UPDATEARCHVAL

UPDATEARCHVAL action

Function

Modification, or storing of a value into the archive.

Declaration

```
UPDATEARCHVAL valueIdent, archIdent [,[retIdent_Int], bRecalcStat_Bool]
```

Parameters

valueIdent	in	Identifier of a value that is to be written into the archive.
archIdent	in	Reference to one of (historical) value - (not controlled whether the archive object is used - adding reference to object is enough).
retIdent_Int	out	Identifier of the *Int* type - return code: the action success (optional parameter).
bRecalcStat_Bool	in	Identifier of the *Bool* type - enable/disable the calculation of depending statistical historical values (optional parameter). Possible values: • @TRUE - enable calculation (default value) • @FALSE - disable calculation

Description

The action modifies (or store a new) a value into the for the archive object `archIdent`. The value is given by the identifier `valueIdent`. Time of its storing into the database is the time when the value `valueIdent` occurred. If the identifier retIdent_Int is not specified, the action doesn't wait for the storing confirmation. If the identifier is stated, gains one of the following values:

- _ERR_TRANS_ABORT
- _ERR_TRANS_ERROR
- _ERR_TRANS_IGNORED
- _ERR_NO_ERROR

Note

If the identifier retIdent_Int is not specified, there is no feedback to detect the action success. The action execution time is short, because it is only a request that is sent to the system.
If the identifier is specified, the script waits for the physical storing of a value into the archive database.

Consequence:

If I use the action to store a value into the archive without waiting and then I want to read this value, the read value almost certainly is not the one that was stored by previous action.

Calculation of depending statistical historical values can be disabled by setting the parameter *bRecalcStat_Bool* to the value of @FALSE.

Example

Storing a value into the archive:

```
REAL _archVal
TIME _archTime
INT _retCode
```

```
; archiving time
_archTime := %StrToTime("12:0:0 1-12-2000")
```

```
; prepare a value together with time
_archVal := 1 TIME _archTime
```

```
; storing into the archive
UPDATEARCHVAL _archVal, H.Int, _retCode
```

```
; action success test
IF _retCode # _ERR_NO_ERROR THEN
; error at storing
ENDIF
```

Related pages:

Script actions