CMD - LU 6.2 INTERCOMMUNICATION (CHANGED)

```
CX CMD,Hx,command
```

Hx = HACC0 ID (x=1-9)
command = Any valid (HCC) command

CX CMD sends HCC commands using an existing conversation to another complex where they are executed.

Example:

```
CX CMD,H2,VI
```

Complex H2 is tasked with a **VOLUME INSERT** from any one of the other complexes.

MVS commands can be transferred using an existing conversation.

Entering the following command (for example):

```
CX CMD,H2,OS S TASK1
```

triggers an attempt to start the Started Task "TASK1" on another complex.

4.8 CX GETMAJOR - LU 6.2 MAJOR COMPLEX (CHANGED)

```
CX GETMAJOR
```

CX GETMAJOR uses a HCC table to determine the MAJOR complex defined there.

This command should be used before switching a HOST complex to the MAJOR function (refer to CX SETMAJOR); it can also be used to retrieve information at any time, independent of switching.

Example:

```
CX GETMAJOR
```

The following message appears:

```
HAC349I HACCID Hx IS MAJOR COMPLEX
```

4.9 CX MSG - LU 6.2 SEND MESSAGE (CHANGED)

```
CX MSG,Hx,text
```

Hx = HACC0 ID (x=1-9)
text = Text

CX MSG sends any optional information to another complex through an existing conversation.

The information then appears on the (master) console of the addressed HACC0 complex as follows:

```
HAC036I Hx: MESSAGE TEXT,
```

where Hx (x=1-9) is the sender.

Example:

```
CX MSG,H2,PLEASE INSERT CARTRIDGES
```

Information is sent to the H2 complex from any other complex.

4.10 CX SETMAJOR - LU 6.2 SET MAJOR (CHANGED)

```
CX SETMAJOR[,FORCE]
```

FORCE = Skip the **CX GETMAJOR** and **CX SETMINOR** commands.

CX GETMAJOR must be performed before switching a HOST complex to the MAJOR function in order to use a HCC table to determine the previous, possibly still active, MAJOR complex (Hy).

After successful completion of the **GETMAJOR** command, the **CX SETMAJOR** command triggers the following:

1. The new MAJOR ID is noted in a HCC table and the assignment to the previous MAJOR deleted.

2. The new MAJOR Hx sends **CX SETMINOR** to the previous MAJOR Hy.

3. The new MAJOR Hx passes information on the switch to all partners. This information is stored in a HCC table on each MINOR.

Example:

```
CX GETMAJOR
CX SETMAJOR
```

The following messages appear:

```
HAC349I HACCID Hy IS MAJOR COMPLEX          (y=1-9)
HAC351I Hy IS NOW COMPLEX MAJOR             (y=1-9)
```

4.11 CX SETMINOR - LU 6.2 SET MINOR (CHANGED)

```
CX SETMINOR
```

CX SETMINOR switches an active MAJOR complex to MINOR.

This command is only provided for exceptional situations, for example, when the **SETMINOR** command issued automatically during a **CX SETMAJOR** command did not arrive at the receiver.

This command should be used with utmost care because execution deletes all wait queue information received from MINOR systems.

Example:

```
CX SETMINOR
```

The following message appears:

```
HAC351I Hx IS NOW COMPLEX MINOR          (x=1-9)
```

4.12 CX D - DISPLAY TOKEN RING (DELETED)

This HCC/MVS 2.4.0 operator command was deleted in HCC/MVS 3.0.0.

4.13 CX ECHO - TOKEN RING TEST (DELETED)

This HCC/MVS 2.4.0 operator command was deleted in HCC/MVS 3.0.0.

4.14 CX GETLINE - TOKEN RING COMMUNICATION PATH (DELETED)

This HCC/MVS 2.4.0 operator command was deleted in HCC/MVS 3.0.0.

4.15 CX SHUTDOWN - SHUTDOWN TOKEN RING (DELETED)

This HCC/MVS 2.4.0 operator command was deleted in HCC/MVS 3.0.0.

4.16 CX SWLINE - SWITCH TOKEN RING PATH (DELETED)

This HCC/MVS 2.4.0 operator command was deleted in HCC/MVS 3.0.0.

4.17 DC - DISPLAY COORDINATE (NEW)

```
DC [s,]coor
DC [s,]co**or**[,nnnn][,SKIP=NO/YES]
```

coor	= Valid coordinate (8 byte expression)	
co**or**	= Coordinate "under mask" ,* = wild character	
nnnn	= Number of records to be displayed	Default = 10
s	= AML System number	Default = 1
SKIP	= Coordinate increment	
YES	= Process every 2nd coordinate For cartridges, process only every 2nd medium. For optical disks, generate 1 command for each medium.	
NO	= Process all coordinates (Default) For cartridges, generate 1 command for each medium. For optical disks, generate 2 commands for each medium.	

DC provides information on COORDINATE-VOLSER assignments for various types of coordinate definitions, pointers and status between coordinates and Archive mirror tables.

Example 1:

```
DC 01020304
DC 4*****09,50
DC **02**09,100,SKIP=YES
DC 01******
```

Example 2: Cartridge area display

```
HAC036I   KEY    P HOMECOOR TEMPCOOR VOLSER ST DEV ROB SYS TYPE COMMENT
HAC036I 01010101 Y 01010101 00000000 001001 MB T/R 1 2  1  ---- o.K.
HAC036I 01010102 Y 01010102 00000000 001002 MB T/R 1 2  1  ---- o.K.
HAC036I 01010103 Y 01010103 00000000 001003 ME T/R 1 2  1  ---- o.K.
HAC036I 01010104 Y 01010104 00010104 001004 MB T/R 1 2  1  ---- o.K.
...
HAC036I 01010109 Y 01010109 04010606 001009 MB T/R 1 2  1  FPMA o.K.
HAC036I 01010110 Y 01010110 04010607 001010 MB T/R 1 2  1  FPMA o.K.
```

Example 3: Display coordinates on areas on optical disk

```
HAC036I DC 010201**
HAC360I   KEY    P HOMECOOR TEMPCOOR VOLSER ST DEV ROB SYS TYPE COMMENT
HAC036I 01020101 Y 01020101 A-SIDE   OD001A OM T/R 1    1  ---- O.K.
HAC036I 01020102 Y          B-SIDE   OD001B    T/R 1    1  ---- O.K.
HAC036I 01020103 Y 01020103 A-SIDE   OD002A OJ T/R 1    1  ---- O.K.
HAC036I 01020104 Y          B-SIDE   OD002B    T/R 1    1  ---- O.K.
HAC036I 01020105 Y 01020105 A-SIDE   OD003A OJ T/R 1    1  ---- O.K.
HAC036I 01020106 Y          B-SIDE   OD003B    T/R 1    1  ---- O.K.
HAC036I 01020107 Y 01020107 A-SIDE   OD004A OJ T/R 1    1  ---- O.K.
HAC036I 01020108 Y          B-SIDE   OD004B    T/R 1    1  ---- O.K.
HAC036I 01020109 Y 01020109 A-SIDE   OD005A OJ T/R 1    1  ---- O.K.
HAC036I 01020110 Y          B-SIDE   OD005B    T/R 1    1  ---- O.K.
HAC361I NUMBER OF DISPLAYED COORDINATES: 0010
```

Field description:

KEY: Coordinate or text "NO PTR" when no pointer to the Archive mirror is available

P:
- Y = Valid pointer
- N = Invalid pointer

DEV:
- T/R = Tower/rack
- I = Insert area
- E = Eject area
- F = Foreign mount
- D = Tape unit (device)

TYPE:
- FPMA = Fixed Preferred Mount Area
- HWS = HCC Work Storage
- DYN = Dynamic area

COMMENT:
- o.K. = Valid
- DIFF = Invalid

4.18 DEL - DELETE VOLSER FOR ROBOT FUNCTIONS (DELETED)

This HCC/MVS 2.4.0 operator command was deleted in HCC/MVS 3.0.0.

4.19 DISPLAY - DISPLAY FUNCTIONS (CHANGED)

```
D[ISPLAY]
D[ISPLAY] ACTIVITY
D[ISPLAY] COM[,PATH=pppp][,STATUS=sssss][,HID=hh]
D[ISPLAY] CSA
D[ISPLAY] FUNCTION
D[ISPLAY] JB[,NAME[=jbname]][,DET=Y/N]
D[ISPLAY] LABEL
D[ISPLAY] LOC[ATION] [,SYS=n]
D[ISPLAY] MSG,NUM=nnn/nn*/n*
D[ISPLAY] SESSION
D[ISPLAY] STAT
D[ISPLAY] STORGRP[,NAME=stgname][,DET=Y/N][,NUM=nnn]
D[ISPLAY] USER
```

ACTIVITY	=	Display the number of all robot actions and the time required
COM	=	Display the connections
CSA	=	Display the CSA data
FUNCTION	=	Display all active functions
JB	=	Display all relevant information on jukeboxes
LABEL	=	Display all entries in the ZHCLAB
LOCATION	=	Display the current location of all robots together with the values and coordinate ranges defined in the LDEV/UNIT statements.
MSG	=	Display the HCC message number nnn (generic n* or nn*)
SESSION	=	Display the connection status of all communication partners
STAT	=	Display the time of the first and last contact and the last message received per communication partner (default)
STORGRP	=	Display all defined Storage groups
USER	=	Display all active ABS users

DISPLAY shows information according to the parameters specified.

'STAT' is assumed as parameter when no parameter follows the command word (default).

The parameters ‘LABEL’ and ‘JB’ are only valid for optical disks (OD), and can cause processing errors when used otherwise.

DISPLAY command examples are shown on the following pages.

Example 1: **Display robot functions**

```
D ACTIVITY

HAC450I *-------------------------- ACTIVITY DISPLAY -------------------------*
HAC447I HOUR   SYS1/ROB1      SYS1/ROB2       SYS2/ROB1       SYS2/ROB2
HAC447I        ACT.  TIME     ACT.  TIME      ACT.  TIME      ACT. TIME
HAC405I   00     2  00:17       0  00:00        0  00:00        0 00:00
HAC405I   01     4  00:35       0  00:00        0  00:00        0 00:00
.
.
.
HAC405I *=====================================================================*
HAC405I *SUM ACT.   39              0               0               0
HAC405I *SUM TIME 00:10:20      00:00:00        00:00:00       00:00:00
HAC405I *=====================================================================*
HAC451I *-------------------------- END OF DISPLAY --------------------------*
```

HOUR = Start time, hour interval
ACT = Number of robot activities
TIME = Time the robot was last active in minutes and seconds

Example 2: **Display COM connections**

```
- D COM

  HAC450I *-------------------------- Commun.  Display -----------*
  HACnnnI                             APPC Section
  HAC453I State of ZHC05400 : Active
  HAC454I * Hid Mid Typ Status   LU name  Sendcnv  Recvcnv  Sen...
  HAC405I   H1  M1  Lcl          A5OB6200
  HAC405I   H2  00  Rem Pri Used A5OB6201 xxxxxxx  xxxxxxxx xxx...
  HAC405I   H3  00  Rem Pri Used A5OB6202
  HAC405I   A1  00  Rem Pri Used TOBLI24  nnnnnnn  nnnnnnnn nnn...
  HAC405I   A1  00  Rem Alt      TOBLI24  mmmmmmm  mmmmmmmm mmm...
  HAC405I   No APPC Communications defined
  HACnnnI                             DASD Section
  HACnnnI * Hid Status  Dsname                  ...Reads    Writes
  HAC405I   S1  active  OBI1WD1.EXCLOG1         ...00000000 00000000
  HAC405I   S4  active  OBI1WD1.EXCLOG4         ...00000000 00000000
  HAC405I   S5  active  OBI1WD1.EXCLOG5         ...00000000 00000000
  HAC405I   SB  inact   OBI1WD1.EXCLOGB         ...00000000 00000000
  HAC405I   No EXCLOG Communications defined
  HACnnnI                             LU2  Section
  HACnnnI * Hid   ACB name Luname   Status
  HAC405I   A1    A5OBL200 TOBLI24  Pri Act Used
  HAC405I   A1    A5OBL201 TOBLI25  Pri Act Unused
  HAC405I   No 3270 Communications defined
  HACnnnI                             EXCP Section
  HACnnnI * Hid   CUU-Addr Status
  HAC405I   A2    01CA     Pri Act Used
  HAC405I   A2    01CB     Pri Act Unused
  HAC405I   No 3270 Communications defined
  HAC451I *-------------------------- End of Display --------------*
```

Syntax:

```
D[ISPLAY] COM[,PATH=pppp][,STATUS=sssss][,HID=hh]
```

PATH=APPC	to display connections through LU 6.2.
PATH=3270	to display connections through LU2 / EXCP.
PATH=DASD	to display connections through shared DASD.
PATH=ALL	to display ALL defined connections (default).
STATUS=ACT	to display all active connections.
STATUS=INACT	to display all inactive connections.
STATUS=ALL	to display all defined connections (default).
HID=An	to display the connection to a special AR/AMU. (n=1-9/A-Z)
HID=Hn	to display the connection to a special MVS-PRIMARY-HCC (s = 1-9/A-Z).
HID=Sn	to display the connection to a special MVS-SECONDARY-HCC (s = 1-9/A-F).
HID=ALL	to display all defined partners (default)

Example 3: **Display CSA data**

```
D CSA
HAC450I *-------------------------- CSA DISPLAY ------------------------------*
HAC488I HACCCVT ENTRY ADDRESS ..: 00B36160  SSI-NAME...........: HAC0
HAC489I CSA-SIZE (BYTES HEX/DEC): E4000D64/03428  SUBPOOL-NUMBER: 228
HAC490I SSCVT ADDRESS ..: 00B980A0  SFM EP-ADDRESS.....: 00B36340
HAC491I HACC SYSTEM ....: PRIMARY   HACC ASCB-ADDRESS..: 00F7C200
HAC492I WSECB: 809DF081  WSSYNC: 00000000
HAC493I ERROR COMPCODE: 00000000 PSW-ADDRESS: 00000000 OFFSET: 00000000
HAC494I SFM  WAITCOUNT: 00000 TOTAL EVENTS: 0000004 STATUS: READY
HAC495I BUFNO SYNC-FLD JOBNAME  MESSAGE
HAC405I     1 00000000 OBI1US1A IEC502E K 92B,000300
HAC405I     2 00000000 OBI1US1A IEC501A M 92B,000300,SL,,TEST02,STEP02,O
HAC405I     3 00000000 OBI1US1A IEC502E K 92A,000100
HAC405I     4 00000000 OBI1US1A IEF233A M 92A,000100
HAC405I     5 00000000 OBI1US1A IEF233A M 92A,000100
HAC405I     6 00000000 OBI1US1A IEF233A M 92A,000100
HAC405I     7 00000000 OBI1US1A IEF233A M 92A,000100
HAC405I     8 00000000 OBI1US1A IEF233A M 92A,000100
HAC405I     9 00000000 OBI1US1A IEF233A M 92A,000100
HAC405I    10 00000000 OBI1US1A IEF233A M 92A,000100
HAC496I RECOVERY AREA CONTENTS / LOCATION: 00B36CB0
HAC497I UNIT HC-SQ M/K-SQ JOBNAME  MESSAGE
HAC405I 092A     2      1 OBI1US1A IEF233A M 92A,000100
HAC405I 092A     2      2 OBI1US1A IEC502E K 92A,000100
HAC405I 092B     3      3 OBI1US1A IEC501A M 92B,000300,SL,,TEST02,STEP0
HAC405I 092B     3      4 OBI1US1A IEC502E K 92B,000300
HAC405I 092C     0      0
HAC405I 092C     0      0
HAC451I *------------------------- END OF DISPLAY --------------------------*
```

D CSA displays the 10 latest buffers in the queue filled by the SFM module HAC23012.

Example 4: **Display active functions**

```
D FUNCTION

HAC450I *-------------------------- FUNCTION DISPLAY -------------------------*
HAC456I   FUNCTION TYPE     USERID   ORIGIN   STATUS PRIORITY
HAC405I            DEFAULT  HAC23101 CONSOLE  RUNNING  020
HAC405I   D FUNCTI COMMAND  Y6003_RE ABBASEND RUNNING  000
HAC451I *-------------------------- END OF DISPLAY --------------------------*
```

FUNCTION	= Command which initiated the function
TYPE	= DEFAULT (Default function entry for this TCB) COMMAND (Function initiated by a command)
USERID	= User initiating the function may be a BATCH JOB, TSO, or a HACCTASK
ORIGIN	= Origin of the initiating command. Is used for directing the Output. May be CONSOLE or ABBASEND
STATUS	= Status of the function (running/waiting)
PRIORITY	= Dispatching priority of the originating command

Example 5: **Display jukebox information (only for OD)**

```
D JB

HAC450I *-------------------------- JUKEBOX DISPLAY -------------------------*
HAC500I   JB name  Dev-Num  Capacity  curr.used  Sys  Rob  Status
HAC405I   ODLIB1     D05       144         4      1    2   OK
HAC451I *-------------------------- END OF DISPLAY --------------------------*
```

```
D JB,NAME=ODLIB1,DET=Y

HAC450I *-------------------------- JUKEBOX DISPLAY -------------------------*
HAC500I   JB name  Dev-Num  Capacity  curr.used  Sys  Rob  Status
HAC405I   ODLIB1     D05       144         4      1    2   OK
HAC503I   VOL-A  VOL-B   VOL-A  VOL-B   VOL-A  VOL-B   VOL-A  VOL-B
HAC405I   10076A 10076B  10077A 10077B  10078A 10078B  10080A 10080B
HAC405I   10081A 10081B  10082A 10082B  10083A 10083B  10084A 10084B
HAC405I   --     --      --     --      --     --      --     --
HAC405I   --     --      --     --      --     --      --     --
  .
  .
  .
HAC405I   --     --      --     --      --     --      --     --
HAC405I   --     --      --     --      --     --      --     --

HAC451I *-------------------------- END OF DISPLAY --------------------------*
```

Example 6: **Display all entries in ZHCLAB (only for OD)**

```
D LABEL

HAC450I *------------------------ LABEL DISPLAY ------------------------------*
HAC532I A-VOLS B-VOLS STORGRP HOMECOOR  L-TYPE   DRVNAME  STATUS
HAC405I OD001A OD001B STG001  01030101  3995WORM ODDRV5   MOUNT SEN
HAC405I OD002A OD002B STG001  01030103  3995-133          INITIAL
HAC405I OD003A OD003B STG001  01030105  3995REWR          INITIAL
.
.
.
HAC451I *-------------------------- END OF DISPLAY --------------------------*
```

D LABEL displays all entries in the ZHCLAB. The status of the process currently active is displayed. All inactive processes have status ‘INITIAL’. D LABEL is only allowed for ODs.

Example 7: **Display Robot location / coordinate ranges**

```
D LOC

HAC450I *------------------------ LOCATION DISPLAY ---------------------------*
HAC460I DIFFERENCE BETWEEN HIGHEST AND LOWEST LOCATION IS 45
HAC462I Robot 1 from System 1 is at location      30
HAC461I   ABBA-Unit   Startcoor   Endcoor     Location
HAC405I     T01       01010101    01321810           5
HAC405I     T02       02010101    02321810          15
.
.
.
HAC451I *-------------------------- END OF DISPLAY --------------------------*
```

Example 8: **Display HCC message**

```
D MSG,NUM=401

HAC450I *-------------------------- MESSAGE DISPLAY --------------------------*
HAC477I   Message  Status  Text
HAC405I   HAC401I  DISA    Invalid continuation in Statement:
HAC451I *-------------------------- END OF DISPLAY --------------------------*
```

```
D MSG,NUM=40*

H30450I *-------------------------- Message  Display -------------------------*
H30477I   Message  Status  Text
H30405I   H30040I  ENA     SCRATCH-TAPE UPDATE COMPLETED
H30405I   H30400I  ENA     There is no output buffer available for User:
H30405I   H30401I  ENA     Invalid continuation in Statement:
H30405I   H30402I  ENA     Statement:
H30405I   H30403I  ENA     Continuation must start in Col.16
H30405I   H30404I  ENA     Statement is too long
H30405I   H30405I  ENA
H30405I   H30406I  ENA     Receiver is not defined in HACPARM1   CMD:
H30405I   H30407I  ENA     There is no SEND conversation for xx and CMD:
H30405I   H30408I  ENA     Attach of SEND Subtask for xx failed
H30405I   H30409I  ENA     Open of ACB xxxxxxxx failed. RC is xxxx
H30451I *-------------------------- End of Display --------------------------*
```

D MSG displays the status (refer to **SET MSG** command) and message text (here 401 or 40 and 400-409).

Example 9: **Display connection status**

```
D SESSION

HAC450I *-------------------------- SESSION  DISPLAY -------------------------*
HAC453I * STATE OF ZHC05400 : ACTIVE                                           *
HAC454I * HID MID TYP LUNAME   SENDCNV  RECVCNV  SENDMODE SENDSESS RECVSESS    *
HAC405I * H2  00  REM OBAPPC03                   ________                      *
HAC451I *-------------------------- END OF DISPLAY --------------------------*
```

STATE OF ZHC05400:	= Status of the APPC MAINTASK (ACTIVE / INACTIVE)
HID	= HCC ID
MID	= 00 or ID of the Major complex (M1)
TYP	= LCL / REM Remote or local HCC
LUNAME	= Name defined in VTAM (APPL statement)
SENDCNV	= Conversation ID of the SEND conversation or empty
RECVCNV	= Conversation ID of the RECEIVE conversation
SENDMODE	= Mode name for the SEND SESSION
SENDSESS	= Session ID for Send SESSION
RECVSESS	= Session ID for Receive SESSION

Example 10: **Display first and last contact**

```
D STAT
HAC450I *-------------------------- SOL      DISPLAY -------------------------*
HAC452I HID MID FIRST-CONTACT LAST-CONTACT  MESSAGE HEADER
HAC405I H1  M1  ----- -----   ----- -----   ____________________
HAC405I H2  00  ----- -----   ----- -----   ____________________
HAC451I *-------------------------- END OF DISPLAY ---------------------------*
```

HID	= HCC identifier
MID	= Major identifier
FIRST/LAST-CONT.	= First/last contact with HID
MESSAGE HEADER	= Last message from HID

Example 11: **Display Storage group**

```
D STORGRP
H30450I *-------------------------- Storgrp  Display ------------------------*
H30527I SG name    Mask
H30405I STGGRP01   1007%%
H30405I STGGRP02   OD001A OD001B OD002A OD002B
H30405I STGGRP03   OD00%A OD00%B
H30405I STGGRP04   OD01*
H30405I STGGRP05   *
H30451I *-------------------------- End of Display --------------------------*

D STORGRP,NAME=STGGRP01
H30450I *-------------------------- Storgrp  Display ------------------------*
H30527I SG name    Mask
H30405I STGGRP01   1007%%
H30451I *-------------------------- End of Display --------------------------*

D STORGRP,NAME=STGGRP01,DET=Y
H30450I *-------------------------- Storgrp  Display ------------------------*
H30527I SG name    Mask
H30405I STGGRP01   1007%%
H30405I 10075A  10075B  10076A  10076B  10077A  10077B  10078A  10078B
H30405I 10079A  10079B
H30451I *-------------------------- End of Display --------------------------*

D STORGRP,NAME=STGGRP01,DET=Y,NUM=1
H30450I *-------------------------- Storgrp  Display ------------------------*
H30527I SG name    Mask
H30405I STGGRP01   1007%%
H30405I 10075A  10075B
H30451I *-------------------------- End of Display --------------------------*
```

Example 12: **Display active ABS users**

```
D USER

HAC450I *------------------------- USER     DISPLAY -------------------------*
HAC458I   USER     STARTADDR TEXT
HAC405I   OPERATOR 02929FB4  HAC430I No communication path active for M1.
HAC405I   Y6003§RE 0292FA78  D USER
HAC451I *------------------------- END OF DISPLAY --------------------------*
```

USER	= Userid/Jobname from ABBASEND
STARTADDR	= Start address of the ABS output buffer header
TEXT	= Data from the first entry in the ABS output buffer

4.20 DLC - DOWNLOAD ARCHIVE USING COORDINATES (CHANGED)

```
DLC [s,] coor1-coor2[,ST=E/B][,DAY=yyddd]
DLC [s,] coor1-coor2                      [,SKIP=NO/YES]
DLC [s,] COOR=coor..                      [,SKIP=NO/YES]
DLC [s,] MASK=mm**nn**                    [,SKIP=NO/YES]
DLC [s,] STOP
```

s	= AML System	Default: s = 1
COOR=coor	= Generic coordinate 1 - 8 bytes	
coor1-coor2	= Coordinate range	
ST	= Status: ejected(E) or occupied (B)	
DAY	= Date for E or B	
yyddd	= Date	
MASK	= Selection "under mask"	
mm,nn	= Selected value	
**	= Wildcard	
SKIP	= Coordinate increment	
YES	= Process every 2nd coordinate For cartridges, process only every 2nd medium. For optical disks, generate 1 command for each medium.	
NO	= Process all coordinates	(Default)
	For cartridges, generate 1 command for each medium. For optical disks, generate 2 commands for each medium.	
STOP	= Stop download	Default: end of specified range

DLC transfers data such as volser, status and coordinates of the HCC Archive mirror to the Archive mirror of the AR or AMU.

When coor1-coor2 is specified, the download can be restricted with the ST=E/B and DAY=yyddd parameters. The restrictions are processed as follows:

ST=B	Download all occupied coordinates.
ST=E	Download all empty (ejected) coordinates, all free coordinates in the dynamic area.
ST=B,DAY=yyddd	Download all occupied coordinates where an **INSERT** was made as from the date specified.
ST=E,DAY=yyddd	Download all empty (ejected) coordinates where an **EJECT** was made as from the date specified. Exception: All free coordinates in the dynamic area are copied to the AR/AMU Archive irrespective of the date.
DAY=yyddd	Download all occupied coordinates where a movement, **INSERT** or **EJECT**, was made as from the date specified. Empty coordinates in the dynamic area are ignored.

The restricting parameters are ignored when coor1-coor2 are the same.

The run time must be considered when transferring the complete Archive.

Examples:

```
DLC COOR=0003
```

This command transfers coordinate information on all foreign mount slots to the Archive computer.

```
DLC MASK=01**05**
```

This command transfers information on all segments (**) and all positions (**) in row 05 on tower 01 to the Archive computer (horizontal storage).

Only one **DLC** command should be active at any one time, unexpected results can occur when several **DLC** commands are active simultaneously.

4.21 DLV - DOWNLOAD ARCHIVE USING VOLSER (CHANGED)

```
DLV [s,]      fromvol-tovol
DLV [s,]      volser
DLV [s,]      STOP
```

s	= AML System	Default: s = 1
volser	= VOLSER 6 bytes	
fromvol-tovol	= VOLSER range	
STOP	= Stop download	Default: end of specified range

DLV transfers data such as volser, status and coordinates of the HCC Archive mirror to the Archive mirror of the AR or AMU.

The run time must be considered when transferring the complete Archive.

This command has been extended and now supports using optical disks.

Example:

```
DLV 010001-011000
```

4.22 DRT - DISPLAY ACTIVE ROBOT-FUNCTIONS (CHANGED)

```
DRT
```

DRT displays the active subtasks and the selected robot service functions.

These functions are:

- Tape label check (refer to **LCK**, **SLCK**, **PLCK**)
- Tape label initialization (refer to **TLI**, **STLI**, **PTLI**)

Example:

```
DRT

HAC000I
=======================================================
HAC036I UNIT   FUNCTION    STATUS  VOLSER
HAC036I 04A0   LABELCHECK   BUSY   001005
HAC036I 04A1   LABELCHECK   BUSY   001004
HAC036I 04A7                IDLE   000000
HAC036I                                          LABELCHECK
STARTVOL=001001,CURVOL=001005,ENDVOL=001100
HAC000I
=======================================================
```

A header check is active on units 04A0 and 04A1 whereas 04A7 is reserved by HCC but has no active tasks.

4.23 DSC - DISPLAY SCRATCH DATA MEDIA (CHANGED)

```
DSC [SYS=s,][nnn][,TYPE=CAR]
DSC          1    ,TYPE=OPT
```

s	= AML System number	Default: 1
nnn	= Number of volser numbers to be displayed, display the Scratch pool when 1 and TYPE=CAR	Default: 10
CAR	= Display scratch tapes	Default
OPT	= Display optical disk with **FREESPACE**	

DSC serves to display current lists of scratch data media.

Example 1:

Display scratch tape information

```
DSC

HAC000I ====================================================
HAC036I 001011
HAC036I 001103
HAC036I 001106
HAC036I 001201 INQ
HAC036I ...
HAC036I 002016
HAC036I 003049
HAC247I AVAILABLE SCRATCH TAPES 00657
HAC000I ====================================================
```

Default(s): - SYS=1
- 10 (count)
- TYPE=CAR

Example 2:

Display pool information for tapes

```
  DSC 1

  HAC000I
=====================================================
  HAC036I 001011
  HAC247I AVAILABLE SCRATCH TAPES 00657
  HAC192I  NOSCR=000020    000000-004999    (SCRTCH)
  HAC192I  NOSCR=000020    000000-004999    (PRIVAT)
  HAC192I  NOSCR=000010    010000-011999    (YELLOW)
  HAC192I  NOSCR=000010    012000-012999    (RED   )
  HAC192I  NOSCR=000030    020001-024000    (GREEN )
  HAC000I
=====================================================
```

Default(s): - SYS=1
- TYPE=CAR

Example 3:

Display pool information for optical disks

```
  DSC 2,TYPE=OPT

  HAC000I
=====================================================
  HAC036I W0001A  1
  HAC036I W0002A  1
  HAC247I AVAILABLE SCRATCH OPTS  000002
  HAC041I REQUEST OK
```

Default(s): - SYS=1

4.24 DU - DISPLAY UNIT ACTIVITY (CHANGED)

```
DU [s,r]
DU cuu/drv[,DET=Y/N]
```

s	= AML System number	Default: s=1
r	= Robot number	Default: r=1
cuu/drv	= MVS unit address	
DET=Y	= Detail display	
DET=N	= Standard display	Default

DU provides information on usage of tape units managed by AML. The DET=Y/N option is only allowed in combination with a MVS unit address.

Example:

```
HAC036I DU
HAC000I ==============================================================
HAC036I CUU/ODN  AU CPU  JOBNAME  VOLSER SQNR MOUNTS CLMO  SECS STATE
HAC036I 092A     01                                   0         -
HAC036I 092B     02 S500 *HACC0*  AP0002 0002      1    1     6 MOMI
HAC036I                           AP0002 0003                    KEKT
HAC036I                  *HACC0*  AP0003                         PM
HAC036I 092C     03                                   0         -
HAC036I 092D     04                                   0         -
HAC036I 092E     05                                   0         -
HAC036I 092F     06                                   0         -
HAC036I ODLIB1   07 IOS device (use DISPLAY JUKEBOX command)
HAC036I ODDRV5   08                                   0         -
HAC000I ==============================================================
HAC036I TOTALS:                                     1          6
HAC000I ==============================================================
```

Further examples:

```
HAC036I DU 92B
HAC000I ============================================================
HAC036I CUU/ODN  AU CPU  JOBNAME  VOLSER SQNR MOUNTS CLMO  SECS STATE
HAC036I 092B     02 S500 *HACC0*  AP0002 0002      1    1     6 MOMI
HAC036I                           AP0002 0003                   KEKT
HAC036I                  *HACC0*  AP0003                        PM
HAC000I ============================================================
```

```
HAC036I DU 92B,DET=YES
HAC000I ============================================================
HAC036I CUU/ODN  AU CPU  JOBNAME  VOLSER SQNR MOUNTS CLMO  SECS STATE
HAC036I 092B     02 S500 *HACC0*  AP0003 0004      1    1    45 MOS
HAC000I ============================================================
HAC036I MMSG=1 MTDB=1 MRQE=1 MTOU=0 MASE=1  MAP=0 MCLA=0  MAN=0
HAC036I UNLI=0 UNLT=0 NHAC=0 MIND=0        UCBI=0   FM=0 JES3=0
HAC036I MSRQ=0 MSCR=0 LSCR=0 MVUA=0
HAC036I KMSG=0 KTBD=0 KRQE=0        KASE=0  KAP=0         KAN=0
HAC036I PMSG=0 PTBD=0 PNHAC=0 PFM=0
HAC000I ============================================================
```

```
HAC036I DU ODDRV5
HAC000I ============================================================
HAC036I CUU/ODN  AU CPU  JOBNAME  VOLSER SQNR MOUNTS CLMO  SECS STATE
HAC036I ODDRV5   08               AP0002                0       MOM
HAC000I ============================================================
```

Further examples:

```
HAC036I DU ODDRV5,DET=YES
HAC000I ================================================================
HAC036I CUU/ODN  AU CPU  JOBNAME  VOLSER SQNR MOUNTS CLMO  SECS STATE
HAC036I ODDRV5   08              AP0002                0       MOM
HAC000I ================================================================
HAC036I MMSG=0 MTDB=0 MRQE=0 MTOU=1 MASE=0  MAP=0 MCLA=0  MAN=0
HAC036I UNLI=0 UNLT=0 NHAC=0 MIND=0        UCBI=0   FM=0 JES3=0
HAC036I MSRQ=0 MSCR=0 LSCR=0 MVUA=0
HAC036I KMSG=0 KTBD=0 KRQE=0        KASE=0  KAP=0         KAN=0
HAC036I PMSG=0 PTBD=0 PNHAC=0 PFM=0
HAC000I ================================================================
```

```
HAC036I DU ODLIB1
HAC000I ================================================================
HAC036I CUU/ODN  AU CPU  JOBNAME  VOLSER SQNR MOUNTS CLMO  SECS STATE
HAC036I ODLIB1   07 IOS device (use DISPLAY JUKEBOX command)
HAC000I ================================================================
```

```
HAC036I DU ODLIB1,DET=YES
HAC000I ================================================================
HAC036I CUU/ODN  AU CPU  JOBNAME  VOLSER SQNR MOUNTS CLMO  SECS STATE
HAC036I ODLIB1   07 IOS device (use DISPLAY JUKEBOX command)
HAC000I ================================================================
HAC036I MMSG=0 MTDB=0 MRQE=0 MTOU=0 MASE=0  MAP=0 MCLA=0  MAN=0
HAC036I UNLI=0 UNLT=0 NHAC=0 MIND=0        UCBI=0   FM=0 JES3=0
HAC036I MSRQ=0 MSCR=0 LSCR=0 MVUA=0
HAC036I KMSG=0 KTBD=0 KRQE=0        KASE=0  KAP=0         KAN=0
HAC036I PMSG=0 PTBD=0 PNHAC=0 PFM=0
HAC000I ================================================================
```

MOUNTS:	Number of mounts since the last **CC** command	
CLMO:	Number of mounts since the last unit cleaning	
SECS:	Average AML response time	
MK:	QSM	= Mount status
	QSK	= Keep status
	I	= Individual mount
	PM	= Pending mount
	Q	= In wait queue
	S	= Sent
	M	= Tape loaded
	F	= Foreign mount

4.25 DV - DISPLAY VOLSER INFORMATION (CHANGED)

```
DV volser
```

volser = Valid VOLSER from the AML library
additional display of the 2nd volser

DV displays significant information concerning a data medium from the HCC Archive.

Example 1: Display tape cartridge

```
DV 004711
 ==========================================================
 VOLSER=004711,HOME=02291401,TEMP=10020311,RECV=02201401
 SYS=1 ROB=1        LAST-UNIT=092B      LAST-JOB=RZ17A001
 STATUS=MB,HOME-POS      SCRATCH
 LAST MOUNT/KEEP DATE: 94098/16:22
 LAST EJECT DATE ....: 94103/17:40
 INSERT DATE ........: 94105/06:14
 SWAP COUNT OF TAPE .: 00000
 TOTAL USE COUNT ....: 00061  (DISP=OLD USECOUNT: 00000)
 ==========================================================
 CLT=0  INA=1  EJD=0  SCR=1  OD=0   IDEV=0 FPMA=0 HOME=1
 VIR=0  MVD=0  MVH=0  ALI=0  MP=0   HWS=0  INIT=0 INVC=0
 INQ=0  IUS=0  ART=0  FKES=0 CLC=0  PML=0  PMU=0  CVUP=0
 HOLD=0 SEND=0 VUA=0
 ==========================================================
```

Status byte significance - cartridges:

CLT	0	Medium is data cartridge	1	Medium is cleaning cartridge
INA	0	Ejected	1	In robot archive
EJD	0	Not significant	1	Medium should be ejected
SCR	0	Not significant	1	Volser has Scratch status
OD	0	Medium is cartridge	1	Not allowed
IDEV	0	Not significant	1	Medium located in insert area
FPMA	0	Not significant	1	Medium located in FPMA; Temporary coordinate is present
HOME	0	Volser without home coordinate	1	Volser has a home coordinate
VIR	0	Not significant	1	Volser is a virtual volume
MVD	0	Not significant	1	**MOVE** to be executed Target coordinate set
MVH	0	Not significant	1	**MOVE** to home coordinate to be executed
ALI	0	Not significant	1	Not allowed
MP	0	Not significant	1	Medium in manual pool
HWS	0	Not significant	1	Medium in HWS
INIT	0	Not significant	1	Initial status (unassigned volume)
INVC	0	Not significant	1	Invalid coordinate
INQ	0	Not significant	1	Volser in robot wait queue
IUS	0	Not significant	1	Volser in use
ART	0	Not significant	1	Volser allocated for robot test
FKES	0	Not significant	1	Not allowed
CLC	0	Not significant	1	Cleaning cartridge in use
PML	0	Not significant	1	Medium to be loaded in FPMA
PMU	0	Not significant	1	Medium to be removed from FPMA
CVUP	0	Not significant	1	Home coordinate to be changed
HOLD	0	Not significant	1	Request for volser in error wait queue
SEND	0	Not significant	1	Request for volser sent to AML
VUA	0	Volume available	1	Volume not available

Example 2: Display optical disk
- Volser of an A side
- Status=OB - at home position

```
DV OD020A
 HAC000I
=========================================================
 HAC036I VOLSER=OD020A,HOME=01020209,TEMP=       ,RECV=00000000
 HAC036I  SYS=1,ROB=1              LAST-UNIT=D08          LAST-
JOB=*HACC0*
 HAC036I STATUS=OB,HOME-POS
 HAC036I OTHER SIDE VOLSER=OD020B    STORAGE GROUP=STG001
 HAC036I SWAP COUNT OF TAPE .: 00000
 HAC036I TOTAL USE COUNT ....: 00000
 HAC000I
=========================================================
 HAC036I MOU=0  INA=1  EJD=0  SCR=0  OD=1  IDEV=0 JB=0  HOME=1
 HAC036I  VIR=0   MVD=0   MVH=0   ALI=0   MP=0   HWS=0   INIT=0
INVC=0
 HAC036I INQ=0  IUS=0  ART=0  FKES=0 CLC=0 PML=0  PMU=0
 HAC036I HOLD=0 SEND=0 VUA=0
 HAC000I
=========================================================
```

Example 3: Display optical disk
- Volser of an A side
- Status=OE - ejected

```
DV OD001A
 HAC000I
=========================================================
 HAC036I VOLSER=OD001A,HOME=01020101,TEMP=      ,RECV=00000000
 HAC036I  SYS=1  ROB=1              LAST-UNIT=D08          LAST-
JOB=*HACC0*
 HAC036I STATUS=OE,HOME-POS                    SIDE=A
 HAC036I OTHER SIDE VOLSER=OD001B    STORAGE GROUP=STG001
 HAC036I LAST EJECT DATE ....: 94223/20:25
 HAC036I SWAP COUNT OF TAPE .: 00000
 HAC036I  TOTAL  USE  COUNT  ....:  00000  (DISP=OLD  USECOUNT:
00000)
 HAC000I
=========================================================
 HAC036I MOU=0  INA=0  EJD=0  SCR=0  OD=1  IDEV=0 JB=0  HOME=1
 HAC036I  VIR=0   MVD=0   MVH=0   ALI=0   MP=0   HWS=0   INIT=0
INVC=0
 HAC036I INQ=0  IUS=0  ART=0  FKES=0 CLC=0 PML=0  PMU=0
 HAC036I HOLD=0 SEND=0 VUA=0
 HAC000I =========================================================
```

Example 4: Display optical disk
- Volser of a B side
- Status=OE - ejected

```
DV OD001B
 HAC000I
=====================================================
 HAC036I VOLSER=OD001B,HOME=01020102,TEMP=      ,RECV=00000000
 HAC036I  SYS=1  ROB=1              LAST-UNIT=D08          LAST-
JOB=*HACC0*
 HAC036I STATUS=OE,HOME-POS                    SIDE=B
 HAC036I OTHER SIDE VOLSER=OD001A    STORAGE GROUP=STG001
 HAC000I
=====================================================
 HAC036I MOU=0  INA=0  EJD=0  SCR=0  OD=1  IDEV=0 JB=0  HOME=1
 HAC036I  VIR=0   MVD=0   MVH=0   ALI=1   MP=0   HWS=0   INIT=0
INVC=0
 HAC036I INQ=0  IUS=0  ART=0  FKES=0 CLC=0 PML=0  PMU=0
 HAC036I HOLD=0 SEND=0 VUA=0
 HAC000I
=====================================================
```

Example 5: Display optical disk
- Volser of an A side
- Status=OJ - in jukebox

```
DV OD003A
 HAC000I
=====================================================
 HAC036I VOLSER=OD003A,HOME=01020105,TEMP=      ,RECV=00000000
 HAC036I  SYS=1  ROB=1              LAST-UNIT=D08          LAST-
JOB=*HACC0*
 HAC036I STATUS=OJ,IN JUKEBOX                  SIDE=A
 HAC036I OTHER SIDE VOLSER=OD003B    STORAGE GROUP=STG001
 HAC036I SWAP COUNT OF TAPE .: 00000
 HAC036I  TOTAL  USE  COUNT  ....:  00000  (DISP=OLD  USECOUNT:
00000)
 HAC000I
=====================================================
 HAC036I MOU=0  INA=1  EJD=0  SCR=0  OD=1  IDEV=0 JB=1  HOME=1
 HAC036I  VIR=0   MVD=0   MVH=0   ALI=0   MP=0   HWS=0   INIT=0
INVC=0
 HAC036I INQ=0  IUS=0  ART=0  FKES=0 CLC=0 PML=0  PMU=0
 HAC036I HOLD=0 SEND=0 VUA=0
 HAC000I
=====================================================
```

Example 6: Display optical disk
- Volser of a B side
- Status=OJ - in jukebox

```
DV OD003B
 HAC000I
=====================================================
 HAC036I VOLSER=OD003B,HOME=01020106,TEMP=      ,RECV=00000000
 HAC036I  SYS=1  ROB=1              LAST-UNIT=D08          LAST-
JOB=*HACC0*
 HAC036I STATUS=OJ,IN JUKEBOX                  SIDE=B
 HAC036I OTHER SIDE VOLSER=OD003A    STORAGE GROUP=STG001
 HAC000I
=====================================================
 HAC036I MOU=0  INA=1  EJD=0  SCR=0  OD=1  IDEV=0 JB=1  HOME=1
 HAC036I  VIR=0   MVD=0   MVH=0   ALI=1   MP=0   HWS=0   INIT=0
INVC=0
 HAC036I INQ=0  IUS=0  ART=0  FKES=0 CLC=0 PML=0  PMU=0
 HAC036I HOLD=0 SEND=0 VUA=0
 HAC000I
=====================================================
```

Example 7: Display optical disk
- Volser of an A side
- Status=OM - loaded on OAD

```
DV OD001A
 HAC000I
=====================================================
 HAC036I VOLSER=OD001A,HOME=01020101,TEMP=      ,RECV=00000000
 HAC036I  SYS=1  ROB=1              LAST-UNIT=D08          LAST-
JOB=*HACC0*
 HAC036I STATUS=OM,ON UNIT ODDRV5              SIDE=A
 HAC036I OTHER SIDE VOLSER=OD001B    STORAGE GROUP=STG001
 HAC036I LAST MOUNT/KEEP DATE: 94265/11:50
 HAC036I INSERT DATE ........: 94257/17:05
 HAC036I SWAP COUNT OF TAPE .: 00000
 HAC036I  TOTAL  USE  COUNT  ....:  00010  (DISP=OLD  USECOUNT:
00010)
 HAC000I
=====================================================
 HAC036I MOU=1  INA=1  EJD=0  SCR=0  OD=1  IDEV=0 JB=0  HOME=1
 HAC036I  VIR=0   MVD=0   MVH=0   ALI=0   MP=0   HWS=0   INIT=0
INVC=0
 HAC036I INQ=1  IUS=1  ART=0  FKES=0 CLC=0 PML=0  PMU=0
 HAC036I HOLD=0 SEND=0 VUA=0
 HAC000I
=====================================================
```

Status byte significance - optical disk:

MOU	0	Not significant	1	Medium is loaded on OAD
INA	0	Medium is ejected	1	In robot archive
EJD	0	Not significant	1	Medium should be ejected
SCR	0	Not significant	1	Volser has Scratch status
OD	0	Not allowed	1	Medium is optical disk
IDEV	0	Not significant	1	Medium located in insert area
JB	0	Not significant	1	Medium located in jukebox
HOME	0	Not allowed	1	Volser has a home coordinate
VIR	0	Not significant	1	Not allowed
MVD	0	Not significant	1	**MOVE** to be executed Target coordinate is set
MVH	0	Not significant	1	**MOVE** to home coordinate to be executed
ALI	0	A side of an optical disk	1	B side of an optical disk
MP	0	Not significant	1	Not allowed
HWS	0	Not significant	1	Not allowed
INIT	0	Not significant	1	Initial status (unassigned volume)
INVC	0	Not significant	1	Invalid coordinate
INQ	0	Not significant	1	Volser in robot wait queue
IUS	0	Not significant	1	Volser in use
ART	0	Not significant	1	Not allowed
FKES	0	Not significant	1	Not allowed
CLC	0	Not significant	1	Not allowed
PML	0	Not significant	1	Not allowed
PMU	0	Not significant	1	Not allowed
CVUP	0	Not significant	1	Not allowed
HOLD	0	Not significant	1	Request for volser in error wait queue
SEND	0	Not significant	1	Request for volser sent to AML
VUA	0	Volume available	1	Volume not available

4.26 EJ - EJECT MEDIUM (CHANGED)

```
EJ volser[,Enn][,MANPOOL/NOMANPOOL][WAIT]
```

volser	= VOLSER number	
Enn	= Eject unit	Default: E01 or E00[1)]
WAIT	= Wait option (refer to SYS.REF.GUIDE)	
MANPOOL	= Eject to a manual pool	Default: NOMANPOOL

EJ initiates medium eject.

The Archive mirror is checked to see whether the medium is actually in the Archive, and then marked in the Archive mirror as "planned for ejection". This ensures that this information is not lost should the HCC task abort.

The **EJECT** statement is then entered in the wait queue for execution. The command is then passed on when no further tasks are pending or when the cartridge is no longer in use.

When the "MANPOOL" option is specified, the xRXMP flag in the Archive mirror is set after a successful eject. This flag can be checked by products used to control allocation (for example, MAPS from COMPAREX).

When this command is entered before a **KEEP**, the **KEEP** is made directly to the eject slot according to the following rules:
For **EJ**, **EJDSN** or **EJVOL** with eject area (for example, E01)
- when the robot is the same, in the eject area specified,
- when the robot is different, back to the tower.

The **EJECT** is then generated as an independent request.
For **EJ**, **EJDSN** or **EJVOL** without eject area, in the default eject area of the respective robot.

1)

When an eject unit is not specified for twin-robot systems, the eject slot to be used is first determined, for performance reasons, when the eject command is actually generated. The following rules are applied in this case:
- towers 01/03 ... are assigned to robot 01
- towers 02/04 ... are assigned to robot 02

whilst considering whether one of the robots is active. The default value E01 is applicable for all other systems. In the case of optical disks, the first eject unit valid for optical disks is selected.

Examples:

```
EJ 014567
EJ 004711,E11,MANPOOL
```

The following message confirms successful ejection:

```
HAC124I VOLUME 014567 IS EJECTED
HAC124I VOLUME 004711 IST EJECTED-MANPOOL
```

Refer to the System Reference Guide, Appendix, Section BATCH examples

4.27 FLIP - TURNOVER OPTICAL DISK (NEW)

```
FLIP UNIT=drv
```

drv = Drive name of the Operator Accessible Drive (OAD)

FLIP tasks the AML system to turn the optical disk (OD) on the specified unit (OAD) over.

The OD is turned over without the AMU checking the volser.

Example:

```
FLIP UNIT=ODDRV5
```

This command is only provided for exceptional circumstances and can cause processing errors when used incorrectly.

4.28 FMA - ADD FOREIGN MOUNT CARTRIDGE ENTRY (CHANGED)

```
FMA [sd,]nn,volser[,jobname][,G,I]
```

sd	= System *s* designator and the foreign mount unit d= last character of the foreign mount unit F0d
nn	= Position number in the foreign mount area (*11001 - *sdnnn)
volser	= Actual VOLSER of the MVS mount (6 characters)
jobname	= Job name that requests real VOLSERs
G	= Generic job name (1-8 characters per job name)
I	= Deactivate the entry after cartridge processing

FMA assigns the "volser,jobname" combination to the foreign mount slot position. The operator should set the corresponding "volser" to any free position "nn" in the foreign mount area before the selected "job" starts.

As soon as the "volser,jobname" combination matches during a **MOUNT** process, AML loads the cartridge positioned in slot "nn" without checking the barcode label and returns it to the same position after KEEP.

The HAC182I message appears when the "VOLSER,JOBNAME" combination is already defined for another foreign mount slot position.

This message also appears when the foreign mount slot position is already in use.

Example:

```
FMA 12,01,004711,UZ4333
```

Message:

```
HAC041I REQUEST OK
```

This command is only allowed for cartridges and incorrect use can cause processing errors.

Another possibility is to wait for the message

```
HAC332A VOLSER volser FOR JOB jobname NOT FOUND IN HACC ARCHIVE
```

and then position the respective cartridge in a free slot. The following **FMA** command then continues the foreign mount process from the external slot.

The same procedure can be used when an error occurs during definition of the external slot (nn, volser or jobname). The foreign mount process continues after the erroneous data have been corrected.

4.29 FMD - DELETE FOREIGN MOUNT ENTRY (CHANGED)

```
FMD [sd,]nn
```

sd = System designator and the foreign mount unit
d= last character of the foreign mount unit F0d
nn = Position number in the foreign mount area (*11001 - *sdnnn)

FMD deletes the entry for the slot nn generated by FMA.

Example:

```
FMD 12
```

Message:

```
HAC036I FM-ENTRY *11002,004711,Y6011A   DELETED
```

This command is only valid for cartridges and incorrect use can cause processing errors.

4.30 FREE - RELEASE TAPE UNITS (CHANGED)

```
FREE cuu[,FORCE]
```

cuu = MVS unit address
FORCE = Execute independent of pending requests

FREE releases the tape unit reserved with the **ALLOC** command and deletes the corresponding subtask.

FORCE can be necessary when, for example, the subtask demands a tape during a label check (refer to the LCK function) and the tape is not in the Archive. The pending **OPEN** command can only be overcome by using **FORCE** for the unconditional removal of the subtask in question.

This command is only valid for tape units and incorrect use can cause processing errors.

Example:

```
FREE 0481
```

4.31 FREEVOL - DETERMINE SCRATCH DATA MEDIA (CHANGED)

```
FREEVOL   ADD [,RESET][,TYPE=CAR]
FREEVOL   REP [,RESET][,TYPE=CAR]

FREEVOL   REP            ,TYPE=OPT
```

ADD	= Add new SCRATCH information; existing information is retained. Only valid for TYPE=CAR	(Default)
REP	= Replace existing SCRATCH information Only valid for TYPE=CAR and TYPE=OPT	
RESET	= Reset to the first SCRATCH data medium available Only valid for TYPE=CAR	
CAR	= Selection of SCRATCH tapes	
OPT	= Selection of optical disks with FREESPACE	(Default)

FREEVOL serves to inform HCC on new SCRATCH data media.

Working with tapes

HCC substitutes non-specific **MOUNT** requests for tape stations (SCRTCH, PRIVAT, ...) with the next available free cartridge marked as such in the Archive mirror.

A pointer to the next scratch tape is maintained. This pointer is not normally reset by a **FREEVOL** command which supports an even load across a Scratch pool.

RESET can reset the pointer. The table is recreated during a HCC start and the pointer is then set to the first available data medium.

The following message appears when the scratch warning level is underflowed:

```
HAC058A SCRATCH TAPE WARNING LEVEL ...
```

The following message appears when no more free cartridges are available in the Archive mirror:

```
HAC028A WARNING: NO MORE SCRATCH TAPES AVAILABLE FOR POOL pppppp
```

Pending non-specific **MOUNT** requests for the pool involved can no longer be processed.

Pending requests continue automatically after successful execution of a **FREEVOL** command.

A "FREEVOL REP" is automatically executed when the initialization parameter AUTOFREEVOL=YES is set and the scratch warning level is then reached and the last available scratch cartridge has been used.

Another method of automating this process is to issue the **FREEVOL** command through **ABBASEND** during the daily job run by the tape management system.

Working with optical disks

Non-specific requests for optical disks are serviced with the next available optical disk matching the requested Storage group.

Example 1:

Replace scratch tape information

```
FREEVOL REP
```

Default(s): - TYPE=CAR

```
HAC029I SCRATCH-TAPE ARCHIVE UPDATE IN PROGRESS
HAC148I NO.OF PASSED SCRATCH-TAPES BY TMS-EXIT = 000144
HAC149I NO.OF EJECTED SCRATCH-TAPES            = 000011
HAC150I NO.OF CLEAN-CARTRIDGES                 = 000000
HAC151I NO.OF UNKNOWN CARTRIDGES               = 000000
HAC030I TOTAL NO. OF AVAILABLE SCRATCH TAPES   = 000144
HAC249I ARCHIVE UPDATE FOR SCRATCH TAPES STARTED
HAC250I  ARCHIVE  UPDATE  FOR  SCRATCH  TAPES  SUCCESSFULLY
COMPLETED
HAC215I SCRATCH-GROUPS:
HAC192I  NOSCR=000133 001001-010000 (SCRTCH)
HAC192I  NOSCR=000133 001001-010000 (PRIVAT)
HAC040I SCRATCH-TAPE UPDATE COMPLETED
```

Example 2:

Replace free space information for optical disks

```
FREEVOL REP,TAPE=OPT
```

```
HAC230I FREEVOL FOR OPT IN PROGRESS
HAC234I FREEVOL FOR OPT FOUND TOTAL VOLSERS   : 00108
HAC234I FREEVOL FOR OPT FOUND A-VOLSERS       : 00102
HAC234I FREEVOL FOR OPT FOUND B-VOLSERS       : 00001
HAC234I FREEVOL FOR OPT FOUND UNKNOWN VOLSERS : 00005
HAC234I FREEVOL FOR OPT FOUND NON-OD VOLSERS  : 00000
HAC234I FREEVOL FOR OPT FOUND EJECTED VOLSERS : 00000
HAC234I FREEVOL FOR OPT FOUND IN JB VOLSERS   : 00000
HAC249I ARCHIVE UPDATE FOR SCRATCH OPTS. STARTED
HAC284I FREEVOL PROCESSING FOR OPT COMPLETED SUCCESSFULLY
HAC250I   ARCHIVE   UPDATE   FOR   SCRATCH   OPTS.   SUCCESSFULLY
COMPLETED
```

4.32 HELP - DISPLAY COMMAND OVERVIEW (CHANGED)

```
HELP  ALL
HELP generic cmd
```

ALL = Displays all operator commands
generic cmd = HCC command as 1 - 8 byte generic character string

HELP displays all available operator commands.

Example:

```
HELP D
```

Display all HCC commands starting with 'D':

```
HAC000I ==============================================
HAC036I DA             DISPLAY ACTIVE FUNCTIONS
HAC036I DU             DISPLAY UNITS
HAC036I DRQ            DISPLAY SEND QUEUE
HAC036I DSC (NNN)      DISPLAY SCRATCH VOLSERS
HAC036I DRE (N***)     DISPLAY ROBOT ERROR TEXT
HAC036I DHR            DISPLAY HACC-RELEASE
HAC036I DRT            DISPLAY ROBTEST/LABELCHK/-INIT
HAC036I DCSA           DISPLAY CSA-INFORMATIONS
HAC036I  ...           cont.
HAC036I ==============================================
```

4.33 INC - INVENTORY USING COORDINATES (CHANGED)

```
INC [s,r,] coor1-coor2[,UPD][,SKIP=NO/YES]
INC [s,r,] COOR=coor..[,UPD][,SKIP=NO/YES]
INC [s,r,] STOP
```

s	= AML System	Default: s = 1
r	= Robot No.	Default: r = 1
UPD	= Update the HCC Archive with recognized status and VOLSER	
COOR=coor	= Generic coordinate 1 - 8 bytes	
coor1-coor2	= Coordinate range	
SKIP	= Coordinate increment	
YES	= Process every 2nd coordinate For cartridges, process only every 2nd medium. For optical disks, generate 1 command for each medium.	
NO	= Process all coordinates	(Default) For cartridges, generate 1 command for each medium. For optical disks, generate 2 commands for each medium.
STOP	= Stop download	Default: end of specified range

INC triggers an inventory for the specified coordinate range.

The robot reads the barcode labels of all cartridges in slots and checks empty slots. The inventory results are compared with the AML Archive mirror and the HCC Archive mirror and the following messages appear when differences are detected:

```
HAC295I SYS s,ROB r: INVENTORY DIFFERENCES AT:
HAC296I ABBA: VOLSER=volser,STATUS=ss,COORD=coord
HAC297I HACC: VOLSER=volser,STATUS=ss,COORD=coord
```

It is recommended to start the HCC Message log to record possible inventory errors (refer to LOGSTART.)

Example:

```
INC 2,1,01010101-01011510
```

Only 1 **INC** command may be active for a robot at any one time. Several simultaneous **INC** commands for one robot can lead to unexpected actions.

The UPD parameter restricts the changes to the HCC Archive mirror (the corresponding AML system is **not** updated). A **DL** command must be executed afterwards to update the AML Archive mirror (update the AMU or AR Archive).

4.34 INV - INVENTORY USING VOLSER (CHANGED)

```
INV [s,]volser[,UPD]
```

s	=	AML System	Default: s = 1
UPD	=	Update the HCC Archive with recognized status and VOLSER	
volser	=	Library VOLSER	

INV triggers an inventory for the specified volsers.

The robot reads the barcode labels of all cartridges in slots and checks empty slots. The inventory results are compared with the AML Archive mirror and the HCC Archive mirror and the following messages appear when differences are detected:

```
HAC295I SYS s,ROB r: INVENTORY DIFFERENCES AT:
HAC296I ABBA: VOLSER=volser,STATUS=ss,COORD=coord
HAC297I HACC: VOLSER=volser,STATUS=ss,COORD=coord
```

It is recommended to start the HCC Message log to record possible inventory errors (refer to LOGSTART.)

Examples:

```
INV 004711
INV 2,001001,UPD
```

4.35 K - INDIVIDUAL KEEP (CHANGED)

```
K cuu[,volser]
K drv[,volser]
```

cuu	= MVS unit address (3 or 4 digits)
drv	= Drive name of the OAD[1]
volser	= VOLSER of the AML library (including the foreign mount area)

When the **K** command is entered, the HOST tasks the AML system to remove the specified cartridge from the desired unit (**KEEP**).

This command is only provided for exceptional circumstances and can cause processing errors when used incorrectly.

[1] **O**perator **A**ccessible **D**rive

KEEP command example:

Example 1:

```
K 0482,004711
```

KEEP from a tape unit.

Example 2:

```
K 0483
```

KEEP from a tape unit without specifying the volser number.

Example 3:

```
K 0481,*11003
```

KEEP from a tape unit to move a cartridge from unit 0481 to the foreign mount position 003.

Example 4:

```
K ODDRV5
```

KEEP from an OAD[1] without specifying a volser number.

[1] Operator Accesible Drive

4.36 LCK - DEFINE LABEL CHECK (CHANGED)

```
LCK fromvol-tovol
```

fromvol = Start VOLSER
tovol = End VOLSER

LCK selects a cartridge number range for an automatic header check.

HCC includes an internal header check function as a service function. This function is useful when it is suspected that the barcode label of a tape no longer matches the VOL1 label.

This command is only allowed for cartridges and incorrect use can cause processing errors.

Example:

```
LCK 002506-002511
```

4.37 LJB - LOAD JUKEBOX (NEW)

```
LJB NAME=jbname,VOL=volser
```

jbname = Jukebox name
volser = Volser of the side facing upwards

LJB serves to load the jukebox, for example, ODs are moved from the home coordinate to I/O-S.

The AMU does not maintain an Archive on the jukebox. The home coordinate is switched to status "In jukebox" after execution.

An optical disk (OD) moved to the jukebox is marked in the HCC Archive with status "OJ" (OD in jukebox).

An optical disk has the logical status "O".

Example:

```
LJB NAME=ODLIB1,VOL=04711A
```

4.38 M - INDIVIDUAL MOUNT (CHANGED)

```
M cuu,volser
M drv,volser
```

cuu	= MVS unit address (3 or 4 digits)
drv	= Drive name of the Operator Accessible Drives (OAD)
volser	= VOLSER of the AML library (including foreign mount area and pool names)

The **M** command tasks the AML system to load the specified medium on the desired unit.

This command is only provided for exceptional circumstances and can cause processing errors when used incorrectly.

MOUNT command examples:

Example 1:

```
M 0482,004711
```

Individual cartridge **MOUNT** on a tape unit.

Example 2:

```
M 0483,SCRTCH
```

MOUNT the next available SCRATCH tape from the SCRTCH pool. Refer to the **FREEVOL** command and VOLGR statement (INSTALLATION/CUSTOMIZATION Manual for HCC/MVS 3.0.0).

Example 3:

```
M 0481,*11003
```

Individual **MOUNT** from foreign mount slot 003.

Example 4:

```
M ODDRV5,OD001A
```

Individual **MOUNT** of an OD onto to an OAD[1].

[1] Operator Accesible Drive

4.39 MOVEH - MOVE TO HOME COORDINATE (CHANGED)

```
MOVEH [s,] coor1,coor2
MOVEH [s,] vols1,coor2
```

s	= AML System	Default: s = 1
coor1	= Home coordinate	
vols1	= VOLSER of the home coordinate	
coor2	= Target coordinate	

MOVEH generates a **MOVE** command to the home.

MOVE moves media from any position to another.

Two cases must be considered: movement to a new HOME coordinate (**MOVEH**) and movement to a TEMPORARY coordinate (**MOVET**). The HOME coordinate is the fixed Archive slot for the cartridge as defined during installation and, for example, automatically searched for during insert.

In hierarchical archives, movement to a new HOME coordinate is a step towards random archives, and therefore a MOVE to a new HOME position should only be used to clear an error situation.

Example:

```
MOVEH 01021015,02010101
MOVEH 2,004711,04010102
```

4.40 MOVET - MOVE TO TEMPORARY COORDINATE (CHANGED)

```
MOVET [s,] coor1,coor2
MOVET [s,] vols1,coor2
```

s	= AML System	Default: s = 1
coor1	= Home coordinate	
vols1	= VOLSER of the home coordinate	
coor2	= Target coordinate	

MOVET generates a **MOVE** command to a temporary coordinate.

MOVE moves media from any position to another.

Two cases must be considered: movement to a new HOME coordinate (**MOVEH**) and movement to a TEMPORARY coordinate (**MOVET**). The HOME coordinate is the fixed Archive slot for the cartridge as defined during installation and, for example, automatically searched for during insert.

In hierarchical archives, movement to a new HOME coordinate is a step towards random archives, and therefore a MOVE to a new HOME position should only be used to clear an error situation

Moving a cartridge to a TEMPORARY address is done for performance reasons. When FPMA is installed on the system, HCC uses temporary **MOVE**s to or from this intermediate storage without losing the HOME position. The user can also use the **MOVET** command but, apart from the possibility for manual corrections, this has no further significance for the installation.

This command is not allowed for optical disks and incorrect use can cause processing errors.

Example:

```
MOVET 01021014,02010101
MOVET 2,004711,04010102
```

4.41 MRO - ALLOWING MOUNTS (DELETED)

This HCC/MVS 2.4.0 operator command was deleted in HCC/MVS 3.0.0.

4.42 PCL - STOP AUTOM. TAPE CLEANING (CHANGED)

```
PCL [s][,r]
```

s	= AML System No.	Default = 1
r	= Robot No.	Default = 1

PCL inhibits all cleaning processes for the specified system and robot.

This command is only allowed for tape units and incorrect use can cause processing errors.

Example:

```
PCL 2,1
```

4.43 PLCK - STOP LABEL CHECK (CHANGED)

```
PLCK cuu
```

cuu = MVS unit address (3 or 4 digits) reserved with an **ALLOC** command

PLCK stops an automatic header check prematurely.

This command is only allowed for tape units and incorrect use can cause processing errors.

Example:

```
ALLOC 0483
LCK 000001-001000
SLCK 0483
PLCK 0483
```

4.44 PRT - STOP ROBOT TEST (DELETED)

This HCC/MVS 2.4.0 operator command was deleted in HCC/MVS 3.0.0.

4.45 PTLI - STOP TAPE LABEL INITIALIZATION (CHANGED)

```
PTLI cuu
```

cuu = MVS unit address (3 or 4 digits) reserved with an **ALLOC** command

PLTI stops an active tape label initialization prematurely.

This command is only allowed for tape units (3480/3490) and incorrect use can cause processing errors.

Example:

```
PTLI 04B3
```

4.46 SCL - START AUTOM. TAPE UNIT CLEANING (CHANGED)

```
SCL [s][,r]
```

s	= AML System No.	Default = 1
r	= Robot No.	Default = 1

SCL releases all tape unit cleaning processes for the specified system and robot.

This command is only allowed for tape units (3480/3490) and incorrect use can cause processing errors.

Example:

```
SCL 2,1
```

4.47 SET PRTY SET BASIC PRIORITY (CHANGED)

		De
`SET P[RTY] ,MO=nn[,ALL]`	**MOUNT command**	**35**
`SET P[RTY] ,KE=nn[,ALL]`	**KEEP command**	**30**
`SET P[RTY] ,VI=nn[,ALL]`	**VOLUME INSERT command**	**20**
`SET P[RTY] ,EJ=nn[,ALL]`	**EJECT command**	**20**
`SET P[RTY] ,MV=nn[,ALL]`	**MOVE command**	**10**
`SET P[RTY] ,SCH=nn`	**SEARCH command**	
`SET P[RTY] ,sqnr=nn`	**Priority for existing sequence number**	
`SET P[RTY] ,JOB=jobname,INCR=ii` `SET P[RTY] ,J=jobname,I=ii`	**Relative priority for MO/KE**	
`SET P[RTY] ,JOB=jobname,INCR=ii[,G]` `SET P[RTY] ,J=jobname,I=ii[,G]`	**Relative priority for MO/KE for generic job names**	
`SET P[RTY] ,JOB=jobname,DEL` `SET P[RTY] ,J=jobname,D`	**Delete job name from table**	
`SET P[RTY] ,JOB=jobname,DEL[,G]` `,J=jobname,D[,G]`	**Delete generic job name from table**	
`SET P[RTY] ,LIST` `SET P[RTY] ,L`	**Display priorities**	

nn	= New priority value	(00-80)
ALL	= Set all pending requests to the new priority	
sqnr	= HCC sequence number	(refer to the **DRQ** command)
jobname	= Valid job name	(1 to 8 chars.)
ii	= Change the existing priority by ii	
G	= Generic	

The De column shows the default values for the corresponding **SET PRTY** command.

SET PRTY supports individual priority control of the AML system.

Normally, **MOUNT/KEEP** tasks should have execution preference over **EJECT/VOLUME INSERT**. **MOVE** commands are generated by HCC, for example, to load/unload the FPMA.

Changing the **PRTY** values can influence the sequence to match operational requirements during the life cycle of a HCC. The priority default value nn can be defined in the HACPARM1 member and can deviate from the HCC default values.

SET P,MO=nn also changes the priority value for the **LJB**, **UJB** and **FLIP** commands.

HCC priority control:

The priority of the first pending task within the groups, starting at the defined basic priority, is incremented by a specific value during each cycle of the dispatching algorithm .
This has the effect that, for example, **EJECT** requests pending in an EJECT wait queue are not completely pushed back during high **MO**/**KE** activities in the AML system.

Priority values 81-99 are used internally by HCC, for example for priority insertion of cartridges with pending **MOUNT** request.

JOB priority modifications are lost after HCC termination. Fixed JOB priorities should therefore be defined in the HACPARM1 member (refer to the INSTALLATION/CUSTOMIZATION GUIDE, HACPARM1, PRTY statement).

The current respective priority can be displayed with the **DRQ P** or **SET P,L** commands.

Examples:

```
SET PRTY,VI=22,ALL
SET P,EJ=18
SET P,J=HSM,I=+10
SET P,J=JOBA,I=+15,G
SET P,J=TESTJOB,I=-10
SET P,J=TEST123,DEL
SET P,L
```

(**VI** before **EJ**)

(Job HSM has priority)
(Jobs JOBA* have priority)
(TESTJOB with lower priority)
(Remove job from priority list)
(Display current priorities)

Display:

```
HAC036I ==============================================
HAC036I CMD PRTY:MO=35,KE=30,EJ=18,VI=22,MV=10,SCH=25
HAC036I ==============================================
HAC036I JOBNAMES AND RELATIVE MO/KE-PRIORITIES
HAC036I W0273001  7             TESTJOB   10-
HAC036I OBS04711 25             JOB11111   5
HAC036I HSM      10             JOB00001   1
HAC036I JOB00002  2             JOB00003   3
HAC036I JOBA*    15
HAC036I ==============================================
```

4.48 SETUP - CORRECT THE UNIT LOG (CHANGED)

```
SETUP cuu,CLEAR[,ALL]
      cuu,volser,jobname
      drv,CLEAR[,ALL]
      drv,volser,jobname
```

cuu	= MVS unit address (3 or 4 digits)
drv	= Name of the OAD (Operator Accessible Drive)
ALL	= Optional operand
CLEAR	= Mandatory operand
volser	= Valid VOLSER
jobname	= Job name

SETUP serves to clear error situations. The ALL operand also deletes all pending mounts.

The interrelation of the MVS, application job, HCC, AR or AMU and robot components can lead to situations where automatic recovery is no longer possible.

The AML control retrieves information from the UNIT log which can be corrected with the **SETUP** command.

The DU command can be used to display the Unit log information.

Example:

```
SETUP 0487,CLEAR
```

A tape was manually removed from tape station 0487 after a robot malfunction. The next mount for this unit is already in the wait queue (refer to **DRQ**) and is not sent by HCC because the AR or AMU has not yet acknowledged the previous **KEEP**.

SETUP deletes the pending **KEEP** from the UNIT log and the following mount is sent.

4.49 SETUP AR - CORRECT UNIT AR/AMU (CHANGED)

```
SETUP AR,cuu/drv/libname,CLEAR
SETUP AR,cuu/drv/libname,VOLSER
```

MVS unit identification:
- cuu = MVS unit address for cartridge unit
- drv = OAD drive - name for optical disk
- libname = Library name for optical disk

CLEAR = Mandatory operand to clear unit information (AR/AMU)
VOLSER = Valid VOLSER to update unit information (AR/AMU)

SETUP serves to clear an error situation.

This command can be used to clear error N202 issued by the AR/AMU when the unit is definitely not occupied, for example, when a medium was manually removed from the unit. The message acknowledged negatively with N202 should then be sent to the AR/AMU again with the **REP** command.

Examples:

```
SETUP AR,0487,CLEAR
```

```
SETUP AR,ODDRV5,OD001B
```

```
SETUP AR,ODLIB1,CLEAR
```

AR/AMU = Archive computer (ABBA/1) or Archive Management Unit (AML/2, AML/E, AML/J)

4.50 SLCK - START LABEL CHECK (CHANGED)

```
SLCK cuu
```

cuu = An MVS unit address (3 or 4 digits) reserved with **ALLOC** command.

SLCK starts an automatic header check for cartridges.

At least on tape unit must first be reserved with an **ALLOC** command. After the number range has been determined with **LCK**, **SLCK** can be used to start the automatic header check and stopped prematurely with **PLCK** when necessary.

Up to 4 label checks can run parallel.

The following message appears when differences are detected between the barcode label and the VOL1 header:

```
HAC163A LABEL DISCREPANCIES IN BARCODE/HEADER=volser/volser
```

It is recommended to start the HCC Message log to record errors.

The **LCK**, **PLCK** and **ALLOC** commands are related to the **SLCK** command.

This command is only allowed for tape units and incorrect use can cause processing errors.

Example:

```
SLCK 04B0
```

4.51 SRT - START ROBOT TEST (DELETED)

This HCC/MVS 2.4.0 operator command was deleted in HCC/MVS 3.0.0.

4.52 START LABEL START OPTICAL DISK LABEL PROCESS (NEW)

```
S[TART] LABEL,VOL=volser,PARM=p
```

volser = A-volser of the medium to be labeled
p = Label type to be started. The following values are possible:

W	for F OAM,LABEL, **3995WORM**	(Default)
R	for F OAM,LABEL, **3995REWR**	
A	for F OAM,LABEL, **3995-133**	

START starts a labeling process for an optical disk on an IBM 3995-13x. An F **OAM,LABEL,labtyp** command is issued.

Preconditions:

- the A side of the medium **must** be entered as volser,
- the volser **must** be assigned to a Storage group,
- automatic message handling for OAM messages must be active.

This command is only allowed for optical disks (OD) and incorrect use can cause processing errors.

Example:

```
START LABEL VOL=OD002A,PARM=A
```

Refer also to the **CANCEL LABEL** and **DISPLAY LABEL** commands

4.53 STLI - START TAPE LABEL INITIALIZATION (CHANGED)

```
STLI cuu
```

cuu = MVS unit address (3 or 4 digits) reserved with an **ALLOC** command

STLI starts an automatic tape initialization.

At least one tape unit must first be reserved with an **ALLOC** command. After the number range has been determined with **TLI**, **STLI** can be used to start the automatic initialization and stopped prematurely with **PTLI** when necessary.

A password is sometime necessary for the **STLI** command (depending on the HACPARM parameter TLIPW).

It must be ensured that the MVS function MONITOR DSNAME is active before issuing the **SRT** command. This is because the full DSNAME only appears in the MOUNT message when DSNAME monitoring is active, otherwise only the MOUNT message appears:

An **STLI** command can be issued for each assigned tape station, and up to 4 tape label initialization function can run parallel, however only for 1 range (TLI fromvol-tovol).

This command is only allowed for tape units (3480/3490) and incorrect use can cause processing errors.

Example:

```
ALLOC 0483
TLI 001000-001009
MN DSNAME
TLIPW OBIS
STLI 0483
```

4.54 TLI - DEFINE TAPE LABEL INIT (CHANGED)

```
TLI fromvol-tovol
```

fromvol = Start VOLSER
tovol = End VOLSER

HCC includes a service function for tape header initialization.
The standard tape header records VOL1, HDR1, HDR2, TAPE MARK are created.

TLI serves to select a range of cartridges for automatic initialization.

Example:

```
TLI 000001-001000
```

This command is only allowed for tape units (3480/3490) and incorrect use can cause processing errors.

Refer also to **ALLOC**, **STLI**, **PTLI**, **TLIPW**.

The range that can be specified with **TLI** must be unique within a HCC complex.

Multiple inputs overwrite previous ranges.

4.55 UJB - UNLOAD JUKEBOX (NEW)

```
UJB NAME=jbname,VOL=volser
```

jbname = Jukebox name up to 8 characters
volser = Volser of the side facing upwards

UJB serves to unload the jukebox, for example, ODs are removed from the I/O-S[1] and returned to their home coordinate.

The volser sent identifies whether the A or B side of the OD is face upwards I/O-S.

Vision-System cannot read the barcode of the disk during removal because the OD is removed horizontally.

Example:

```
UJB NAME=ODLIB1,VOL=04711A
```

[1] Input-Output Station

4.56 ULC - UPLOAD ARCHIVE USING COORDINATES (CHANGED)

```
ULC [s,] coor1-coor2 [,SKIP=NO/YES]
ULC [s,] COOR=coor.. [,SKIP=NO/YES]
ULC [s,] STOP
```

s	= AML System	Default: s = 1
COOR=coor	= Generic coordinate 1 - 8 bytes	
coor1-coor2	= Coordinate range	
SKIP	= Coordinate increment	
YES	= Process every 2nd coordinate For cartridges, process only every 2nd medium. For optical disks, generate 1 command for each medium.	
NO	= Process all coordinates For cartridges, generate 1 command for each medium. For optical disks, generate 2 commands for each medium.	(Default)
STOP	= Stop download	Default: end of specified range

ULC transfers data such as volser, status and coordinates (but **not** the AML system in question) of the AR or AMU Archive mirror to the HCC Archive mirror.

The run time must be considered when transferring the complete Archive or large sections of the Archive.

Example:

```
ULC COOR=03
```

This transfers all the coordinates of tower 03 to the HCC Archive mirror.

```
HAC000I ==============================================
HAC036I COMMAND=ULC SQNR=2506
HAC036I ===== VOLSER SYS HOMECOOR TEMPCOOR STATUS
HAC036I ABBA: FREI00   1   03010101            0,L,-
HAC036I HACC: 001001  1  03010101         M,B,N
HAC036I DIFF: ******                      * *
HAC000I ==============================================
```

Volume status:

Position 1:	0	= Special status (AR/AMU)
	C	= Cleaning cartridge
	M	= Data cartridge
	O	= Optical disk
Position 2:	B	= In Archive
	E	= Ejected
	J	= Optical disk in the jukebox (phys. status = O)
	L	= Initial (phys. status = 0)
	M	= Loaded
	O	= ‘Other side’ of the optical disk loaded (phys status = O)
Position 3:	N	= Marked as no-scratch (cataloged)
	S	= Marked as scratch
Position 4:	I	= Volume in send wait queue
Position 5:	D	= Volume in use (for example, loaded on unit)

4.57 ULV - UPLOAD ARCHIVE USING VOLSER (CHANGED)

```
ULV [s,] fromvol-tovol
ULV [s,] volser
ULV [s,] STOP
```

s	= AML System	Default: s = 1
volser	= VOLSER 6 bytes	
fromvol-tovol	= VOLSER range	
STOP	= Stop upload	Default: end of specified range

ULV transfers data such as volser, status and coordinates (but **not** the AML system in question) of the AR or AMU Archive mirror to the HCC Archive mirror..

The run time must be considered when transferring the complete Archive or large sections of the Archive.

Example:

```
ULV 010001-011000
```

```
HAC000I ==============================================
HAC036I COMMAND=ULV SQNR=2606
HAC036I ===== VOLSER SYS HOMECOOR TEMPCOOR STATUS
HAC036I ABBA: 001001  1  01010101          M,E,-
HAC036I HACC: 001001  1  01010101          M,B,N
HAC036I DIFF:                                *
HAC000I ==============================================
```

Volume status:

Position 1:	0	= Special status (AR/AMU)
	C	= Cleaning cartridge
	M	= Data cartridge
	O	= Optical disk
Position 2:	B	= In Archive
	E	= Ejected
	J	= Optical disk in the jukebox (phys. status = O)
	L	= Initial (phys. status = 0)
	M	= Loaded
	O	= ‘Other side’ of the optical disk loaded (phys status = O)
Position 3:	N	= Marked as no-scratch (cataloged)
	S	= Marked as scratch
Position 4:	I	= Volume in send wait queue
Position 5:	D	= Volume in use (for example, loaded on unit)

4.58 UNLOAD - UNLOAD UNIT (CHANGED)

```
U[NLOAD] cuu
```

cuu = MVS unit address of the 3480/3490 unit (3 or 4 digits)

UNLOAD- serves to unload the cartridge unit (3480/3490). An MVS Unload command is generated (U cuu).

This command is only allowed for tape units (3480/3490) and incorrect use can cause processing errors.

Example:

```
U 0441
```

4.59 UPC - UPDATE ARCHIVE USING COORDINATES (CHANGED)

```
UPC [s,]coord,lp
```

s = AML System — Default: s=1
coord = 8 byte coordinate

l = Logical status
- M= Normal tape
- C = Cleaning tape
- O = Optical disk

p = Physical status
- B = Tape in library
- E = Tape ejected
- M = Medium loaded (on OAD for optical disk)
- O = Other side loaded (only when 1st position = ‘O’)
- J = Optical disk in jukebox

UPC serves to correct the volser status; both Archive mirrors (HCC and AML) are corrected.

UPC can be used to adjust the Archive mirror when inventory errors are reported, as well as to eject cartridges in manual operation.

Example:

```
UPC 01020102,ME
```

4.60 UPV - UPDATE ARCHIVE USING VOLSER (CHANGED)

```
UPV volser,lp [NAME=lib]
```

volser = VOLSER from the library

l = Logical status
- M= Normal tape
- C = Cleaning tape
- O = Optical disk (OD)

p = Physical status
- B = Tape in library
- E = Tape ejected
- M = Medium loaded (on OAD for optical disk)
- O = Other side loaded (only when 1st position = ‘O’)
- J = Optical disk in jukebox

NAME = IOS name, only required for status OJ

UPV serves to correct the volser status; both Archive mirrors (HCC and AML) are corrected.

UPV can be used to adjust the Archive mirror when inventory errors are reported, as well as to eject cartridges in manual operation.

Example:

```
UPV 000001,CE
```

Cleaning cartridge 000001 is marked as ejected.

4.61 VARY - CONTROL COMMUNICATION (CHANGED)

```
VARY   ACT,HID=Hx|Ax|Sx[,TYPE=PRI|ALT][,PATH=APPC|DASD]
VARY IN  ACT,HID=Hx|Ax|Sx[,TYPE=PRI|ALT][,PATH=APPC|DASD],I
```

ACT	= Activate connections
INACT	= Deactivate connections
HID=Hx	= HACC0 ID of the partner primary HCC (currently only when PATH=APPC) x = 1-9 and A-Z
HID=Ax	= AMU ID of the partner AMU (currently only when PATH=APPC) x = 1-9 and A-Z
HID=Sx	= Partner secondary HCC (only when PATH=DASD) x = 1-9 and A-F
PRI	= Connection to primary AR/AMU (default)
ALT	= Connection to alternative AR/AMU (only when PATH=APPC and Ax)
APPC	= Start/terminate a LU 6.2 connection (default)
DASD	= Start/terminate an EXCLOG connection
I	= The APPC main task (HAC23054) is terminated immediately

VARY ACT, HID=Hx/Ax establishes a single conversation to Hx/Ax, and VARY INACT, HID=Hx/Ax terminates this conversation again.

VARY INACT, APPC terminates all active conversations as well as the APPC subtask. VARY ACT, APPC starts the APPC subtask and attempts to establish conversations with all possible partners.

VARY INACT,HID=Hx/Ax,I immediately terminates the conversation pair and the corresponding subtasks in the following manner:

Terminate the RECEIVE conversation:
`APPCCMD CONTROL=REJECT,QUALIFY=CONV,...`

Terminate the SEND conversation:
`APPCCMD CONTROL=DEALLOC,QUALIFY=FLUSH,...`

Terminate the SUBTASKs with DETACH.

Example:

```
VARY ACT,PATH=APPC
```

4.62 VI - START INSERTION (CHANGED)

```
VI
VI Inn
VI Inn,s
```

Inn	= Insertion area nn	Default: nn=01
s	= AML System	Default: s=1

VI starts insers. The data media can be positioned to the corresponding insert racks Inn in any sequence (random).

VI tasks AML to identify the medium with the Vision system / barcode reader. Data media with home coordinates are returned to this position.

HCC assigns a home position in the dynamic area to virtual cartridges without home positions.

Optical disks (OD) cannot be managed in the dynamic area.

Each insert is acknowledged with the message:

```
HAC144I CARTRIDGE volser INSERTED

or

HAC144I OPTICAL DISK volser A INSERTED
```

Media with illegible barcodes (BLANK-VOLSER) are moved to the Problem box.

Example:

```
VI
HAC399I INSERT DEVICE I01 SYSTEM 1 ACTIVE
```

Start insert; default unit I01, system 1

```
VI I93,2
HAC399I INSERT DEVICE I93 SYSTEM 2 ACTIVE
```

Start insert; default unit I93, system 2

```
DA
HAC000I ==========================================
 ...
HAC399I INSERT DEVICE I01 SYSTEM 1 ACTIVE
HAC399I INSERT DEVICE I93 SYSTEM 2 ACTIVE
 ...
HAC000I ==========================================
```

The **DA** command can be used to check the insert status.

4.63 VI TC - START INSERTION WITH TARGET COORDINATE (CHANGED)

```
VI TC=nnssrrpp[,CONT[INUE]=A[SCENDING]]
                          =D[ESCENDING]]

VI Inn,TC=nnssrrpp[,CONT[INUE]=A[SCENDING]]
                          =D[ESCENDING]]

VI Inn,s,TC=nnssrrpp[,CONT[INUE]=A[SCENDING]]
                          =D[ESCENDING]]
```

Inn	= Insertion area nn		Default: nn=01
s	= AML System		Default: s=1
nn	= Tower/rack number	or **	
ss	= Segment number	or **	or BLANK
rr	= Row number	or **	or BLANK
pp	= Position number	or **	or BLANK

CONT[INUE]=A[SCENDING] When insert started with TC= does not find any more free slots in this TC= area, an attempt is made to continue insert in the same area on the next higher tower/rack.

CONT[INUE]=D[ESCENDING] When insert started with TC= does not find any more free slots in this TC= area, an attempt is made to continue insert in the same area on the next lower tower/rack.

VI TC starts inserts with target coordinates. This command serves to insert virtual cartridges in defined slots in the dynamic area.

Using this command supports an even load spread in dynamic archives with several towers.

This command is only allowed for cartridges and incorrect use can cause processing errors.

Example:

```
VI TC=02****01
HAC399I INSERT DEVICE I01 SYSTEM 1 ACTIVE
```

Start insert; default unit I01, system 1
Target coordinate: tower 02, all segments, all rows, position 01.

```
VI I12,TC=01**01
HAC399I INSERT DEVICE I12 SYSTEM 1 ACTIVE
```

Start insert; default unit I12, system 1
Target coordinate: tower 01, all segments, row 01, all positions.

```
VI I93,2,TC=03
HAC399I INSERT DEVICE I93 SYSTEM 2 ACTIVE
```

Start insert; default unit I93, system 2
Target coordinate: tower 03, all segments, all rows, all positions.

```
DA
HAC000I ==========================================
 ...
HAC399I INSERT DEVICE I01 SYSTEM 1 ACTIVE
HAC399I INSERT DEVICE I12 SYSTEM 1 ACTIVE
HAC399I INSERT DEVICE I93 SYSTEM 2 ACTIVE
 ...
HAC000I ==========================================
```

The **DA** command can be used to check the insert status.

5 SMF RECORD SUBTYPE CHANGES

This Section describes the new and changed SMF record subtypes as compared to HCC/MVS 2.4.0. No SMF record subtypes have been deleted for HCC/MVS 3.0.0.

59F: FULL UPDATE ARCHIVES LDEV RECORD

```
  POS    LEN   DESCRIPTION
-----------------------------------------------------------------
25- 26     2   RECORD TYPE  (01 LDEV)
                  - EQU '01' IDENTIFIER
28- 29     2   SYSTEM
31- 33     3   DEVICE
               31-31  1  - LDEV TYPE (E,I,F,W,R,T..)
               32-33  2  - LDEV DEVICE NO.
35- 37     3   NO. OF REDEFINITIONS (0-255)
39- 42     4   RELATIVE POSITION OF DEVICE
44- 45     2   TYPE FLAGS
                  80   FPMA
                  40   MPMA (ONLY RACK)
                  20   HWS (ONLY TOWER)
                  10   DYNAMIC AREA
                  08   NO ASR-POINTER VALID
47- 42     1
42- 42     1
47- 47     1   DEVICE TYPE (E,F,I,R,T,D,W)
49- 50     2   DEVICE NO.
52- 53     2   SYSTEM NO.
55- 56     2   ROBOT NO.
58- 59     2   TOWER/RACK
60- 61     2   FIRST-SEGMENT
62- 63     2   LAST-SEGMENT
64- 65     2   FIRST-ROW
66- 67     2   LAST-ROW
69- 70     2   FIRST-POSITION OF FIRST ROW
71- 72     2   LAST-POSITION  OF LAST ROW
72- 74     2   NO.OF SLOTS PER ROW (10/25)
76- 78     3   NO.OF REDEFINITION
80- 80     1   ORGANIZATION (V=VERT,H=HORZ)
82- 89     8   START-COORD
91- 98     8   END-COORD
100-109   10    CAPACITY (NO.OF SLOTS)
111-120   10    NO.OF ASSIGNED SLOTS
122-125    4    OFFSET OF NEXT FREE SLOT IN DEVICE
```

59F: FULL UPDATE ARCHIVES UNIT RECORD

```
   POS    LEN   DESCRIPTION
-----------------------------------------------------------------
 25- 26     2   RECORD TYPE  (20 UNIT)
                  - EQU '20' IDENTIFIER
 28- 29     2   SYSTEM
 31- 38     8   COORD
                31-32  2  - TOWER/RACK  (ALWAYS=00)
                33-34  2  - SEGMENT     (ALWAYS=00)
                35-36  2  - ROW=UNIT    (RANGE 01-99)
                37-38  2  - POSITION    (ALWAYS 01)
 40- 41     2   ROBOT
 42- 42     1   'D'   ID
 44- 45     2   ABBA-UNIT NO (01-FF)
 46- 48     3   MVS-UNIT ADDRESS
 50- 52     3   ALTERNATE MVS-UNIT ADDRESS
 54- 58     5   DATE YYDDD OF FIRST INIT
 60- 64     5   DATE YYDDD OF LAST REFERENCE
 66- 75    10   TOTAL MOUNT COUNTER
 77- 86    10   SUM OF ABBA RESPONSE TIME (SEC'S)
 88- 97    10   TOTAL CLEANING COUNTER
 99-108    10   TOTAL SWAP COUNTER
110-119    10   TOTAL NO.OF ABBA ERRORS
121-124     4   MVS-UNIT ADDRESS MVS/ESA
126-129     4   ALTERN. MVS-UNIT ADDRESS MVS/ESA
```

59F: FULL UPDATE ARCHIVES JUKEBOX RECORD

```
POS    LEN   DESCRIPTION
-------------------------------------------------
 25- 26     2   RECORD TYPE (21 JUKEBOX)
                   - '21' IDENTIFIER
 28- 30     2   SYSTEM
 31- 32     2   ROBOT
 34- 35     2   ABBA UNIT
 37- 39     3   SUBRECORD NUMBER (000 CONTROL REC.
                001-256 FOR VOLSER RECORDS)
 41- 46     8   NAME OF JUKEBOX
 50- 54     5   MAX. CAPACITY OF JUKEBOX
 56- 60     5   USED SLOTS IN JUKEBOX
 62- 63     2   FLAGBYTE FOR CONTROL RECORD
                (NOT YET USED)
 65- 66     2   FLAGBYTE FOR VOLSER RECORD
                   80   - IF ON: VOLSER RECORD IS FULL
-------------------------------------------------
                IF VOLSER RECORD (> 000)
 25- 26     2   RECORD TYPE (21 JUKEBOX)
                   - '21' IDENTIFIER
 28- 30     2   SYSTEM
 31- 32     2   ROBOT
 34- 35     2   ABBA UNIT
 37- 39     3   SUBRECORD NUMBER (000 CONTROL REC.
                001-256 FOR VOLSER RECORDS)
 50- 51     2   FLAGBYTE FOR VOLSER RECORD
                   80   - IF ON: VOLSER RECORD IS FULL
 53- 58     6   VOLSER1 (A-SIDE)
 60- 65     6   VOLSER2 (B-SIDE)
 67- 72     6   VOLSER3 (A-SIDE)
 74- 79     6   VOLSER4 (B-SIDE)
 81- 86     6   VOLSER5 (A-SIDE)
 88- 93     6   VOLSER6 (B-SIDE)
 95-100     6   VOLSER7 (A-SIDE)
102-107     6   VOLSER8 (B-SIDE)
```

59F: FULL UPDATE ARCHIVE VOLSER RECORD

PART 1:

```
 POS       LEN   DESCRIPTION
-------------------------------------------------------------------
 25- 30     6   VOLSER (>40)
 32- 33     2   STATUS FLAG 1
                        VOLUME STATUS
                   80   - CLEANING TAPE ELSE DATATAPE
                   40   - IN ARCHIVE ELSE EJECTED
                   20   - EJECT TO BE DONE
                   10   - SCRATCH TAPE
                   08   - OD BIT MUST BE OFF
                   04   - VOLUME IN INSERT DEVICE
                   02   - IN FPMA
                   01   - ON HOME POSITION
-------------------------------------------------------------------
 32- 33     2   STATUS FLAG 1
                        REDEFINITION FOR OPTICALS
                        -------------------------
                   80   - MOUNTED ON OAD ELSE HOME OR IN JB
                   40   - IN ARCHIVE ELSE EJECTED
                   20   - EJECT TO BE DONE
                   10   - SCRATCH TAPE
                   08   - OD BIT MUST BE ON
                   04   - VOLUME IN INSERT DEVICE
                   02   - VOLUME IN JUKEBOX
                   01   - ON HOME POSITION
 35- 36     2   STATUS FLAG 2
                        RECOVERY STATUS
                   80   - IS VIRTUAL-TAPE
                   40   - MOVE TO BE DONE
                   20   - MOVE HOME
                   10   - IF ON: B-SIDE OFF: A-SIDE
                   08   - MANUAL POOL - MAPS
                   04   - IN HWS
                   02   - INITIAL VOLSER ENTRY
                   01   - INVALID COORDINATE
 38- 39     2   TEMP.VOLUME FLAG USED BY HACC/HAA
                   80   - VOLUME WAS PLACED INTO ROBQUE
                   40   - VOLUME IN USE (ON UNIT)
                   20   - VOLUME ALLOCATED FOR ROBTEST
                   10   - FORCE KES DUE TO EJECT-DEVICE
                   08   - IS USED AS CL-TAPE BY ANY ROBOT
                   04   - PMA LOAD
                   02   - PMA UNLOAD
                   80   - HAA: HOME COORD UPDATE
                   80   - HAC23059: RECOVER COORDINATE
 41- 42     2   SYSTEM-NO TO WICH TAPE BELONGS
                   01   - SYSTEM 1
                   02   - SYSTEM 2
 44- 45     2   ROBOTS WICH CAN MOUNT THE TAPE
                   01   - ROBOT 1
                   02   - ROBOT 2
 47- 48     2   TEMP.EJECT-DEVICE NO. (BINARY 01-99)
 50- 57     8   HOME-COORDINATE
            49-50  2  - TOWER/SIDE
            51-52  2  - SEGMENT
            53-54  2  - ROW
            55-56  2  - POSITION
-------------------------------------------------------------------
 50- 57     6   VOLSER A-SIDE (FOR VOLSER OF B-SIDE)
            2   FREE
```

```
-----------------------------------------------------------------
 59- 66     8   TEMP-COORDINATE
            58-59  2  - TOWER/SIDE
            60-61  2  - SEGMENT
            62-63  2  - ROW
            64-65  2  - POSITION
-----------------------------------------------------------------
 59- 66     8   RESEVED   (INDEX FOR OD A-SIDE)
-----------------------------------------------------------------
 68- 72     5   YYDDD OF LAST MO/KE
 74- 81     8   HH:MM:SS OF LAST MO/KE
 83- 87     5   YYDDD OF EJ
 89- 96     8   HH:MM:SS OF EJ
 98-102     5   YYDDD OF IN
104-111     8   HH:MM:SS OF IN
113-117     5   TOTAL USECOUNT OF VOLSER
119-123     5   USECOUNT OF VOLSER DISP=OLD
```

PART 2:

```
 25- 29     5   NO.OF SWAPS FOR THIS TAPE
 31- 35     5   CREATION DATE OF TAPE   (YYDDD)
 37- 41     5   EXPIRATION DATE OF TAPE (YYDDD)
 43- 46     4   USER-FIELD
 48- 50     3   MOUNT RESPONSE TIME (SECS)
 52- 54     3   LAST MVS-UNIT FOR MOUNT/KEEP
 56- 63     8   JOBNAME OF LAST MO/KE
 65- 68     4   LAST MVS-UNIT FOR MOUNT/KEEP MVS/ESA
 70- 71     2   EJECT STATUS
 73- 74     2   EJECT DEVICE
 76- 77     2   UPDATE STATUS
 79- 86     8   MOVE TARGET COORDINATE
```

U13: HANDLE MOUNT MESSAGE
(AFTER SYNTAX CHECK, WHEN IT IS ACCEPTED BY HACC TO WORK WITH)

POS	LEN	DESCRIPTION
25- 29	4	MVS UNIT
31- 32	2	HACC ID
34- 41	8	JOBNAME
43- 46	2	CPU ID (PRIMARY)
52- 59	8	FUNCTION (MOUNT)
61- 68	8	UNIT NAME
70- 72	3	ABBA UNIT
75- 80	6	VOLSER

U14: HANDLE KEEP MESSAGE
(AFTER SYNTAX CHECK, WHEN IT IS ACCEPTED BY HACC TO WORK WITH)

POS	LEN	DESCRIPTION
25- 29	4	MVS UNIT
31- 32	2	HACC ID
34- 41	8	JOBNAME
43- 46	2	CPU ID (PRIMARY)
52- 59	8	FUNCTION (KEEP)
61- 68	8	UNIT NAME
70- 72	3	ABBA UNIT
75- 80	6	VOLSER

491: HANDLE OAD MOUNT MESSAGE
(AFTER SYNTAX CHECK, WHEN IT IS ACCEPTED BY HACC TO WORK WITH)

POS	LEN	DESCRIPTION
25- 29	4	MVS UNIT
31- 32	2	HACC ID
34- 41	8	JOBNAME
43- 46	2	CPU ID (PRIMARY)
52- 59	8	FUNCTION (MOUNT)
61- 68	8	UNIT NAME
70- 72	3	ABBA UNIT
75- 80	6	VOLSER

492: HANDLE OAD KEEP MESSAGE
(AFTER SYNTAX CHECK, WHEN IT IS ACCEPTED BY HACC TO WORK WITH)

POS	LEN	DESCRIPTION
25- 29	4	MVS UNIT
31- 32	2	HACC ID
34- 41	8	JOBNAME
43- 46	2	CPU ID (PRIMARY)
52- 59	8	FUNCTION (KEEP)
61- 68	8	UNIT NAME
70- 72	3	ABBA UNIT
75- 80	6	VOLSER

493: HANDLE OAD FLIP MESSAGE
(AFTER SYNTAX CHECK, WHEN IT IS ACCEPTED BY HACC TO WORK WITH)

POS	LEN	DESCRIPTION
25- 29	4	MVS UNIT
31- 32	2	HACC ID
34- 41	8	JOBNAME
43- 46	2	CPU ID (PRIMARY)
52- 59	8	FUNCTION (FLIP)
61- 68	8	UNIT NAME
70- 72	3	ABBA UNIT
75- 80	6	VOLSER

494: HANDLE LOAD JUKEBOX MESSAGE
(AFTER SYNTAX CHECK, WHEN IT IS ACCEPTED BY HACC TO WORK WITH)

POS	LEN	DESCRIPTION
25- 29	4	MVS UNIT
31- 32	2	HACC ID
34- 41	8	JOBNAME
43- 46	2	CPU ID (PRIMARY)
52- 59	8	FUNCTION (LOAD)
61- 68	8	UNIT NAME
70- 72	3	ABBA UNIT
75- 80	6	VOLSER

495: HANDLE UNLOAD JB MESSAGE
(AFTER SYNTAX CHECK, WHEN IT IS ACCEPTED BY HACC TO WORK WITH)

POS	LEN	DESCRIPTION
25- 29	4	MVS UNIT
31- 32	2	HACC ID
34- 41	8	JOBNAME
43- 46	2	CPU ID (PRIMARY)
52- 59	8	FUNCTION (UNLOAD)
61- 68	8	UNIT NAME
70- 72	3	ABBA UNIT
75- 80	6	VOLSER

013: READ AND PROCESS HACCPARM1

POS	LEN	DESCRIPTION
25- 26	2	HACC ID
28-131	103	HACCPARM1 STATEMENT

101: HACC START PARAMETER

POS	LEN	DESCRIPTION
25- 26	2	HACC ID
28-131	103	HACC START PARAMETERS

No SMF record subtypes have been deleted for HCC/MVS 3.0.0.

6 HCC MESSAGES CHANGES

This Section describes the new and changed HCC messages as compared to HCC/MVS 2.4.0. No HCC messages have been deleted for HCC/MVS 3.0.0.

6.1 CHANGED HCC MESSAGES

HAC016I HACC READY FOR COMMUNICATION

Module(s): ZHC10100
Explanation: HCC is fully operational and can start automatic operation as soon as the **ROSA** command has been entered.
System action: None - keine
Operator action: None - keine

HAC017I HACC ENDED

Module(s): ZHC10100
Explanation: The HCC system task has terminated completely, the subsystem support module remains active internally for recovery reasons (refer to ABBASEND DSCA).
System action: None - keine
Operator action: None - keine

HAC117I TOTAL NO. ARCHIVE TAPES *nnnnn*

Module(s): ZHC05900

Explanation:
System action:
Operator action:

HAC118I NO. EJECTED CARTRIDGES *nnnnn*

Module(s): ZHC05900
Explanation: Information at HCC start or after FREEVOL
System action: None - keine
Operator action: None - keine

HAC119I NO. CLEAN-CARTRIDGES *nnnnn*

Module(s): ZHC05900
Explanation: Information at HCC start or after FREEVOL
System action: None - keine
Operator action: None - keine

HAC144I CARTRIDGE *volser* INSERTED

Module(s): ZHC00700
Explanation: Information message after each successful insertion with the **VI** command.
System action: None - keine
Operator action: None - keine

HAC156I SEARCH COMPLETED ON DEVICE I*nn*, SYSTEM *s*, TAPES FOR INSERT=*nnnnn*

Module(s): ZHC06900
Explanation: Completion message for an insert (**VI** command), when the search of the input/output unit is completed.
System action: None - keine
Operator action: None - keine

HAC166I TAPES TO BE EJECTED *nnnnn*

Module(s): ZHC05900
Explanation:
System action:
Operator action:

HAC246I TOTAL NO. SCRATCH TAPES *nnnnn*

Module(s): ZHC05900
Explanation: Information at HCC start or after FREEVOL
System action: None - keine
Operator action: None - keine

HAC247I AVAILABLE SCRATCH *text1 nnnnn*

Module(s): ZHC01534 - ZHC05900
Explanation: Display available SCRATCH media at HCC start or after the **DSC** command has been entered. *text1* can be:

OPTS - for optical media
TAPES - for cartridges

System action: None - keine
Operator action: Check the number of SCRATCH data media.

HAC249I ARCHIVE UPDATE FOR SCRATCH *text1* STARTED

Module(s): ZHC05900
Explanation: Message during FREEVOL process execution; Archive update with SCRATCH information started. *text1* can be:

OPTS. - for FREEVOL,... ,TYPE=OPT
TAPES - for FREEVOL,... .TYPE=CAR

System action: Archive update with new SCRATCH information starts. Message HAC250I appears after successful completion.
Operator action: None - keine

HAC250I ARCHIVE UPDATE FOR SCRATCH *text1* SUCCESSFULLY COMPLETED

Module(s): ZHC05900
Explanation: Message during FREEVOL process execution; after Archive update with SCRATCH information. *text1* can be:

OPTS. - for FREEVOL,... ,TYPE=OPT
TAPES - for FREEVOL,... .TYPE=CAR

System action: Archive has been updated.
Operator action: None - keine

HAC304A TAPE *volser* IS EJECTED.
MOUNT FOR JOB *jobname* SUSPENDED UNTIL "VI" COMPLETION

Module(s): ZHC01900
Explanation: A cartridge to be loaded is marked in the HCC as ejected.
System action: The **MOUNT** remains suspended (refer to the **DU** command). The **MOUNT** request (JOB) should be canceled when the cartridge is not available.
Operator action: The **MOUNT** request (JOB) should be canceled when the cartridge is not available. If, however, the cartridge can be inserted, HCC processes the pending **MOUNT** automatically after successful insert of the requested cartridge.

HAC419I `The APPC/*HACCn* subtask` *xxxxxxxx* `is started/terminated`

Module(s): ZHC0156E - ZHC00300
Explanation: An attempt was made to start the APPC subtask even though one was already active or to terminate the APPC even though it is not active.
System action:
Operator action:

6.2 NEW HCC MESSAGES

HAC230I FREEVOL FOR *ttt* IN PROGRESS

Module(s): ZHC05700
Explanation: Message when a FREEVOL process starts. *ttt* can be:
OPT - FREEVOL for optical disk
CAR - FREEVOL for cartridges
System action: FREEVOL process started.
Operator action: None - keine

HAC234I FREEVOL FOR OPT FOUND *tttttttttttttt* : nnnnn

Module(s): ZHC05700
Explanation: FREEVOL process statistic for optical disk via the read-in data. *tttttttttttttt* can be:

TOTAL VOLSERS	- Number of volsers read in
A-VOLSERS	- A-volsers included
B-VOLSERS	- B-volsers included
UNKNOWN VOLSERS	- Undefined included
NON-OD VOLSERS	- Non-ODs included
EJECTED VOLSERS	- No. of ejected A-volsers

System action:
Operator action: None - keine

HAC284I FREEVOL PROCESSING FOR *ttt* COMPLETED *text2*

Module(s): ZHC05700
Explanation: Message after FREEVOL process completion. *ttt* can be:
OPT - FREEVOL for opticals
CAR - FREEVOL for cartridges
text2 can be:
SUCCESSFUL
NOT SUCCESSFUL
System action: FREEVOL process has completed.
Operator action: None - keine

HAC433I The **HACCn** task for *Sx* has been started/terminated

Module(s): ZHC07000
Explanation: The task *HACCn* has been started or terminated for secondary system Sx.
System action:
Operator action:

HAC451I `*---------- End of Display ----------*`

Module(s): ZHC04501 - ZHC04502 - ZHC04503 - ZHC04504 - ZHC04505
ZHC04506 - ZHC04508 - ZHC04509 - ZHC0450A - ZHC0450B
ZHC0450C - ZHC0450D - ZHC06908
Explanation: Terminates a Display command. This message follows messages HAC450I and HAC405I.
System action: None - keine
Operator action: None - keine

HAC497I `UNIT HC-SQ M/K-SQ JOBNAME  MESSAGE`

Module(s): ZHC04500
Explanation: Column headings for the Recovery Area display. Unit number, highest recovery sequence number, MOUNT/KEEP sequence number, job name, MOUNT/KEEP messages.
System action: None - keine
Operator action: None - keine

7 HAA MENU CHANGES

This Section describes the new and changed HAA Menus as compared to HCC/MVS 2.4.0. HAA Menus deleted from HCC/MVS 2.4.0 are listed.

7.1 NEW AND CHANGED HAA MENUS

7.1.1 OPTICAL LIBRARY

Volsers for optical media must now be defined when these are included in this Archive.

```
HAA 01/1    HACC ARCHIVE ADMINISTRATION (HAA 3.0.0/M00)             OBI1WD2
==============================================================================

PFK
---    ---------------------
 1     CARTRIDGE LIBRARY
 2     OPTICAL LIBRARY
 3
 4     SAVE/RESTORE/COPY/EXPORT/IMPORT
 5     STATISTICS
 6     DISPLAY ALL DEVICES
 7
 8
 9     GENERATE ARCHIVE FROM HACCPARM
10
11
12     EXIT

NO.COORDS=00000,NO.VOLUMES=00000-VOLSER=OBIS02,DSN=OBI1WD1.H301ARC
   ARCHIVE CLUSTER IS EMPTY, APPLY „GENERATE ARCHIVE FROM HACCPARM“

==============================================================================
ENTER:      <-- SELECT BY NUMBER OR PFK
```

Select Menu item 2 - OPTICAL LIBRARY with
- [PF 2] or
- [2] and [ENTER]

7.1.1.1 DEFINE AND ASSIGN OPTICAL VOLSERs

```
HAA 03/0       OPTICAL LIBRARY MAIN MENU                          OBI1WD2
==========================================================================
PFK   FUNCTION
---   -------------------------------
    1    DEFINE AND ASSIGN VOLSERS
    2    DISPLAY / MODIFY / DELETE VOLSERS
    3
    4    DISPLAY JUKEBOX(ES)
    5    DISPLAY OPTICAL DEVICES
    6
    7
    8
    9
   10
   11
   12    RETURN TO MAIN MENUE

==========================================================================
ENTER:      <-- SELECT BY NUMBER OR PFK
```

This example continues with the definition and assignment of volsers.

Select Menu item 1 - DEFINE AND ASSIGN VOLSERS with
- [PF 1] or
- [1] and [ENTER]

```
HAA 31/0       OPTICAL LIBRARY DEFINE AND ASSIGN VOLSERS          OBI1WD2
==============================================================================
VOLSER VOLSER STAT COORD.   JB-NAME  VOLSER VOLSER STAT COORD.  JB-NAME
A-SIDE B-SIDE      A-SIDE            A-SIDE B-SIDE      A-SIDE
od033a od033b ob   01020101

Enter data or terminate
ENTER:
```

The following display appears after [ENTER] has been pressed:

```
HAA 31/0       OPTICAL LIBRARY DEFINE AND ASSIGN VOLSERS          OBI1WD2
==============================================================================
VOLSER VOLSER STAT COORD.   JB-NAME  VOLSER VOLSER STAT COORD.  JB-NAME
A-SIDE B-SIDE      A-SIDE            A-SIDE B-SIDE      A-SIDE
OD033A OD033B OB   01020101

ENTER: Y Input correct ??? Enter Y or N
```

Enter [Y] - when the input is correct or
[N] - when changes are necessary.

```
HAA 31/0       OPTICAL LIBRARY DEFINE AND ASSIGN VOLSERS          OBI1WD2
==============================================================================
VOLSER VOLSER STAT COORD.   JB-NAME  VOLSER VOLSER STAT COORD.  JB-NAME
A-SIDE B-SIDE      A-SIDE            A-SIDE B-SIDE      A-SIDE

ENTER: _  C to continue input, G to generate or PF12 to terminate
```

Enter [C] - when further input is necessary or
[G] - to generate optical volsers or
[PF12] - to terminate input.

The following display appears after [G] has been entered:

```
Optical VOLSER Creation in progress
ARCHIVE SAVED, RECORDS=0006581,SAVEDSN=OBI1WD2.HACCARCH.SAVE.P.D...
NO.OF RECORDS COPIED=006581,OUPUT-DSN=OBI1WD2.HACCARCH.SAVE.P.V...
TOTAL RECORDS=006583,OUPUT-DSN=OBI1WD2.HACCARCH.SAVE.P.VOLSER
SORT COMPLETED. SORTOUT-DSN=OBI1WD2.HACCARCH.SAVE.P.VOLSER
<== DECIDE RELOAD ARCHIVE WITH NEW DATA:
<== ENTER "YES" TO PROCEED OR ANY CHAR TO IGNORE RELOAD
YES
ARCHIVE LOADED,NO.OF RECORDS=0006583,DSN=OBI1WD2.HACCARCH.SAVE.P.V..
Creation of optical volsers completed
<-- ANY CHAR TO CONFIRM
```

Pressing any key returns to the OPTICAL LIBRARY MAIN MENU.

7.1.1.2 DISPLAY / MODIFY / DELETE VOLSERs

```
HAA 03/0       OPTICAL LIBRARY MAIN MENU                          OBI1WD2
==========================================================================
PFK   FUNCTION
---   -------------------------------
    1    DEFINE AND ASSIGN VOLSERS
    2    DISPLAY / MODIFY / DELETE VOLSERS
    3
    4    DISPLAY JUKEBOX(ES)
    5    DISPLAY OPTICAL DEVICES
    6
    7
    8
    9
   10
   11
   12    RETURN TO MAIN MENUE

==========================================================================
ENTER:      <-- SELECT BY NUMBER OR PFK
```

This example continues with the display of volsers.

Select Menu item 2 - DISPLAY/MODIFY/DELETE VOLSERS with
- [PF 2] or
- [2] and [ENTER]

```
HAA 32/0  OPTICAL LIBRARY DISPLAY / MODIFY / DELETE VOLSERS OBI1WD2
==============================================================================

SEL A-VOLS B-VOLS ST SYS ROB HOMECOOR JB-NAME
  s   OD001A OD001B OB  1  1 - 01020101
  _   OD002A OD002B OE  1  1 - 01020103
  _   OD003A OD003B OB  1  1 - 01020105
  _   OD004A OD004B OB  1  1 - 01020107
  _   OD005A OD005B OB  1  1 - 01020109
  _   OD006A OD006B OB  1  1 - 01020111
  _   OD007A OD007B OB  1  1 - 01020113
  _   OD008A OD008B OB  1  1 - 01020115
  _   OD009A OD009B OB  1  1 - 01020117
  _   OD010A OD010B OB  1  1 - 01020119
  _   OD011A OD011B OB  1  1 - 01020121
  _   OD012A OD012B OB  1  1 - 01020123
  _   OD013A OD013B OB  1  1 - 01020125
  _   OD014A OD014B OB  1  1 - 01020127
  _   OD015A OD015B OB  1  1 - 01020129
  _   OD016A OD016B OB  1  1 - 01020201
  _   OD017A OD017B OB  1  1 - 01020203
  _   OD018A OD018B OB  1  1 - 01020205
Enter S to display, U to Update or D to Delete a VOLSER at SEL field
ENTER:   X or PF12 to terminate / PF7 or PF8 to scroll
```

Enter [S] - for a detailed display or
[U] - to modify the data for a selected volser or
[D] - to delete a selected volser or
[X] - to terminate the display or
[PF7] - to scroll the display or
[PF8] - to scroll the display or
[PF12] - to terminate the display.

```
HAA 32/2                OPTICAL LIBRARY DISPLAY OD001A
==============================================================================

A-VOLSER: OD001A   B-VOLSER: OD001B   HOMECOOR: 01020101   LIBNAME:
SYSTEM #: 1   ROBOT #: 1 -   STATUS : OB/HOME POS

STATISTICS:
LAST MO/KE AT DATE: ---   AND TIME: ---
LAST EJECT AT DATE: ---   AND TIME: ---
LAST INSRT AT DATE: ---   AND TIME: ---

FLAG SETTINGS
AROSTA MOU: 0  INA: 1  EJD: 0  SCR: 0  OD : 1  IDV: 0  JB : 0  HOM: 1
ARSTA1 VIR: 0  MVD: 0  MVH: 0  ALI: 0  IP : 0  HWS: 0  INT: 0  INV: 0
ARFLG  INQ: 0  IUS: 0  ART: 0  FKE: 0  CLC: 0  PML: 0  PMU: 0



ENTER   Press any key to continue
```

This display is identical for all three functions (DISPLAY, MODIFY, DELETE).

MODIFY saves all fields that can be modified.

The following message appears in the message line for DELETE:

```
   .
   .
   .
ENTER   Y to confirm
```

The function is exited after completion or when [PF12] is pressed. The application returns to the OPTICAL LIBRARY MAIN MENU.

7.1.1.3 DISPLAY JUKEBOXES / OPTICAL DEVICES

The OPTICAL LIBRARY MAIN MENU also serves to display the defined jukeboxes and optical units (towers, racks, insertion/rejection units).

```
HAA 03/0       OPTICAL LIBRARY MAIN MENU                          OBI1WD2
==============================================================================
PFK   FUNCTION
---   -------------------------------
    1    DEFINE AND ASSIGN VOLSERS
    2    DISPLAY / MODIFY / DELETE VOLSERS
    3
    4    DISPLAY JUKEBOX(ES)
    5    DISPLAY OPTICAL DEVICES
    6
    7
    8
    9
   10
   11
   12    RETURN TO MAIN MENUE

==============================================================================
ENTER:      <-- SELECT BY NUMBER OR PFK
```

Select Menu item 4 - DISPLAY JUKEBOX(ES) with
- [PF 4] or
- [4] and [ENTER]

```
HAA 03/4     OPTICAL LIBRARY DISPLAY JUKEBOX(ES)                  OBI1WD2
==============================================================================
DEV LIBNAME   SYS ROB CAPACITY   USED
D07 ODLIB1     1   1       144      3
D09 ODLIB2     1   1       144      3









==============================================================================
ENTER:     Any char or PF12 to terminate
```

The function is exited after completion or when [PF12] is pressed. The application returns to the OPTICAL LIBRARY MAIN MENU.

```
HAA 03/0       OPTICAL LIBRARY MAIN MENU                          OBI1WD2
==============================================================================
PFK   FUNCTION
---   -------------------------------
    1    DEFINE AND ASSIGN VOLSERS
    2    DISPLAY / MODIFY / DELETE VOLSERS
    3
    4    DISPLAY JUKEBOX(ES)
    5    DISPLAY OPTICAL DEVICES
    6
    7
    8
    9
   10
   11
   12    RETURN TO MAIN MENUE

==============================================================================
ENTER:      <-- SELECT BY NUMBER OR PFK
```

Select Menu item 5 - DISPLAY OPTICAL DEVICES with

- [PF 5] or
- [5] and [ENTER]

```
HAA 03/2     DISPLAY ARCHIVE ALL LIBRARY DEVICES                  OBI1WD2
==============================================================================
DEV S ROB  NO  SEG   ROW   POS  NO/ROW  SLOTS  TYPE
E01 1 1    00 02-02 01-03 01-10   10       30  CAR
E02 1 1    00 02-02 06-07 01-30   30       60  OPT
F01 1 1    00 03-03 10-11 01-10   10       20  CAR
I01 1 1    00 01-01 01-03 01-10   10       30  CAR
I02 1 1    00 01-01 04-05 01-30   30       60  OPT
T01 1 1    01 01-01 01-18 01-10   10      180  CAR
T01 1 1    01 01-01 01-18 01-10   10      180  CAR/DYNA
T01 1 1    01 03-32 01-18 01-10   10     5400  CAR
T01 1 1    01 02-02 01-13 01-30   30      390  OPT
W01 1 1    00 00-00 00-00 01-01   01        1  CAR









==============================================================================
ENTER:    <-- ANY CHAR TO CONFIRM
```

7.1.2 CARTRIDGE LIBRARY MENU

```
HAA 02/1                    CARTRIDGE LIBRARY                    UserID

==============================================================================

  PFK   FUNCTION
  ---   -------------------------------
   1    DISPLAY COMPLETE VOLSER ENTRIES
   2    DISPLAY/MODIFY VOLSER ENTRIES
   3    DEFINE VOLSERS TO ABBA LIBRARY
   4    ASSIGN VOLSERS TO HOME-COORDINATES
   5    UNASSIGN VOLSERS FROM HOME-COORDINATES (IMMEDIATE)
   6    UNASSIGN AND ASSIGN TO "VIRTUAL" (AFTER CARTRIDGE IS EJECTED)
   7    ASSIGN VOLSERS TO "VIRTUAL"
   8    DISPLAY CARTRIDGE LIBRARY DEVICES
   9    DELETE VOLSERS
  10    PRINT VOLSER-COORDINATE ASSIGNMENTS
  11
  12    RETURN TO MAIN MENUE




==============================================================================
ENTER:      <-- SELECT BY NUMBER OR PFK
```

Use [PF12] to return to the Main Menu.

7.2 DELETED HAA MENUS

The HAA Menu HAA 03/1 (GENERATE COORDINATE RECORDS) has been deleted from HCC/MVS 2.4.0.

INDEX

3

3480 UNITS 2-3
3490 UNITS 2-3

A

ACC 4-1
ACL 2-3
ACV 4-3
ADD 4-4
ALLOC 4-5
AML units 2-4, 2-17
AMU
  ARCHIVE MIRROR TRANSFER 4-88
  CORRECTION 4-79
AR
  ARCHIVE MIRROR TRANSFER 4-88
  CORRECTION 4-79
ARCHIVE COMPARISON
  COORDINATE 4-1
  VOLSER 4-3
AUTOFREEVOL 2-1

B

BASIC PRIORITY 4-74

C

CANCEL LABEL 4-6
CL 4-7
CLEANING
  START 4-73
  STOP 4-69
COMMAND
  ACC 4-1
  ACV 4-3
  ALLOC 4-5
  CANCEL LABEL 4-6
  CL 4-7
  CX (LU 6.2) 4-8
  CX CMD 4-8
  CX GETMAJOR 4-9
  CX MSG 4-10
  CX SETMAJOR 4-11
  CX SETMINOR 4-12
  DC 4-14
  DEL 4-16
  DISPLAY 4-17
  DLC 4-26
  DLV 4-29
  DRT 4-30
  DSC 4-31
  DU 4-33
  EJ 4-45
  FLIP 4-47
  FMA 4-48
  FMD 4-50
  FREE 4-51
  FREEVOL 4-52
  HELP 4-56
  INC 4-57
  INV 4-59
  K 4-60
  LCK 4-62
  LJB 4-63
  M 4-64
  MOVEH 4-66
  MOVET 4-67
  PCL 4-69
  PLCK 4-70
  PRT 4-71
  PTLI 4-72
  SCL 4-73
  SET PRTY 4-74
  SETUP 4-77
  SETUP AR 4-79
  SLCK 4-80
  START LABEL 4-82
  STLI 4-83
  TLI 4-84
  UJB 4-85
  ULC 4-86
  ULU 4-90
  ULV 4-88
  UPC 4-91
  UPV 4-92
  VARY 4-93
  VI 4-94
  VI TC 4-96
CONTROL
  PRIORITIES 4-74
COORDINATE
  DISPLAY 4-14
  LDEV statement 2-4, 2-17
  Position 2-4, 2-17
  Row 2-4, 2-17
  Segment 2-4, 2-17
COORDINATE ASSIGNMENT 7-1, 7-10
CORRECTION
  AR/AMU 4-79
  UNIT LOG 4-77
CX (LU 6.2) 4-8
CX CMD 4-8
CX GETMAJOR 4-9
CX MSG 4-10
CX SETMAJOR 4-11
CX SETMINOR 4-12, 4-13

D

DC 4-14
DEL 4-16

DISPATCHING 4-76
DISPLAY 4-17
ABS USERS 4-25
ACTIVE FUNCTIONS 4-21
ACTIVE ROBOT 4-30
COMMAND OVERVIEW 4-56
CONNECTION STATUS 4-23
CONTACTS 4-24
COORDINATE 4-14
CSA DATA 4-19, 4-20
FURTHER FUNCTIONS 4-17
HCC MESSAGE 4-23
JUKEBOX 4-21
ROBOT FUNCTIONS 4-18
ROBOT LOCATION 4-22
SCRATCH TAPES 4-31
ZHCLAB 4-22
DLC 4-26
DLV 4-29
DOWNLOAD ARCHIVE
COORDINATE 4-26
VOLSER 4-29
DRT 4-30
DSC 4-31
DU 4-33

E

EJ 4-45
EJECT
MEDIUM 4-45

F

FLIP 4-47
FMA 4-48
FMD 4-50
FOREIGN MOUNT ENTRY
ADD 4-48
DELETE 4-50
FPMADEF 2-2
FREE 4-51
FREEVOL 4-52

H

HACPARM1
AUTOFREEVOL 2-1
FPMADEF 2-2
KEEPWT 2-3
LDEV 2-4, 2-17
ODSCR 2-25
STORGRP 2-28, 2-29
TMSEXIT 2-30
UNIT 2-31
HCC
PRIORITY CONTROL 4-76
HELP 4-56

I

I/O-S 4-85
INC 4-57
INDIVIDUAL KEEP 4-60
INSERTION
START, TARGET COORDINATE 4-96
INSTALLATION
AML START 3-7
START COMMAND 3-1
INTERCOMMUNICATION
LU 6.2 ..... 4-8, 4-93
INV 4-59
INVENTORY
COORDINATE 4-57
VOLSER 4-59

J

JUKEBOX
INFORMATION 4-21
LOAD 4-63
UNLOAD 4-85

K

K 4-60
KEEP 4-60
KEEPWT 2-3

L

LABEL CHECK
DEFINE 4-62
START 4-80
STOP 4-70
LCK 4-62
LDEV 2-4, 2-17
LJB 4-63
LU 6.2
INTERCOMMUNICATION 4-8
MAJOR COMPLEX 4-9
SEND MESSAGE 4-10
SET MAJOR 4-11
SET MINOR 4-12

M

M 4-64
MAJOR
COMPLEX LU 6.2 4-9
SET LU 6.2 4-11
MEDIUM
EJECT 4-45
MINOR
SET LU 6.2 ..... 4-12, 4-13, 4-68
MOUNT
EXAMPLES 4-65
INDIVIDUAL 4-64
MOVE
HOME COORDINATE 4-66
TEMP. COORDINATE 4-67

MOVEH 4-66
MOVET 4-67

O

OAD 4-47
ODSCR 2-25
OPTICAL DISK
  TURNOVER 4-47

P

PCL 4-69
PDSNEXC 2-26
PLCK 4-70
PRIORITY 4-74
PRT 4-71
PTLI 4-72

R

ROBOT
  DISPLAY LOCATION 4-22
  DISPLAY ROBOT FUNCTIONS 4-18
  TEST 4-4, 4-16, 4-30
  TEST STOP 4-71

S

SCL 4-73
SCRATCH TAPES
  DETERMINE 4-52
  DISPLAY 4-31
SEND MESSAGE
  LU 6.2 4-10
SET MSG 4-23
SET PRTY 4-74
SETUP 4-77
SETUP AR 4-79
SLCK 4-80
START
  AUTOM. TAPE UNIT CLEANING 4-73
  INSERTION 4-94
  LABEL CHECK 4-80
  LABEL INITIALIZATION 4-83
  LABEL PROCESS 4-82
START COMMAND 3-1
START INSERTION 4-94
START LABEL 4-82
STLI 4-83
STOP
  AUTOM. TAPE UNIT CLEANING 4-69
  ROBOT TEST 4-71
  TAPE LABEL INITIALIZATION 4-72
STORGRP 2-28, 2-29

T

TAPE LABEL INIT
  DEFINE 4-84
TAPE UNIT
  CLEANING 4-7
TAPE UNITS
  RELEASE 4-51
  RESERVING 4-5
  START AUTOM. CLEANING 4-73
  STOP AUTOM. CLEANING 4-69
TEST
  ROBOT 4-71
TLI 4-84
TMS EXIT 2-30

U

UJB 4-85
ULC 4-86
ULU 4-90
ULV 4-88
UNIT 2-31
UNIT LOG
  CORRECTION 4-77
UNLOAD TAPE UNIT 4-90
UPC 4-91
UPDATE ARCHIVE
  COORDINATE 4-91
  VOLSER 4-92
UPLOAD ARCHIVE
  COORDINATE 4-86
  VOLSER 4-88
UPV 4-92
USER EXIT
  TMS 2-30

V

VARY 4-93
VI 4-94
VI TC 4-96

Z

ZHCLAB 4-22