REORDATA: Reorder Data Storage

The REORDATA function reorders Data Storage for *all* files. Files not specified are reordered according to their existing definitions. The REORDATA function does not reorder ADAM files. However, it can be used to relocate an ADAM file to different RABNs.

This function requires exclusive EXF control of the database files involved in the operation. In addition, parts of the database are overwritten during ADAORD execution, so we recommend that you back up the database (or file) using the ADASAV utility first, before running ADAORD functions.

This is the syntax of the ADAORD REORDATA function:

```
ADAORD  REORDATA   [FILE = file-number ]
                   [ALLOCATION = { FORCE | NOFORCE } ]
                   [DATAPFAC =  padding-factor  ]
                   [DATAVOLUME = 'Data-Storage-extent-volume' ]
                   [DSDEV = device-type ]
                   [DSRABN = starting-rabn]
                   [DSRELEASE]
                   [DSSIZE =size ]
                   [MAXRECL = record-length ]
                   [SORTSEQ = { descriptor | ISN | physical-sequence } ]
             [LIP = { isn-pool-size  |  16384 } ]
             [LPB = { prefetch-buffer-size  |  ADARUN-lu } ]
             [NOUSERABEND]
             [RAID]
             [TEST]
```

This chapter covers the following topics:

- Optional Parameters and Their Subparameters

- Examples

Optional Parameters and Their Subparameters

ALLOCATION: Action to Follow File Extent Allocation Failure

> ALLOCATION specifies the action to be taken if file extent allocations cannot be obtained according to the placement parameter DSRABN.

> By default (that is, ALLOCATION=FORCE), the utility terminates with error if any file extent allocation cannot be met according to RABN placement parameters.

> If ALLOCATION=NOFORCE is specified and any allocation with placement parameter fails, the utility retries the allocation without the placement parameter.

DATAPFAC: Data Storage Padding Factor

DATAPFAC specifies the new Data Storage padding factor, which is the percentage of each Data Storage block reserved for record expansion when the file is reordered. A value in the range 1-90 may be specified (see the ADALOD LOAD DATAPFAC parameter discussion for more information about setting the padding factor). If this parameter is omitted, the current padding factor for the file is used.

DATAVOLUME: Data Storage Extent Volume

Note:
The value for the DATAVOLUME parameter must be enclosed in apostrophes.

DATAVOLUME specifies the volume on which the file's Data Storage space (DS extents) are allocated. If the number of blocks requested with DSSIZE cannot be found on the specified volume, ADAORD allocates the remaining blocks on other volumes according to its default allocation rules.

If DSRABN is specified, DATAVOLUME is ignored for the related file.

If DATAVOLUME is not specified, the Data Storage space is allocated based on the current size of the file. The DSRELEASE parameter also affects the amount of space required.

DSDEV: Data Storage Device Type

DSDEV is the file's Data Storage device type. The specified device type must already be defined to Adabas, normally when the database was created or by the ADADBS utility's ADD function.

If DSDEV is not specified, ADAORD attempts to allocate the file on the device type used before reordering.

DSRABN: Data Storage Starting RABN

The beginning RABN for the specified file's Data Storage extent. If this parameter is omitted, ADAORD assigns the starting RABN.

DSRELEASE: Release Unused Data Storage Blocks

Specifying DSRELEASE releases unused Data Storage (DS) blocks belonging to the specified file. If DSRELEASE is not specified, ADAORD allocates *at least* the number of DS blocks that were allocated before the file was reordered.

Note:
Adabas calculates the file extent size using any changed padding factor or block size values *before* the file is reordered.

DSSIZE: Data Storage Size

DSSIZE is the number of blocks or cylinders to be allocated for the file's Data Storage (DS) logical extent. If the value is blocks, it must be followed by a "B" (for example, "2000B").

If this parameter is omitted, ADAORD computes the file extent size in proportion to any increase or decrease in the DATAPFAC padding factor used.

FILE: File Number

FILE is the file to which the following parameters apply. Each specified file and its parameters should be on a separate ADAORD statement following the ADAORD REORDATA statement.

For any file whose number is not specified, the file is reordered using the current physical sequence, and the current Data Storage padding factor and space allocation are retained.

LIP: ISN Buffer Pool Size

The LIP parameter can be used to decrease the number of Associator I/O operations when recreating the address converter. For best performance, specify a size that accepts all ISNs of the largest file to be processed.

LIP specifies the size of the ISN pool for containing ISNs and their assigned Data Storage RABNs. The value may be specified in bytes as a numeric value ("2048") or in kilobytes as a value followed by a "K" ("2K"). The default for LIP is 16384 bytes (or 16K).

The length of one input record is ISNSIZE + RABNSIZE. Thus the entry length is at least 6 bytes (the ISNSIZE of the file is 3 and the RABNSIZE of the database is 3) and at most 8 bytes (the ISNSIZE is 4 and the RABNSIZE is 4).

Note:
When ADAORD is processing files that contain spanned records with secondary ISNs, a second LIP will be allocated to contain these ISNs.

LPB: Prefetch Buffer Size

LPB specifies the size, in bytes, of the internal prefetch buffer. The maximum value is 32760 bytes. The default depends on the ADARUN LU parameter. ADAORD may reduce a specified LPB value if the LU value is too small.

MAXRECL: Maximum Compressed Record Length

Use the MAXRECL parameter to change the maximum record length, after compression, permitted in the file. Specifying MAXRECL has two effects:

The DATA data set for the file can be allocated only to devices that support the specified length.

If the file contains Data Storage records that exceed the specified length, ADAORD abends and prints ERROR-126 (Data Storage record too long).

If MAXRECL is not specified, the maximum compressed record length does not change.

NOUSERABEND: Termination without Abend

When an error is encountered while the function is running, the utility prints an error message and terminates with user abend 34 (with a dump) or user abend 35 (without a dump).

If NOUSERABEND is specified, the utility will *not* abend after printing the error message. Instead, the message "utility TERMINATED DUE TO ERROR CONDITION" is displayed and the utility terminates with condition code 20.

RAID: Action to Follow Determination That File Does Not Exist

The RAID parameter instructs ADAORD to ignore any FILE parameters that refer to a file that does not exist in the database.

If RAID is not specified (the default), ADAORD terminates with an error message when it encounters a FILE parameter referring to a file that does not exist in the database.

The RAID parameter is provided for use in recovery jobs built by the Adabas Recovery Aid (ADARAI).

SORTSEQ: Record Processing Sequence

SORTSEQ determines the sequence in which the file is to be processed. If this parameter is omitted, the records are processed in physical sequence.

If a descriptor is specified, the file is processed in the logical sequence of the descriptor values. *Do not* use a null-suppressed descriptor field, a hyperdescriptor, a phonetic descriptor, a multiple-value field, or a descriptor contained in a periodic group.

Note:
Even when the descriptor field is not null suppressed, the record is *not* represented in the inverted list if the descriptor field or a field following it has never been initialized (held a value). Therefore, the record will be dropped when the utility is executed.

If ISN is specified, the file is processed in ascending ISN sequence. For the Adabas checkpoint or security file, only SORTSEQ=ISN is allowed.

TEST: Test Syntax

This parameter tests the operation syntax without actually performing the operation. Only the syntax of the specified parameters can be tested; not the validity of values and variables.

Examples

Example 1:

```
ADAORD REORDATA
```

Data Storage for the entire database is to be reordered.

Example 2:

```
ADAORD     REORDATA
ADAORD               FILE=1,DSRABN=1000,DSSIZE=200B
ADAORD               FILE=4,SORTSEQ=AA
ADAORD               FILE=5,DATAPFAC=15
```

Data Storage is to be reordered. Data Storage for file 1 is to begin with RABN 1,000 with 200 blocks to be allocated. File 4 is to be reordered using descriptor AA for sequence control. The Data Storage block padding factor for file 5 is to be set to 15 percent. All other database files are reordered according to their

existing definitions.