

TRGMAIN: An API To Maintain Triggers

The Adabas triggers and stored procedures facility provides a callable routine TRGMAIN to maintain trigger definitions from user-written programs. TRGMAIN functions include

- add new triggers
- modify existing triggers
- purge existing triggers
- display a specific trigger definition
- activate triggers
- deactivate triggers

To call the trigger maintenance routine

- Enter the following:

```
CALL 'TRGMAIN' function calling-parameters
```

where the functions and calling parameters are as described in the following sections.

This chapter covers the following topics:

- Functions (Format A5)
- Calling Parameters (Format A209)
- Sample User Program
- Response Codes

Functions (Format A5)

The FUNC parameter specifies what action is to be performed on the trigger definition. Valid values are

ADD	Add
MOD	Modify
DEL	Purge
DISP	Display
ACT	Activate
DEACT	Deactivate

Calling Parameters (Format A209)

The parameters that compose CALLING-PARMS provides the information needed to complete the actions. These parameters are consistent with the information requested by the online Trigger Maintenance facility and include the following fields (default values are underlined):

Field Name	Length	Description
TRG-DBNR	N5	Trigger file DBnr
FILE-NAME	A32	File name defined in the FFT
CMD-TYPE	A1	Command type that causes trigger to fire: D delete F find I insert R read U update * all commands
FIELD-NAME	A32	Field name defined in the FFT
TRIGGER-PROG	A8	Name of the Natural trigger program
PRIORITY	N3	Trigger priority
PRE-CMD-SELECT	A1	Trigger timing relative to the command execution: N post-command trigger Y pre-command trigger
PART-NON-FLAG	A1	Participation flag. Valid values are <u>A</u> asynchronous (default) N nonparticipating P participating

Field Name	Length	Description
CALLNAT-FLAG	A1	CALLNAT parameters. Valid values are <u>C</u> control parameters with ACB and response code (default) E response code only N no parameters passed X control parameters with ACBX and response code
RECBUFF-OPTION	A1	Record buffer access. Valid values are A access only <u>N</u> no access to record buffer (default) U updates allowed
ACT-TYPE	A4	Type of activation/deactivation to be processed: PERM permanent TEMP temporary
TRG-RSV	A39	Reserved for future use
RESP	N4	Response code returned from the API
RETURN-MSG	A68	Text description of the response code
VERSION	N3	GCB version
PRODVERN	N3	Database version
NUCID	P5	Nucleus ID

Sample User Program

```

*****
Program      UMAINT      Library SYSTRG
0010 *****
0020 *  Application .. Trigger Maintenance
0030 *  Program ..... UMAINT
0040 *  Function ..... Sample call program to call API TRGMAIN that
0050 *                   maintains trigger definitions.
0060 *                   (Add,Delete,Modify,Display,Activate,Deactivate).
0070 *
0080 *  Parameters ... The following parameters are passed when calling
0090 *                   the API:
0100 *      FUNC          (A5) Action to perform on Trigger Definition

```

```

0110 *           Valid values are ADD - Add
0120 *           MOD - Modify
0130 *           DEL - Delete
0140 *           DISP - Display
0150 *           ACT - Activate
0160 *           DEACT - Deactivate
0170 *   TRG-DBNR      (N5)  Trigger file DBnr
0180 *   FILE-NAME    (A32)  File name defined in the FFT
0190 *   CMD-TYPE     (A1)  Command type that causes trigger to fire
0200 *           Valid values are R - Read
0210 *           F - Find
0220 *           I - Insert
0230 *           U - Update
0240 *           D - Delete
0250 *           * - All commands
0260 *   FIELD-NAME   (A32)  Field name defined in the FFT
0270 *   TRIGGER-PROG (A8)  Name of the Natural trigger program
0280 *   PRIORITY     (N3)  Trigger priority
0290 *   PRE-CMD-SELECT (A1) Pre-trigger or post-trigger
0300 *           Valid values are Y - Pre trigger
0310 *           N - Post trigger
0320 *   PART-NON-FLAG (A1)  Participation flag
0330 *           Valid values are A - Asynchronous
0340 *           N - Non-participating
0350 *           P - Participating
0360 *   CALLNAT-FLAG (A1)  CALLNAT parameters
0370 *           Valid values are N - No parameters passed
0380 *           E - Response code only
0390 *           C - Control parms and
0400 *                   response code
0410 *   RECBUFF-OPTION (A1) Record buffer access
0420 *           Valid values are A - Access only
0430 *           U - Updates allowed
0440 *           N - No access to RB
0450 *   ACT-TYPE     (A4)  Type of activation/deactivation
0460 *           Valid values are: TEMP - Temporary
0470 *                   PERM - Permanent
0480 *   RESP         (N4)  Response code returned from the API
0490 *   RETURN-MSG   (A68) Text description of the response code
0500 *****
0510 DEFINE DATA LOCAL
0520 01 FUNC (A5)
0530 01 CALLING-PARMS (A161)
0540 01 REDEFINE CALLING-PARMS
0550 02 TRG-DBNR      (N5)
0560 02 FILE-NAME    (A32)
0570 02 CMD-TYPE     (A1)
0580 02 FIELD-NAME   (A32)
0590 02 TRIGGER-PROGRAM (A8)
0600 02 PRIORITY     (N3)
0610 02 PRE-CMD-SELECT (A1)
0620 02 PART-NON-FLAG (A1)
0630 02 CALLNAT-FLAG (A1)
0640 02 RECBUFF-OPTION (A1)
0650 02 ACT-TYPE     (A4)
0660 02 RESP        (N4)
0670 02 RETURN-MSG   (A68)
0680 01 MAP-MSG (A68)
0690 01 HOLD-FUNC (A5)
0700 01 HOLD-PRE-CMD-SELECT (A1)
0710 01 PAGE-TITLE (A50)
0720 01 #ATTR (C)

```

```

0730 END-DEFINE
0740 RESET CALLING-PARMS MAP-MSG
0750 MOVE 233 TO TRG-DBNR
0760 SET KEY PF3
0770 **
0780 ** Request function and required fields
0790 **
0800 REPEAT
0810   INPUT (AD=TMIL'_' CD=NE)
0820   MAP-MSG (IP=OFF AD=O)
0830   / 9T 'API Maintenance of Trigger Definitions' (YEI)
0840   // 'Function.....' FUNC 30T '(Add, Del, Mod, Disp, or ".")'
0850   // 'File Name.....' FILE-NAME
0860   / 'Field Name.....' FIELD-NAME
0870   / 'Command Type.....' CMD-TYPE 30T '(R, F, I, U, D, or *)'
0880   / 'Pre-Command.....' PRE-CMD-SELECT 30T '(Y, N, or blank)'
0890   RESET MAP-MSG
0900 **
0910 ** Escape out of here
0920 **
0930   IF FUNC = MASK('.') OR *PF-KEY = 'PF3'
0940   ESCAPE BOTTOM
0950 ** Set up Page Titles
0960   DECIDE ON FIRST VALUE OF FUNC
0970   VALUE 'DISP' MOVE 'Display' TO PAGE-TITLE
0980   VALUE 'ADD' MOVE 'Add' TO PAGE-TITLE
0990   VALUE 'MOD' MOVE 'Modify' TO PAGE-TITLE
1000   VALUE 'DEL' MOVE 'Delete' TO PAGE-TITLE
1010   VALUE 'X' ESCAPE BOTTOM
1020   NONE REINPUT 'Invalid Function Code'
1030   END-DECIDE
1040   COMPRESS PAGE-TITLE 'Trigger Definition' INTO PAGE-TITLE
1050   IF FILE-NAME = ' '
1060   REINPUT 'File Name cannot be BLANK' MARK *FILE-NAME
1070 **
1080 ** Handle Request to Display a Trigger
1090 **
1100   IF FUNC = 'DISP'
1110   DO
1120     PERFORM GET-TRIGGER
1130     INPUT (AD=O CD=NE)
1140     MAP-MSG (IP=OFF AD=O)
1150     / 9T 'API Maintenance of Trigger Definitions' (YEI)
1160     // 10T PAGE-TITLE (IP=OFF)
1170     // 'File name.....' FILE-NAME
1180     / 'Field Name.....' FIELD-NAME
1190     / 'Command Type.....' CMD-TYPE
1200     // 'Trigger Program...' TRIGGER-PROGRAM
1210     / 'Priority.....' PRIORITY
1220     / 'Pre-Command.....' PRE-CMD-SELECT
1230     / 'Trigger Type.....' PART-NON-FLAG
1240     / 'CALLNAT Params....' CALLNAT-FLAG
1250     / 'RecBuffer Access..' RECBUFF-OPTION
1260     RESET FUNC
1270     MOVE RETURN-MSG TO MAP-MSG
1280   DOEND
1290 **
1300 ** Handle Request to Alter Trigger Definitions
1310 **
1320   IF FUNC = 'ADD' OR= 'MOD' OR= 'DEL'
1330   DO
1340     IF (FUNC = 'MOD' OR= 'DEL')

```

```

1350     PERFORM GET-TRIGGER
1360     IF FUNC = 'DEL'
1370     DO
1380         MOVE 'Press ENTER to Delete or PF-3 to Cancel' TO MAP-MSG
1390         MOVE (AD=P) TO #ATTR
1400     DOEND
1410     ELSE DO
1420         MOVE 'Press ENTER to confirm data or PF-3 to Cancel' TO MAP-MSG
1430         MOVE (AD=D CD=NE) TO #ATTR
1440     DOEND
1450     REPEAT
1460         INPUT (AD=TMIL'_' CD=NE)
1470         MAP-MSG (IP=OFF AD=O)
1480         / 9T 'API Maintenance of Trigger Definitions' (YEI)
1490         // 10T PAGE-TITLE (AD=O IP=OFF)
1500         // 'File Name.....' FILE-NAME (AD=O)
1510         / 'Field Name.....' FIELD-NAME (CV=#ATTR)
1520         / 'Command Type.....' CMD-TYPE (CV=#ATTR)
1530         // 'Trigger Program...' TRIGGER-PROGRAM (CV=#ATTR)
1540         / 'Priority.....' PRIORITY (CV=#ATTR)
1550         / 'Pre-Command.....' PRE-CMD-SELECT (CV=#ATTR)
1560         / 'Trigger Type.....' PART-NON-FLAG (CV=#ATTR)
1570         / 'CALLNAT Params....' CALLNAT-FLAG (CV=#ATTR)
1580         / 'RecBuffer Access..' RECBUFF-OPTION (CV=#ATTR)
1590     RESET MAP-MSG
1600     IF *PF-KEY = 'PF3'
1610     DO
1620         MOVE 'Function canceled per user request' TO MAP-MSG
1630         ESCAPE BOTTOM
1640     DOEND
1650 **
1660 ** Perform the update of data (add, del, or mod)
1670 ** and handle the response
1680 **
1690     CALLNAT 'TRGMAIN' FUNC CALLING-PARMS
1700     DECIDE ON FIRST VALUE OF RESP
1710     VALUE 0 MOVE RETURN-MSG TO MAP-MSG
1720         EXAMINE MAP-MSG FOR 'confirmed' REPLACE 'successful'
1730         ESCAPE BOTTOM
1740     VALUE 20 REINPUT WITH TEXT RETURN-MSG MARK *FIELD-NAME
1750     VALUE 23 REINPUT WITH TEXT RETURN-MSG MARK *FIELD-NAME
1760     VALUE 25 REINPUT WITH TEXT RETURN-MSG MARK *CMD-TYPE
1770 *     VALUE 37 REINPUT WITH TEXT RETURN-MSG MARK *PRIORITY
1780 *     VALUE 38 REINPUT WITH TEXT RETURN-MSG MARK *PRIORITY
1790     VALUE 39 REINPUT WITH TEXT RETURN-MSG MARK *TRIGGER-PROGRAM
1800     VALUE 40 REINPUT WITH TEXT RETURN-MSG MARK *PRE-CMD-SELECT
1810     VALUE 41 REINPUT WITH TEXT RETURN-MSG MARK *PART-NON-FLAG
1820     VALUE 42 REINPUT WITH TEXT RETURN-MSG MARK *CALLNAT-FLAG
1830     VALUE 43 REINPUT WITH TEXT RETURN-MSG MARK *RECBUFF-OPTION
1840     NONE MOVE RETURN-MSG TO MAP-MSG
1850     END-DECIDE
1860     LOOP
1870     RESET FUNC
1880     DOEND
1890 ** Loop back up to display the starting screen
1900 LOOP
1910 *****
1920 ** Subroutine to retrieve the *
1930 ** trigger information *
1940 *****
1950 DEFINE SUBROUTINE GET-TRIGGER
1960 MOVE FUNC TO HOLD-FUNC /* Go get the existing

```

```

1970 MOVE 'DISP' TO FUNC                                /* trigger information
1980 MOVE PRE-CMD-SELECT TO HOLD-PRE-CMD-SELECT
1990 CALLNAT 'TRGMAIN' FUNC CALLING-PARMS
2000 MOVE HOLD-FUNC TO FUNC
2010 IF RESP NE 0
2020 REINPUT WITH TEXT RETURN-MSG
2030 IF RESP = 0
2040 DO
2050   IF PRE-CMD-SELECT NE HOLD-PRE-CMD-SELECT
2060     AND HOLD-PRE-CMD-SELECT NE ' '
2070   DO
2080     MOVE HOLD-PRE-CMD-SELECT TO PRE-CMD-SELECT
2090     MOVE 'Trigger does not exist' TO MAP-MSG
2100     IF HOLD-PRE-CMD-SELECT = 'Y'
2110     COMPRESS 'Pre-' MAP-MSG INTO MAP-MSG
2120     ELSE COMPRESS 'Post-' MAP-MSG INTO MAP-MSG
2130     MOVE HOLD-FUNC TO FUNC
2140     ESCAPE TOP
2150   DOEND
2160   IF CMD-TYPE = ' '
2170   MOVE '*' TO CMD-TYPE
2180 DOEND
2190 RETURN
2200 *** End of Subroutine ***
2210 END
2220 **

```

Response Codes

Code	Meaning
000	Function completed successfully.
013	Invalid file-field entry specified. The triggers facility requires access to a file-field table that maps long file names to file numbers and field names to Adabas two-character field identifiers. See section <i>File-Field Tables</i> for more information.
016	No trigger definition found with this criteria.
020	Field name must be blank for delete command class.
023	Field name not found for this file. The field name specified does not exist in the file-field table for this file. See section <i>File-Field Tables</i> for more information.
025	Invalid command type. See list above for valid values.
027	No field found for this file and command type.
029	No trigger found for this file, command, and field criteria.
037	Priority must be between 1 and 900.
038	Priority cannot be set if Adabas field name is **.
039	Natural subprogram name is invalid.
040	Specify "Y" for pre-trigger and "N" for post-trigger.
041	Trigger type may be "A", "N", or "P" only.

Code	Meaning
042	CALLNAT type may be "C", "E", or "N" only.
043	Record buffer usage may be "A", "N", or "U" only.
044	Invalid parameter combination. No record buffer available with asynchronous triggers.
045	Invalid parameter combination. No record buffer available for pre-triggers for read or find commands.
046	Invalid parameter combination. No record buffer available for delete commands.
047	A trigger already exists for this parameter combination.
048	Invalid request. Trigger is already in the specified state.
052	Change in trigger state not possible now.
103	File name must not be blank.
111	Invalid function code.
112	TYPE must be temp or perm. The type of activation or deactivation of a trigger must be either temporary (lasting for this nucleus session only) or permanent.
1xxx	Function resulted in an Adabas response code 22 (ADARSP022) where xxx represents the subcode.
3xxx	Non-zero Adabas response code was returned, where xxx is the actual Adabas response code.
9999	Function not successful. Verify parameters and existing trigger definitions. Contact your Software AG technical support representative with questions.