Functional Overview

ADADCK checks Data Storage and the Data Storage space table (DSST) of a specific file (or files) in the database.

ADADCK reads each used Data Storage block (according to the Data Storage extents in the file control block) and performs the following checks:

- Is the block length within the permitted range? (4 block length physical block size)

- Is the sum of the length of all records in the Data Storage block plus 4 equal the block length?

- Is there any record with a record length greater than the maximum compressed record length for the file or with a length 0?

- Are there any duplicate ISNs within one block?

- If spanned records are used, are the ISNs in the header valid? Does each header contain the ISN of the primary record in the chain and the ISN of the next spanned record in the chain?

- If spanned records are used, are the primary and secondary spanned records correctly identified?

- Does the associated DSST element contain the correct value? If not, a REPAIR of the DSST is necessary (see REPAIR parameter).

Notes:

1. ADADCK does not require the Adabas nucleus to be active.
2. If the nucleus is active, ADADCK synchronizes its operation with the active nucleus unless the NOOPEN parameter is specified.
3. Any pending autorestart condition is ignored.
4. This utility should be used only for diagnostic purposes.

ADADCK returns a condition code 4 or 8 if an error occurs.

ADADCK and Spanned Records

If spanned records are used, the entire spanned record chain is checked for accuracy if any Data Storage RABN specified is part of the chain. If the FROMRABN and TORABN are specified and the file contains spanned Data Storage records, we recommend that the FROMRABN parameter point to the RABN of a primary record. Otherwise, a warning message may result. The ADADCK run will still result in a condition code of zero; the warning just indicates that it is not possible to check the entire chain because the FROMRABN that was specified was not for a primary record, but for a secondary record.

It is also possible that the secondary ISNs reside in a data RABN other than what was specified in the FROMRABN and TORABN parameter specifications. Therefore, additional Data Storage RABNs may need to be read to validate the secondary spanned record chain. ADADCK builds a secondary ISN table on the fly in an effort to optimize performance. If any secondary ISNs are noted missing, the entire data storage will be searched to attempt to locate the missing secondary ISNs.

The maximum number of ISNs that will be checked for a spanned Data Storage file is determined by the setting of the MAXPISN parameter, which defaults to 1000. If the file contains more than primary ISNs than the MAXPISN parameter setting, execution will continue but the following warning message will be displayed.

```
*** Warning ***
          More than MAXPISN primary spanned ISNs.  Only
          the first MAXPISN ISNs will be checked.  Run
          ADADCK again specifying a different
          FROMRABN to check the remaining RABNs or
          specify a higher MAXPISN value.
          Any errors reported after this warning
          may be due to the table limitation.
```

If this warning message appears, Any errors or CC=8 conditions that are reported after this warning may be due to the fact that the utility was unable to track the ISNs due to the size limitation of 1000. The file may be intact and the error may be because the internal table used for ADADCK processing could not hold all of the ISNs. To be sure that the file is fine, run the ADAVAL utility on it.