Device and File Considerations

This section provides information for the following device and system file related topics:

- installing on fixed-block addressing (FBA) devices; and
- defining new devices.
- FBA Devices
- Adding New Devices
- General Rules for Defining Device Block Sizes

FBA Devices

All device definitions for Adabas control statements for FBA disks should specify one of the following devices types:

- FBA SCSI devices: Specify device types of 5512, 6512, or 7512.
- Virtual FBA devices: Specify device types of 5121, 5122, or 5123.

Note:
Virtual FBA devices are not permanent and are, therefore, only suitable for holding temporary or work data sets.

Choose a device type based on the block sizes given in the following tables:

SCSI Device Types:

Dev Type	Asso blksz	Data blksz	Work blksz	Temp blksz	Sort blksz	PLOG blksz	CLOG blksz
5512	2048	4096	4096	4096	4096	4096	4096
6512	4096	8192	8192	8192	8192	8192	8192
7512	4096	16384	16384	16384	16384	16384	16384

Virtual FBA Device Types:

Dev Type	Asso blksz	Data blksz	Work blksz	Temp blksz	Sort blksz	PLOG blksz	CLOG blksz
5121	2048	4096	4096	4096	4096	4096	4096
5122	4096	8192	8192	8192	8192	8192	8192
5123	4096	16384	16384	16384	16384	16384	16384

The pseudo-cylinder for each of these devices has a different number of blocks as described below:

```
5512 cylinder = FBA blocks/1024
6512 cylinder = FBA blocks/1024
7512 cylinder = FBA blocks/1024
5121 cylinder = FBA blocks/960
5122 cylinder = FBA blocks/960
5123 cylinder = FBA blocks/960
```

The size definitions for FBA devices on Adabas control statements can specify the number of pseudo-cylinders or the number of Adabas blocks (RABNs).

Make sure that the number of FBA blocks in the minidisk is a multiple of the pseudo-cylinders listed above. In addition, the minidisk size must be one pseudo-cylinder larger than the size specified in the Adabas size definitions:

- An SCSI pseudo-cylinder comprises 1,024 elements of 512 bytes each, or 512 K per pseudo-cylinder.
- A virtual FBA pseudo-cylinder comprises 960 elements of 512 bytes each, or 480 K per pseudo-cylinder.

Adding New Devices

Support for new device types that include user-defined block sizes can be implemented in Adabas by modifying one of the table of device-constant entries (TDCEs) reserved for this purpose. A TDCE is X’40’ bytes long and the first free TDCE can be identified by X’0000’ in its first two bytes (TDCDT).

Under z/VM for all versions of Adabas prior to Version 6.2, the address of the first TDCE is at offset ADAIOR+ X‘34’.

For Adabas Version 6.2 and 7.1, TDCE entries are in the ADAIOR CSECT TDCON: the first TDCE entry is at offset 0; the first free TDCE entry is at offset X‘680’.

This information is valuable when adding an additional TDCE entry.

The ADDEVICE EXEC distributed on the release tape can be used to add new device types. This EXEC creates zap files called USERDEV*n* ZAP to modify the TDCEs. It also modifies the table of valid device names DEVICE TABLE and the USERDEV TABLE. These tables are used for verification during the online installation procedure.

- Information to be Zapped into the First Free TDCE

Information to be Zapped into the First Free TDCE

The information in the following tables must be zapped into the first free TDCE. The rules described in the section *General Rules for Defining Device Block Sizes* must be followed when changing the TDCE.

Label	Offset	Contents
TDCDT	00	Device type in unsigned decimal (X’3385’), must be numeric, and unique among all TDCEs
TDCKSN	02	Constant set number: must be uniquely chosen from the values X’28’ (reserved for BS2000 device type 2006), X’2B’, or X’2E’

Label	Offset	Contents
TDCF	03	The flag bit must be set—TDCFFBA (X’80’) for FBA/PAM devices or TDCFCKD (X’40’) for CKD devices
TDCDT1	04	see note below
TDCDT2	05	see note below
TDCDT3	06	see note below
TDCDT4	07	see note below
TDCMSBS	08	Refer to the TDCMSBS default table in appendix A in *Maximum Sequential Block Size* in the Adabas z/OS installation instructions for more system and device related information.
TDCTPC	0A	Number of tracks per cylinder
TDCCIPT	0C	Number of FBA blocks or PAM pages per track (if TDCFFBA is set). For BS2000 less than or equal to 16.
TDCBPCI	0E	Number of bytes per FBA block or PAM page (2048 if TDCFFBA is set)
TDCABPT	10	Number of Associator blocks per track
TDCABS	12	Associator block size
TDCACPB	14	Number of FBA blocks or PAM pages per Associator block (if TDCFFBA is set)
TDCDBPT	16	Number of Data Storage blocks per track
TDCDBS	18	Data Storage block size
TDCDCPB	1A	Number of FBA blocks or PAM pages per Data Storage block (if TDCFFBA is set)
TDCWBPT	1C	Number of Work blocks per track
TDCWBS	1E	Work block size
TDCWCPB	20	Number of FBA blocks or PAM pages per Work block (if TDCFFBA is set)
TDCTSBPT	22	Number of TEMP or SORT blocks per track (if TDCFFBA is set)
TDCTSBS	24	TEMP or SORT block size
TDCTSCPB	26	Number of FBA blocks or PAM pages per TEMP or SORT block (if TDCFFBA is set)
TDCPBPT	28	Number of PLOG blocks per track
TDCPBS	2A	PLOG block size
TDCPCPB	2C	Number of FBA blocks or PAM pages per PLOG block (if TDCFFBA is set)
TDCCBPT	2E	Number of CLOG blocks per track
TDCCBS	30	CLOG block size

Label	Offset	Contents
TDCCCPB	32	Number of FBA blocks or PAM pages per CLOG block (if TDCFFBA is set)

Note:
One or more z/VM codes for identifying the device type: the UCB unit type from UCBTBYT4.

In addition, the length of a sequential protection log block may have to be increased. Under z/VM, this length is contained in the corresponding PTT entry in CSECT I_PTT of the load module ADAIOR. PTT entries begin at offset 0 into CSECT I_PTT.

Each PTT entry is X’10’ bytes long and has the structure shown below:

Label	Offset	Contents
PTTPN	00	Program number
PTTFT	01	File type
PTTN	02	DD name characters 2 - 8
PTTF	08	Flags: OUT (X’80’) output BSAM (X’40’) BSAM BACK (X’20’) read backwards JCL (X’10’) BLKSIZE/LRECL/RECFM taken from DATADEF statement or label UNDEF (X’04’) undefined record format VAR (X’02’) variable record format
-	09	Reserved
PTTMBS	0A	Maximum block size
-	0C	Reserved

The PTT entry for the sequential protection log can be identified by X‘12F1’ in its first two bytes.

General Rules for Defining Device Block Sizes

The following general rules must be followed when defining Adabas device block sizes:

- all block sizes must be multiples of 4
- a single block cannot be split between tracks (block size must be less than or equal to the track size)

Block Rules for ASSO/DATA

The following rules apply for Associator and Data Storage blocks:

- Associator block size must be greater than one-fourth the size of the largest FDT, and should be large enough to accept definitions in the various administrative blocks (RABN 1 - 30) and in the FCB

- The block sizes for Associator and Data Storage should be a multiple of 256, less four bytes (for example, 1020) to save Adabas buffer pool space
- The Associator and Data Storage block sizes must be at least 32 less than the sequential block size
- Data Storage block size must be greater than: (maximum compressed record length + 10 + padding bytes)

Block Rule for WORK

The Work block size must be greater than either (maximum compressed record length + 110) or (Associator block size + 110), whichever is greater.

Block Rules for TEMP/SORT

If ADAM direct addressing is used:

```
size > (maximum compressed record length + ADAM record length + 24);
size > 277 (maximum descriptor length + 24)
```

However, TEMP and SORT are generally read and written sequentially; therefore, the larger the TEMP/SORT block size, the better.

Block size for TEMP and SORT must be greater than the block size for Data Storage.

Block Rule for PLOG or SIBA

The following rules apply for PLOG or SIBA blocks:

- The PLOG or SIBA block size must be greater than either (maximum compressed record length + 110) or (Associator block size + 110), whichever is greater.
- It is also recommended that PLOG/SIBA be defined larger than the largest Data Storage block size. This avoids increased I/O caused by splitting Data Storage blocks during online ADASAV operations.

The block size (BLKSIZE) of a sequential file is determined as follows:

```
if PTTF(JCL) then BLKSIZE is taken from file assignment statement or label;
if PTTMBS > 0 then BLKSIZE = PTTMBS;
if PTTMBS = 0 then
if tape then BLKSIZE = 32760;
else BLKSIZE = TDCMSBS;
else if BLKSIZE in file assignment statement or label then use it;
if PTTF(OUT) then
if QBLKSIZE > 0 then BLKSIZE = QBLKSIZE;
if tape then BLKSIZE = 32760;
else BLKSIZE = TDCMSBS;
else error.
```

Note:
QBLKSIZE is an ADARUN parameter.

Using 3480/3490 Tape Cartridge Compression (IDRC)

The use of hardware compression (IDRC) is not recommended for protection log files. The ADARES BACKOUT function will run much longer when processing compressed data. Also, the BACKOUT function is not supported for compressed data on z/VM systems.