ON DB_CHANGE

ON DB_CHANGE action

Declaration

```
ON DB_CHANGE tableIdent GOTO [procName [PRTY prtyIdent]|  NONE]
```

Parameters

tableIdent	in	Identifier of object of Table type.
procName	in	Name of RPC procedure in ESL script.
prtyIdent	in	Optional identifier of *Int* type. A priority when calling RPC.

Description

Action defines a procedure (by the name *procName*) which will be called when the values in database table is changed. This change is identified by process D2000 DbManager that is the only one process executing a database operations in D2000 system. If it identifies some change it will generate automatically a command for calling the registered RPC procedure.

The key word **PRTY** enables to set the priority of executing registered RPC procedure. The priority is defined by the parameter *prtyIdent* after **PRTY**.

Action **ON DB_CHANGE tableIdent GOTO NONE** will cancel the calling of procedure when the table is being changed. One table can have (in context of ESL script instance) only one procedure for handle of its change. That is why the repeated registration of handle for one table will cause ESL run-time error and the script failed.

The declaration of the procedure *procName* must be following:

```
RPC PROCEDURE ProcName (IN INT _dbTable, IN INT _actionId)
END ProcName
```

RPCX can be used instead of RPC.

The value of formal parameter *_actionId* can be one of the following options:

_actionId	0	Monitoring of the table has started successfully.
_actionId	1	Initialization of table monitoring failed because of one of the following reasons: • the process **D2000 DbManager** does not run, • the identifier *tableIdent* is not the identifier of object of Table type, • the monitoring of table was interrupted unexpectedly, e.g. **D2000 DbManager** failed.
_actionId	10	**D2000 DbManager** is executing the action DB_REFRESH_TABLE or a structure definition configured on object of Table type has been saved.
_actionId	11	**D2000 DbManager** has finished the operation DB_*, DBS_* PG_* or some operation has been executed over the table from process D2000 HI by user interface. If the modification are executed within the database transaction (DB_TRANS_OPEN ...), **D2000 DbManager** generates a message about the change of table when the changes are being confirmed (DB_TRANS_COMMIT).
_actionId	12	The table has been deleted from configuration of application.

_actionId	13	The active instance of the process **D2000 DbManager** changed its status to passive or in [redundancy](link) the D2000 Server to which the process **D2000 DbManager** is connected changed its status from *HOT* to *SBS*. It is necessary to re-register at new active instance of the process **D2000 DbManager** via action *ON DB_CHANGE*. **Note:** When switching the active instance or during D2000 redundancy switch (when the process **D2000 DbManager** becomes passive), all the transactions are closed, all database operations are terminated and all handles are invalidated. Therefore it is necessary to repeat all initialization sequences of scripts on new active instance (opening the databases, tables etc.).

ⓘ **Related pages:**

[Script actions](link)