

REVERT: Revert Database to Lower Version

The REVERT function starts from the Adabas version of the last nucleus session.

```
ADACNV REVERT [TOVERS = target-version ]  
              [IGNPPT]  
              [NOUSERABEND]  
              [PLOGDEV = { multiple-PLOG-device-type | ADARUN-device }]  
              [RESTART]  
              [TEST]
```

This chapter covers the following topics:

- Essential Parameter and Subparameter
 - Optional Parameter
 - Reversion Considerations
 - Example
-

Essential Parameter and Subparameter

TOVERS: Target Version

The version of Adabas database (version and revision level) to achieve at the end of the ADACNV run. The TOVERS parameter value must be a version lower than the source version.

The version format is *vr* indicating the version and revision level; for example, *61*.

Optional Parameter

IGNPPT: Ignore Parallel Participant Table PLOG Entries

When reverting from a version of Adabas that uses the parallel participant table (PPT) structure to a lower version of Adabas, an error is printed and conversion fails if the system detects one or more protection logs (PLOGs) from the current version that have not been copied/merged.

If IGNPPT is specified, the utility will continue processing in spite of the uncopied/unmerged PLOGs.

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.

PLOGDEV: Multiple PLOG Device Type

PLOGDEV specifies the physical device type on which the multiple protection log data sets to be reverted is contained. If PLOGDEV is not specified, the device type specified by the ADARUN DEVICE parameter is used.

RESTART: Rerun after Point of No Return

If ADACNV terminates abnormally after the *point-of-no-return*, that is, after all changed blocks have been written to DD/FILEA, the RESTART parameter instructs ADACNV to begin its run by reading the contents of DD/FILEA and continue by writing them to the database.

TEST: Test Conversion

The TEST parameter tests the feasibility of the reversion operation without actually writing any changes to the database.

Reversion Considerations

The following is an overview of the reversion steps performed by ADACNV.

All Versions

- Reversion is not possible if any Adabas feature is used in the current version that is not supported in the target version. This statement applies to all Adabas features that affect the structure of the database.

From Version 8 to Any Prior Version

If a database makes use of any of the following extended features of Adabas 8, ADACNV will not allow you to revert the database to a version prior to Adabas 8:

- More than five ASSO, DATA, or DSST extents
- More than five file extents
- Files that allow spanned records
- Files that allow more than 191 MU and PE occurrences
- Files that make use of large object (LB) fields
- Files with fields that have the NB (no blank compression) option
- System files with two-byte file numbers.

If you want to complete the backward conversion, you must first remove any file with these new features from the Adabas database.

The use of the following other new features provided in Adabas 8 do *not* prevent backward conversion to Adabas 7.4, but, of course, the new features cannot be used in Adabas 7.4:

- Adabas commands issued via the ACBX interface (for example, with long or segmented buffers)
- Commands using the new format buffer features (for example, the length indicator).

From Version 8 to Version 7.1

You cannot use REVERT to revert a database directly from Adabas 8 to Adabas 7.1; instead, you must first revert the database to Adabas 7.4 and then from 7.4 to 7.1.

From Version 7.1 to 6.2

- Version 7.1 extends the free space table (FST) from one RABN (RABN 10) to five RABNs (RABNs 10-14). ADACNV checks whether all FST entries fit into one RABN. If not, the smallest FST extent is removed. This is repeated until the FST fits into one ASSO block. An appropriate message is printed.
- Any Adabas Delta Save DLOG area header is set to the correct version.

From Version 6.2 to 6.1

- Any Adabas Delta Save DLOG area header is set to the correct version.

Example

```
ADACNV REVERT TOVERS=73
```

The Adabas version of the last run of the nucleus is to be converted back (reverted) to a version 7.3 Adabas database.