NEWFIELD: Add New Field

The NEWFIELD function adds one or more fields to a file. The new field definition is added to the end of the field definition table (FDT).

Note:
Although the definition of a descriptor field is independent of the record structure, note that if a descriptor field is not ordered first in a record and logically falls past the end of the physical record, the inverted list entry for that record is not generated for performance reasons. To generate the inverted list entry in this case, it is necessary to unload short, decompress, and reload the file; or use an application program to order the field first for each record of the file.

NEWFIELD cannot be used to specify actual Data Storage data for the new field; the data can be specified later using Adabas add/update or Natural commands.

When adding a field to an Adabas expanded file, the field must be added to *each individual component file* . Each NEWFIELD operation on a component file returns a message that confirms the change and condition code 4.

This utility function does not need to lock the database file for its use; this function can perform its processing in parallel with active users. This means that you do not need to set the file to read-only status to run this utility function.

```
ADADBS  NEWFIELD  FILE = file-number
                  [FNDEF = 'Adabas-field-definition' ]
                  [NOUSERABEND]
                  [PASSWORD = 'password' ]
                  [SUBFN = 'name  = parent-field ( begin , end )' ]
                  [SUPFN = 'name  = { parent-field ( begin, end ) } ...' ]
                  [TEST]
```

This chapter covers the following topics:

- Essential Parameter

- Optional Parameters

- Example

Essential Parameter

FILE: File Number

> FILE specifies the file in which the field to be added is contained. The file may not be an Adabas system file.

Optional Parameters

FNDEF: Adabas Field Definition

FNDEF specifies an Adabas field (data) definition. One FNDEF statement is required for each field to be added. The syntax used in constructing field definition entries is:

FNDEF = '*level, name* [, *length, format*] [, **MU** [(*occurrences*)]] [, *option*] ... '

FNDEF = '*level, name* [, **PE** [(*occurrences*)]]'

Each definition must adhere to the field definition syntax as described for the ADACMP utility in *FNDEF: Field and Group Definition* and *FNDEF: Periodic Group Definition* in the section entitled *Field Definition Statements*, in the ADACMP documentation.

Note the following restrictions:

- A subdescriptor, superdescriptor, hyperdescriptor, or phonetic descriptor definition cannot be specified.

- Text information or sequence numbers are not permitted.

- If you specify an occurrence number when adding an MU or PE field, it is ignored.

The following rules apply when you set the level number in the first FNDEF statement:

1. A level number 01 is always allowed.

2. A level number of 02 or higher means that this field is to be added to an existing group. If so, the following rules apply:

   - The field can be added if the group is a normal (not periodic) group;

   - If the group is a PE group, the field can be added only if the file control block (FCB) for the file does *not* exist; that is, either the file was deleted with the KEEPFDT option, or the FDT was defined using the Adabas Online System Define FDT function, but the Define File function has not yet been run.

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). When NOUSERABEND is specified, it must be specified as the first parameter (before all other parameters) for the utility function.

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.

PASSWORD: File Password

File password. This parameter is required if the file is password-protected.

SUBFN/ SUPFN: Add Subfields or Superfields

These parameters may be used to add subfields and superfields. Each definition must adhere to the definition syntax for sub/superfields as described for the ADACMP utility. Read *COMPRESS: Compress an Adabas File* for more information.

TEST: Test Syntax

This 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. See Syntax Checking with the TEST Parameter for more information on using the TEST parameter in ADADBS functions.

Example

Group AB (consisting of fields AC and AX) is to be added to file 24.

```
ADADBS    NEWFIELD    FILE=24
ADADBS                FNDEF='01,AB'
ADADBS                FNDEF='02,AC,3,A,DE,NU'
ADADBS                FNDEF='02,AX,5,P,NU'
```