Functional Overview

Only one function may be executed during a given execution of ADAORD.

Notes:

1. The format of the sequential dataset produced by the RESTRUCTURE functions is independent of the database device type, and is *not* compatible with the format required by the ADALOD or ADASAV utilities. Therefore, the target database may be contained on a device type different from the source database.
2. The Associator and Data Storage are reordered as part of RESTRUCTURE/STORE processing.
3. Parts of the database are overwritten during ADAORD execution. It is therefore recommended that the database (or file) be backed up with the ADASAV utility before running ADAORD functions.
4. The REORDATA, REORDB, REORFDATA and STORE functions do not reorder ADAM files. However, these functions can be used to relocate an ADAM file to different RABNs.
5. All ADAORD functions except RESTRUCTUREF (file) require exclusive EXF control of the database files involved in the operation. RESTRUCTUREF requires EXU control; other users may access database files being used by RESTRUCTUREF, but only for reading. Note, however, that operations involving either the checkpoint or security files require exclusive database control.
6. If the specified file was originally loaded with ISNREUSE=YES active, a STORE function and all ADAORD reorder functions that affect the file’s Associator will reset the "first unused ISN" value in that file’s control block (FCB) to the actual first unused ISN found in the address converter.
7. When specifying the starting RABN for Associator extents, the space needed for the FCBs, FDTs, and DSST should also be considered.
8. When RESTRUCTUREDB/F restructures an ADAM file that uses the overflow area, and then STORE stores the restructured file in a database with a smaller DATA block size, an ADAORD ERROR-103 may occur. Use the ADAULD/LOD utilities to move ADAM files, instead.

Reorder Functions

The REORASSO function physically reorders all Associator blocks for all files; the REORFASSO function reorders the Associator for a single file. This eliminates Associator space fragmentation and combines multiple address converter, normal and upper index, and Data Storage space table (DSST) component extents into a single logical extent for each component.

The REORDATA function reorders Data Storage for all files in the database; the REORFDATA function reorders Data Storage for a single file. This condenses extents containing only empty blocks, and also eliminates any Data Storage fragmentation caused by file deletion.

The REORDB function performs both the REORASSO and REORDATA functions in a single execution of ADAORD.

The REORFILE function performs both the REORFASSO and REORFDATA functions in a single execution of ADAORD. The records may be reordered in the logical sequence by a descriptor, by ISN, or in the current sequence.

Restructure Functions

The RESTRUCTUREDB function unloads an entire database to a sequential dataset; the RESTRUCTUREF function unloads one or more files to a sequential dataset. This dataset can be used as input to the STORE function.

The RESTRUCTURE functions are used to relocate the database to a different physical device or a file or files to another device.

Store Function

The STORE function loads one or more files into an existing database using the DDFILEA output created by the RESTRUCTUREDB, RESTRUCTUREF, or REORDB function.

Space Allocation

ADAORD allocates the amount of space required by the xxSIZE or MAXISN parameters, if specified. Otherwise, ADAORD allocates space based on the current size of the file. Note that the xxRELEASE parameters affect the amount of space required.

If possible, space is allocated on the volume specified by the xxxxVOLUME parameter. If insufficient free space is available on the specified volume, ADAORD allocates the remainder of the required space on other volumes, according to its default rules of allocation.

An xxRABN parameter overrides the associated xxxxVOLUME parameter.