

# ON DB\_CHANGE

## ON DB\_CHANGE action

Declaration

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

Parameters

tableIdent	in	Identifier of the object of the <a href="#">Table</a> type.
procName	in	Name of RPC procedure in ESL script.
prtyIdent	in	Optional identifier of the <i>Int</i> type. A <a href="#">priority</a> when calling <a href="#">RPC</a> .

Description

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

The keyword **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:

— a c t i o n I d	0	Monitoring of the table has started successfully.
— a c t i o n I d	1	Initialization of table monitoring failed because of one of the following reasons: <ul style="list-style-type: none"><li>the <b>D2000 DbManager</b> process does not run,</li><li>the identifier <i>tableIdent</i> is not the identifier of object of <a href="#">Table</a> type,</li><li>the monitoring of table was interrupted unexpectedly, e.g. <b>D2000 DbManager</b> failed.</li></ul>
— a c t i o n I d	10	<b>D2000 DbManager</b> is executing the action DB_REFRESH_TABLE or a structure definition configured on object of Table type has been saved.
— a c t i o n I d	11	<b>D2000 DbManager</b> 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 ...), <b>D2000 DbManager</b> generates a message about the change of table when the changes are being confirmed (DB_TRANS_COMMIT).
— a c t i o n I d	12	The table has been deleted from configuration of application.

_a ct io n ld	13 The active instance of the <b>D2000 DbManager</b> process changed its status to passive or in <a href="#">redundancy</a> the D2000 Server to which the <b>D2000 DbManager</b> process is connected changed its status from <i>HOT</i> to <i>SBS</i> . It is necessary to re-register at new active instance of the <b>D2000 DbManager</b> process via action <i>ON_DB_CHANGE</i> . <b>Note:</b> When switching the active instance or during D2000 redundancy switch (when the <b>D2000 DbManager</b> process becomes passive), all the transactions are closed, all database operations are terminated and all handles are invalidated. Therefore it is necessary to repeat all of the initialization sequences of scripts on a new active instance (opening the databases, tables etc.).
---------------------------	--



#### Related pages:

[Script actions](#)