

# ADASAV SAVE: Save Database

The SAVE database function saves all blocks of the database that are in use to create a full save dataset. If a Delta Save (DLOG) area is defined at the time of the save, the full save dataset can be specified as input for subsequent MERGE or RESTORE DELTA functions. In any case, the dataset can be specified as input for RESTORE or RESTONL functions.

The SAVE database function enables Delta Save logging at the end of the save if a DLOG area is defined in the database and the ADARUN parameter DSF=YES is specified. The next save operation may then be SAVE DELTA.

SAVE database may be executed with the Adabas nucleus active or inactive.

- If the Adabas nucleus is *inactive*, it cannot be started while the SAVE database function is executing, and no utility that makes changes to the database (e.g. ADALOD) can be run during this time. SAVE database cannot be executed offline if a nucleus session autorestart is pending, or if another offline utility (ADALOD or ADASAV) is currently running.
- If the Adabas nucleus is *active*, users have full access to the database. They can perform read, find, update, insert, and delete commands. However, utilities that make changes to the database (ADALOD, ADAINV, ADADBS DELETE, etc.) may not be running and cannot be started while the SAVE database function is performed. An online save operation is also not possible if the nucleus is running without protection logging.

If the nucleus is active during the SAVE database function, an ET-synchronization is performed at the end of the save operation to bring all user transactions to ET status. During ET synchronization, transactions already begun are allowed to continue while the start of new transactions is delayed until the end of the ET synchronization. The maximum time required for this synchronization can be limited by the TTSYN parameter.

For an online SAVE database function using the Delta Save Facility, a DSIM dataset must be supplied with DD name/link name DD/DSIMR1. This dataset receives all database blocks changed by the nucleus during the execution of SAVE database. The DSIM dataset must be supplied together with the created online save dataset for a MERGE or RESTORE DELTA function. Until the DSIM dataset has been specified for a subsequent MERGE operation, it cannot be reused for another online SAVE or SAVE DELTA operation (unless it is specifically reset by the ADAFRM DSIMRESET function).

If the execution of the SAVE database function is interrupted, it can be restarted using procedures outlined in the section Restarting an Interrupted Save Operation.

This chapter covers the following topics:

- Syntax
  - Optional Parameters
  - Example
-

## Syntax

```

ADASAV SAVE  [ BUFNO = number-of-buffers | 1 ]
              [ DRIVES = count | 1 ]
              [ DSIMDEV = device-type | ADARUN-device ]
              [ NOUSERABEND ]
              [ PERDRIVE = disks-per-tape ]
              [ TEST ]
              [ TTSYN = seconds | ADARUN-tt ]
              [ TWOCOPIES ]

```

## Optional Parameters

### BUFNO: Count of Buffers Per Drive

The BUFNO value, multiplied by the DRIVES parameter value, allocates fixed buffers for the SAVE operation. A value of 2 or 3 usually provides optimum performance; a value up to 255 is possible. A value greater than 5, however, provides little advantage and allocates a lot of space. The default is 1 (one buffer per drive).

See also the DRIVES parameter .

### DRIVES: Tape Drives for Parallel Save Processing

DRIVES is the number of tape drives to be used for parallel SAVE operations. A maximum of eight drives may be specified. The default is 1.

### DSIMDEV: DSIM Device Type

The DSIMDEV parameter specifies the device type of the DSIM dataset. This parameter is required only if the DSIM device type is different from that specified by the ADARUN DEVICE parameter (which is the default).

### 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.

### PERDRIVE: Disk Drives Per Tape Drive

PERDRIVE specifies the number of disk drives to be assigned to a single output tape drive. For example, if the database is contained on seven disk drives and three tape drives are available for SAVE processing, PERDRIVE=3,2,2 would cause the first three disk drives to be written to tape drive 1, the next two disk drives to be written to tape drive 2, and the next two disk drives to be written to tape drive 3. The drive sequence corresponds to the DDSAVEn/ DDDUALn or SAVEn/ DUALn job control specifications, as described in the section JCL/JCS Requirements and Examples.

The total number of drives specified by PERDRIVE must equal the sum of all Associator (ASSO) and Data Storage (DATA) disks; if both ASSO and DATA are on a single disk, this counts as two separate disks. If the DRIVES parameter is used and the PERDRIVE parameter is omitted, ADASAV determines the most efficient utilization of the tape drives.

### **TEST: Test Syntax**

The TEST 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.

### **TTSYN: SYN2 Checkpoint Control**

TTSYN allows the user to decrease the nucleus' ADARUN TT (maximum transaction time) during the synchronized checkpoint processing of the current ADASAV operation. The value specified is the approximate time in seconds (TT + 1.05 seconds), and must be less than the current ADARUN TT value. If TTSYN is not specified or if TTSYN is greater than the nucleus' TT value, that TT value becomes the default.

If the Adabas nucleus is active while ADASAV SAVE is running, a synchronized SYN2 checkpoint is taken at the end of the SAVE operation. This ensures that there is a point in time where all users are at ET status. If a user is not at ET status, no new transactions can be started for other users; they must wait until the SYN2 checkpoint can be taken.

The ADARUN TT value controls the maximum elapsed time permitted for a logical transaction. This is the maximum wait time until the SYN2 checkpoint can be processed. The ADASAV SAVE TTSYN parameter allows the user to decrease the TT value only during the synchronized checkpoint processing. The original TT value becomes effective again when ADASAV ends the SAVE operation.

### **TWOCOPIES: Create Two Copies of Output**

TWOCOPIES creates two physical copies of the ADASAV output.

## **Example**

```
ADASAV    SAVE
ADASAV    DRIVES=4
ADASAV    TTSYN=10
```

The complete database is to be saved using four tape drives in parallel. If running online, the ET synchronization at the end of the save operation should last at most 10 seconds.