RESTORE GCB: Restore Database Incremental from Offline Source

From a database SAVE data set created while the Adabas nucleus was *inactive,* the RESTORE GCB function restores

- the general control blocks (GCBs);

- Associator RABNs 3-30 of the database;

- the checkpoint file;

- the security file (if present); and

- all files specified with the FILES parameter.

Notes:

1. An interrupted RESTORE GCB operation must be reexecuted from the beginning. Until successful completion or reexecution of the restore operation, the database is inaccessible.
2. If the ADASAV RESTORE GCB job control contains the DD names, symbolic names, or link names for DDWORKnn/ WORKnn, these data sets are reset.

This chapter covers the following topics:

- Conditions

- Result

- Syntax

- Optional Parameters

- Examples

Conditions

▶ **To use the RESTORE GCB function, the following conditions must be met:**

1. The correct SAVE data set must be supplied. It must have been created by an offline database SAVE operation with the same version of Adabas as is used for the RESTORE and must contain the file(s) to be restored.

2. The output database must have the same physical layout (device types, extent sizes) as the original database. The Associator and Data Storage data sets must be present and must have been previously formatted. The SAVE data set to be restored may have originated for this or from a different database.

3. No Adabas nucleus may be active on the output database or on a database with the DBID of the output database.

4. If the SAVE operation was performed with the DRIVES parameter, the SAVE data sets created can also be restored with the DRIVES parameter. In that case, the restore operation is performed from the different SAVE data sets in parallel. Alternatively, the SAVE data sets can be concatenated to a single SAVE data set for a restore operation without the DRIVES parameter.

5. For restoring just a few files from a multivolume database SAVE data set, only those tape volumes that actually contain data of the files to be restored need to be supplied in the ADASAV job control. The job protocol of the SAVE operation as well as the corresponding SYNV checkpoints indicate the files or parts of files contained on each volume.

Result

The result of this function is a database containing the specified files and the checkpoint and security files with the same physical status they had at the time of the ADASAV SAVE operation.

This operation is equivalent to a RESTORE (database), but excludes any files not specified in the FILES parameter.

Important:
Any existing database in the target Associator and Data Storage data sets is completely overwritten and any files in that database are lost.

Syntax

```
ADASAV RESTORE GCB   [BUFNO = { number-of-buffers | 1 } ]
                     [CLOGDEV = CLOG1-device-type ]
                     [DRIVES = { count | 1 } ]
                     [EXCLUDE = file-list ]
                     [FILES = file-list ]
                     [NEWDBID = new-database-id ]
                     [NEWDBNAME = new-database-name ]
                     [NOUSERABEND]
                     [OVERWRITE]
                     [PLOGDEV = PLOG-device-type ]
                     [READONLY = ' ro-file-list ' ]
                     [RPLACTIVE = ' inactive-flag-file-list ' ]
                     [RPLDATA = ' restore-data-to-be-sent-file-list ' ]
                     [RPLDSBI = ' before-image-file-list ' ]
                     [RPLKEY = ' primary-key-file-list ' ]
                     [RPLTARGETID = ' target-ID-file-list ' ]
                     [RPLUPDATEONLY = ' upd-only-file-list ' ]
                     [TEST]
```

Optional Parameters

BUFNO: Count of Buffers Per Drive

The BUFNO value, multiplied by the DRIVES parameter value, allocates fixed buffers for RESTORE operation. A value of 2 or 3 usually provides optimum performance; 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).

CLOGDEV: Command Log Device Type

The device type of the command log (CLOG). This parameter is required only if the device type of the CLOG is different from that specified by the ADARUN DEVICE parameter.

DRIVES: Tape Drives for Parallel Restore

DRIVES is the number of tape drives to be used for parallel restore processing. The number can range 1 to 8, inclusively; the default is 1.

EXCLUDE: Exclude Specified Files from Restore

EXCLUDE lists the numbers of the files to be excluded from the restore operation; that is, the files that are not to be restored. This list can include a list of more than one Adabas file number or a range of file numbers. Ranges of file numbers should be specified using a dash (-) in the format: *fnfirst-fnlast.*

No files specified in the EXCLUDE parameter will exist in the restored database.

All files specified in the EXCLUDE parameter must exist on the save data se (if they are not included in a range of files).

The parameter is optional: if not specified, no files are excluded. A file number may be listed only once individually or in a range.

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

FILES: Files to Be Restored

FILES specifies one or more Adabas file numbers or a range of file numbers to be included in the database restore operation. Ranges of file numbers should be specified using a dash (-) in the format: *fnfirst-fnlast.*

If the specified file is a component file of an Adabas expanded file, all other component files of the expanded file must also be specified here. If a specified file is coupled to other files, the coupled files must also be specified. The checkpoint and security files are always restored.

NEWDBID: New ID for Restored Database

NEWDBID may be used to assign a different database ID to the restored database. The ID can be in the range 1-65,535; if Adabas Online System Security is installed, DBID 999 is reserved.

If NEWDBID is specified, the ADARUN DBID parameter must specify the ID of the database on the SAVE data set.

No Adabas nucleus may be active with the DBID specified on NEWDBID.

NEWDBNAME: New Database Name

NEWDBNAME assigns a new name to the restored database. If NEWDBNAME is not specified, the restored database keeps its old name.

NOUSERABEND: Termination without Abend

When a parameter error or a functional error occurs while this utility function is running, the utility ordinarily 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.

Note:
When NOUSERABEND is specified, we recommend that it be specified as the first parameter of the utility function (before all other parameters). This is necessary to ensure that its parameter error processing occurs properly.

OVERWRITE: Overwrite Existing Database

If the restore operation is to overwrite an existing database, the OVERWRITE parameter must be specified.

No Adabas nucleus may be active on the database to be overwritten.

PLOGDEV: Protection Log Device Type

The device type of the dual/multiple protection log (PLOG). This parameter is required only if the device type of the PLOG is different from that specified by the ADARUN DEVICE parameter.

READONLY: Read-only Status Indicator

READONLY indicates whether the read-only status is on or off for a file or a list of files. Valid values for this parameter are "YES" (read-only status is on) and "NO" (read-only status is off).

When restoring more than one file, the read-only status must be specified as positional values corresponding to the file number positions in the FILES list.

If READONLY is not specified, the read-only status of the file will be the same as it was on the SAVE data set.

Note:
The READONLY parameter is not allowed if a range of files is specified in the FILES parameter.

RPLACTIVE: Reset the Replication Inactive Flag in the FCB

RPLACTIVE is an optional parameter that specifies the inactive flag setting for a file during restore processing. Valid inactive flag settings are "YES", "NO", or no setting at all. A setting of "YES" turns *off* the replication inactive flag (YFSTQRPI) for a file. A setting of "NO" turns *on* the

replication inactive flag.

If no setting is specified, the default value is used. The default depends on the target database ID of the restore processing and its replication state. If *any of the following conditions* are met, the default is "YES"; otherwise the default is "NO":

- The original replication is turned off.

- The restore DBID is not the same as the original saved DBID.

- The original replication target ID has been changed.

- The original replication-before-image has been changed.

- The replication primary key has changed.

- The original replication is turned off.

When restoring more than one file, the correct RPLACTIVE settings must be specified as positional values corresponding to the file numbers' positions in the FILES list. For example, if four files are listed in the FILES file list, the following might be the setting for the RPLACTIVE parameter:

```
RPLACTIVE='YES,NO,,YES'
```

In this example, the inactive flag is turned off (YES) for the first and fourth files and turned on (NO) for the second file. No value is provided for the third file, so an default appropriate for the file is used.

RPLDATA: Send Data of the Restoring File to Replication Target ID

RPLDATA is an optional parameter that indicates whether the data in a file should be replicated to the replication target ID (RPLTARGETID parameter).

Valid replication settings are "YES", "NO", "CREATE", or no setting at all. A setting of "YES" causes the restore function to replicate the file data to the replication target during restore processing. A setting of "NO" will not replicate the data to the replication target during restore processing. A setting of CREATE causes the restore function to replicate the file data to the replication target during restore processing, but also sends a "create file" transaction to the replication target. If no setting is specified, the default "NO" is used.

Note:
Values of "YES" or "CREATE" can only be specified if replication is turned on for the corresponding file.

When restoring more than one file in the FILE file list, the RPLDATA settings must be specified as positional values corresponding to the file numbers' positions in the FILES list. For example, if four files are listed in the FILES file list, the following might be the setting for the RPLACTIVE parameter:

```
RPLDATA='YES,NO,,YES'
```

In this example, the data in the first and fourth files (YES) will be replicated to the replication target, but it will not be replicated for the second and third files (NO and no setting for the third file).

RPLDSBI: Replication Data Storage Before Image

RPLDSBI is an optional parameter that indicates whether the collection of before images of data storage should occur for an update command to a file.

Valid RPLDSBI settings are "YES", "NO", or no setting at all. A setting of "YES" indicates that the collection of before images of data storage will occur for the file during restore processing. A setting of "NO" indicates that the collection of before images of data storage will *not* occur for the file during restore processing.

If no setting is specified, the default value is used. The default depends on the target database ID of the restore processing and its replication state. If the restore DBID is the same as the originally saved DBID and REPLICATION=YES in the target DBID, the default is "YES"; otherwise the default is "NO".

Note:
A values of "YES" can only be specified if replication is turned on for the corresponding file.

When restoring more than one file in the FILE file list, the RPLDSBI settings must be specified as positional values corresponding to the file numbers' positions in the FILES list. For example, if four files are listed in the FILES file list, the following might be the setting for the RPLDSBI parameter:

```
RPLDSBI='YES,NO,,YES'
```

In this example, the before images are collected for the first and fourth files (YES), but are not collected for the second file (NO). No value is provided for the third file, so an default appropriate for the file is used..

RPLKEY: Primary Key for Replication

RPLKEY is an optional parameter that specifies the primary key for replication.

Valid RPLKEY settings are a two-character field name, "OFF", or no setting at all. Specifying a field name identifies that field as the primary key for replication. A setting of "OFF" indicates that no primary key should be used for replication.

If no setting is specified, the default value is used. The default depends on the target database ID of the restore processing and its replication state. If the restore DBID is the same as the originally saved DBID and REPLICATION=YES in the target DBID, the original RPLKEY value for the file is used; otherwise the default is "OFF".

Note:
A primary key can only be set if replication is turned on for the file and if the field name is a valid Adabas field according to the field definition table (FDT) for the file. When a new RPLKEY is specified it will not be confirmed as a valid Adabas field until the end of the ADASAV run. At that time, if any RPLKEY is found to be invalid, a warning message is issued, the RPLKEY is set to "OFF", and condition code 8 is returned.

When restoring more than one file in the FILE file list, the RPLKEY settings must be specified as positional values corresponding to the file numbers' positions in the FILES list. For example, if four files are listed in the FILES file list, the following might be the setting for the RPLKEY parameter:

```
RPLKEY='AA,BB,,OFF'
```

In this example, field AA is used as the replication primary key for the first file, BB is used as the replication primary key for the second file, and no replication primary key is used for the fourth file (OFF). No value is provided for the third file, so an default appropriate for the file is used..

RPLTARGETID: Replication Target ID

RPLTARGETID is an optional parameter that specifies the target ID of the Event Replicator Server to which the restored transactions should be sent.

Valid RPLTARGETID settings are a valid target ID, "OFF", or no setting at all. Specifying a target ID identifies that as the target for replication. A setting of "OFF" or "0" indicates that no replication target should be used for replication.

If no setting is specified, the default value is used. The default depends on the target database ID of the restore processing and its replication state. If the restore DBID is the same as the originally saved DBID and REPLICATION=YES in the target DBID, the original RPLTARGETID value for the file is used; otherwise the default is "OFF".

When restoring more than one file in the FILE file list, the RPLTARGETID settings must be specified as positional values corresponding to the file numbers' positions in the FILES list. For example, if four files are listed in the FILES file list, the following might be the setting for the RPLTARGETID parameter:

```
RPLTARGETID='23,24,,OFF'
```

In this example, target ID 23 is used as the replication target for the first file, 24 is used as the replication target for the second file, and no replication target is used for the fourth file (OFF). No value is provided for the third file, so an default appropriate for the file is used..

RPLUPDATEONLY: Allow Only Event Replicator Processing Updates

The RPLUPDATEONLY parameter can be used in the ADASAV RESTORE function to indicate whether an Adabas database file may be updated only by the Event Replicator Server as part of Adabas-to-Adabas replication or by other means as well. This parameter is optional.

Valid values are "YES" or "NO". A value of "YES" indicates that the file can only be updated via Event Replicator processing; a value of NO indicates that the file can be updated by any normal means, including Event Replicator processing.

If no value is specified, the default RPLUPDATEONLY setting of the file at the time of the corresponding SAVE operation is used.

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.

Examples

Example 1:

```
ADASAV RESTORE GCB,FILES=2,4,6,8
```

The database Associator GCBs, RABNs 3-30, the checkpoint and security files, and files 2,4,6, and 8 are to be restored.

Example 2:

```
ADASAV RESTORE GCB,FILES=3,4,5,OVERWRITE
```

The Associator's GCBs and files 3, 4, and 5 are to be restored; the existing database will be overwritten.