

MXChange for Melsec Medoc plus 2.40

Additional Manual

Table of Contents

1	MXChange	4
1.1	Global tools	4
1.1.1	Super Project Manager	4
1.1.2	MXChange interface tag editor	4
1.2	MXChange update MM+ 2.40 <-> MXChange server	4
1.2.1	Moments of MXChange updates	4
1.2.2	The online flag.....	4
1.2.3	Transaction logging	4
1.2.4	Offline synchronisation	4
1.3	Import and export	5
1.3.1	ASCII import and export.....	5
1.3.2	Import of Profi-MAP-POU-blocks → MM+ 2.40.....	5
1.4	MM+ 2.40 objects.....	5
1.4.1	PLC information	5
1.4.1.1	Serial communication card	5
1.4.1.2	Ethernet communication card	5
1.4.1.3	MELSEC-NET communication card (serial master)	5
1.4.1.4	MELSEC-NET communication card (Ethernet master).....	6
1.4.2	MM+ 2.40 tag information.....	6
1.4.3	Mapping from ProfiMap I/O references.....	6
1.5	Network communication.....	6
1.6	MXChange tag types	7
1.6.1	PLC parameters and online settings	7
1.6.2	Global variables	7
1.6.3	POU blocks of MM+ 2.40	7
1.6.3.1	Example: POU block	7
1.6.3.2	Automatic transfer within MXChange.....	8
1.7	Mapping from or to MXChange nodes.....	8
1.7.1	PLC information	8
1.7.2	Serial communication card.....	8
1.7.3	Ethernet communication card.....	9
1.7.4	MELSEC-NET communication card (serial master).....	9
1.7.5	MELSEC NET communication card (Ethernet master)	9
1.7.6	Tag information	10
1.7.6.1	Mapping of simple variables.....	10
1.7.6.2	Mapping of strings	10
1.7.6.3	Mapping of arrays	11
1.7.6.4	Mapping of FBs.....	11
1.7.6.5	Mapping of DUTs.....	12
1.7.7	Profi-MAP POU-blocks.....	12
1.8	User interface	13
1.8.1	Marking MXChange tags in MM+ 2.40.....	13
1.8.2	Update with the MXChange database.....	13
1.8.3	Special MXChange menu commands	14
1.8.3.1	New menu commands.....	14
1.8.3.2	Availability of the MXChange menu items	14
1.8.3.3	Project menu.....	15
1.8.3.4	Extra menus	25
1.8.4	New toolbar icon: Logon	25
1.8.5	GVL declaration editor.....	26
1.8.5.1	Declaring as MXChange tag.....	26
1.8.5.2	Deleting an MXChange tag	27
1.8.5.3	Restrictions.....	27
1.8.6	ASCII Import and Export.....	27
1.8.6.1	Import and Export of the MXChange flag.....	27
1.8.6.2	Import of MXChange variables into an MXChange project.....	28

1.8.6.3	Import of MXChange variables into a non-MXChange project	28
1.8.7	User actions influenced by MXChange.....	28
1.8.7.1	Save Project.....	28
1.8.7.2	Save Project as	29
1.8.7.3	ASCII import and export	29
1.8.7.4	Change of the PLC type.....	29
1.9	Database	30
1.9.1	Handling.....	30
1.9.2	Database objects.....	30
1.9.2.1	Status information	30
1.9.2.2	Connection settings	30
1.9.2.3	MXChange tag list	31
1.9.2.4	Clear-flags list	31
1.10	MXChange update mechanisms.....	32
1.10.1	General behaviour of MM+ 2.40	32
1.10.2	UserModification flags	32
1.10.3	Order of events.....	33
1.10.3.1	General update.....	33
1.10.3.2	Project save.....	33
1.10.3.3	Project enable MXChange.....	33
1.10.4	Update MM+ 2.40 → MXChange	34
1.10.4.1	Delete nodes.....	34
1.10.4.2	Modify nodes	34
1.10.4.3	Add nodes	35
1.10.4.4	POU blocks	36
1.10.5	Update MXChange → MM+ 2.40	36
1.10.5.1	Deleted nodes.....	36
1.10.5.2	Added nodes	37
1.10.5.3	Modified nodes	37
1.10.6	Restrictions	37

Publication history

Date	Version	Description
October 1999	1.0	First edition

1 MXChange

By default the MXChange functionality is enabled in MELSEC MEDOC *plus* from version 2.40a onwards.

1.1 Global tools

1.1.1 Super Project Manager

The Super Project Manager provides an overview of the project, allowing convenient browsing and editing all configuration elements. The Super Project Manager administrates the super-project and runs on a 32-bit PC.

1.1.2 MXChange interface tag editor

The tag editor can be accessed from any configuration program and from the Super Project Manager. The tag editor is included in the Super Project Manager and can add tags for any or all packages. MM+ 2.40 can access the tag editor to ask for details needed in other configuration packages by using an MXChange interface call.

1.2 MXChange update MM+ 2.40 <-> MXChange server

1.2.1 Moments of MXChange updates

The MXChange update is cogent:

- when starting MM+ 2.40 or opening an MXChange project respectively
- when saving an MXChange project

The MXChange update is optional:

while running MM+2.40 with an MXChange project (e.g. indicated by a timer or certain actions for example saving the GVL. It is also possible that the user determines the update moment (forced update).

1.2.2 The online flag

Each application can set the online flag. If the online flag has been set, the application will be informed automatically by the MXChange server when any tag used by the application has been changed. If the online-flag is not set, the application must ask the MXChange server for changing any tag.

1.2.3 Transaction logging

All transactions taken by MM+ 2.40 will be stored in a local audit-log. All audit-logs must have a time stamp and will provide the name of the user who has taken the transaction (if available). This is done by the MXChange server.

1.2.4 Offline synchronisation

If the MXChange server is not available, the Super Project Manager and the MXChange interface tag editor will terminate and display an error message. However, MM+ 2.40 still allows to change MXChange tags. The changes are stored in the local MM+ 2.40 database of the project and they are updated when the MXChange server is online again.

1.3 Import and export

1.3.1 ASCII import and export

An ASCII import or export of the additional MXChange data is possible because the additional MXChange data is used also in other projects. Note the following:

- the additional import or export can be disabled to stay compatible with old formats.
- old export files can be imported also.

1.3.2 Import of Profi-MAP-POU-blocks → MM+ 2.40

Profi-Map-POU-blocks can be transferred automatically to the corresponding MM+ 2.40 project. These blocks exchange PLC data with a profibus DP master station. The objects are described in the following.

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
ID	Long		Unique identifier for this node in the hierarchy
Description	Text	254	POU block description
POU block	Long binary		ASCII POU block file for mapping, stored in full.

1.4 MM+ 2.40 objects

In the following all objects are listed which MM+ 2.40 knows (in MXChange database format). How to map these objects to MM+ 2.40 will be described later.

1.4.1 PLC information

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
ID	Long		Reserved to be used by the Super Project Manager
Project name	Text	254	File name of the MM+ 2.40 project (DOS naming conventions)
Home PC	Text	60	Network name of the PC where the project is located
Online flag	Boolean		If set to TRUE, changes are dealt with while running
Password	Text	30	Password for the project

1.4.1.1 Serial communication card

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
ID	Long		Reserved to be used by configuration programs
SlotNo	Integer		Slot number in the PLC
StationNo	Integer		Station number for the serial communication card in the master PLC (range between 0 and 31)

1.4.1.2 Ethernet communication card

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
ID	Long		Reserved to be used by configuration programs
SlotNo	Integer		Slot number in the PLC
TCPaddr	Text	30	TCP/IP address of the card, e.g. 193.1.1.1
Port	Long		Port number to use (range between 256 and 65536)

1.4.1.3 MELSEC-NET communication card (serial master)

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
ID	Long		Reserved to be used by configuration programs
SlotNo	Integer		Slot number in the PLC
StationNo	Integer		Station number for the serial communication card in the master PLC

MXChange for Melsec Medoc plus 2.40

			(range between 0 and 31)
PLCaddr	Integer		Slave address (range between 0 and 64)

1.4.1.4 MELSEC-NET communication card (Ethernet master)

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
ID	Long		Reserved to be used by configuration programs
SlotNo	Integer		Slot number in the PLC
TCPaddr	Text	30	TCP/IP address of the card, e.g. 193.1.1.1
Port	Long		Port number to use (range between 256 and 65536)
StationNo	Integer		Station number for the serial communication card in the master PLC (range between 0 and 31)
PLCaddr	Integer		Slave address (range between 0 and 64)

1.4.2 MM+ 2.40 tag information

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
ID	Long		Reserved to be used by configuration tools.
Position	Integer		Position, to be used with structures
CDE_name	Text	68	MXChange tag name
CDE_parent	Long		Parent hierarchy ID (for structures).
CDE_ID	Long	-	Hierarchy ID of the MXChange tag (for efficiency)
Variable name	Text	68	Variable identifier (16 characters max.)
Description	Text	128	Variable description
Function block	Text	32	Function block identifier
Function instance	Text	32	Function block instance identifier
Variable Class	Integer		VAR, VAR_CONSTANT, VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR_EXTERNAL, VAR_EXTERNAL_CONSTANT, VAR_GLOBAL, VAR_GLOBAL_CONSTANT
IO-type	Integer		BOOL, INT, DINT, WORD, DWORD, READ, TIME
IO-dev	Text	4	I/O-device, e.g. D, TC
IO-addr	Long		I/O address, e.g. 100
IO-bit	Integer		Bit within the I/O address, otherwise -1
Data size	Long		Size of data - normally 1, but it may be longer for arrays

1.4.3 Mapping from ProfiMap I/O references

The Super Project Manager maps ProfiMap structures to single MM+ 2.40 variables.

Automatic conversion:

ProfiMap		MM+ 2.40
SD0-SD255	to	D9000-D9255
SM0-SM255	to	M9000-M9255

1.5 Network communication

To establish the connection to the MXChange server, MM+ 2.40 prompts for the necessary settings. Connection settings can be the MXChange server name, network name, or the TCP/IP address of the server's PC. These data should refer to a single project to be able to establish a connection to different MXChange servers including different projects. To ease the handling, a default connection setting can be included in the INI file.

MM+ 2.40 uses the API interface to provide a local TCP/IP listener. In case of timeouts (default 10 seconds) MM+ 2.40 repeats the request once. If there is still an error, the connection shuts down.

1.6 MXChange tag types

1.6.1 PLC parameters and online settings

The MXChange Super Project Manager contains information about communication cards (serial communication card, Ethernet communication card, MELSEC-NET communication card). Some of the data necessary to define these objects are included in the PLC parameters and the online settings (Transfer setup).

1.6.2 Global variables

All entries in the GVL can be defined to be MXChange tags. The GVL entries can be shared to the following types:

- normal variables
- arrays of normal variables
- global DUTs
- global FBs

1.6.3 POU blocks of MM+ 2.40

1.6.3.1 Example: POU block

(* Created by MELSEC ProfiMap *)

(* PROFIBUS DP Configuration = POU.DP *)

```
PLC_CONFIG
END_PLC_CONFIG
```

```
PROGRAM POU
```

```
'IL'
```

```
BODY
```

```
    WORKSPACE
```

```
        NETWORK_LIST_TYPE := NWTYPEIL ;
```

```
        ACTIVE_NETWORK := 0 ;
```

```
    END_WORKSPACE
```

```
    NET_WORK
```

```
        NETWORK_TYPE := NWTYPEUSR1 ;
```

```
        NETWORK_LABEL := ;
```

```
        NETWORK_TITLE := ;
```

```
        NETWORK_HEIGHT := 14
```

```
        NETWORK_BODY
```

```
TABINFO 6, 14
```

(*Exchange PLC data with Profibus DP*)

```
LD    X1B
```

```
AND  X1D
```

```
TO    H00 K960 K2Y80 K1
```

```
TO    H00 K992 K4YA0 K1
```

```
TO    H00 K1008 D16 K16
```

```
OUT  Y00
```

```
AND  X00
```

```
FROM H00 K0 K2X80 K1
```

```
FROM H00 K16 K4X90 K1
```

```
FROM H00 K48 D0 K16
```

```

        END_NETWORK_BODY
    END_NET_WORK
END_BODY
END_PROGRAM
CONFIGURATION scConfiguration
    RESOURCE scResource ON scResourceRype
        TASK
            POU_ACTIVE(SINGLE:=TRUE,INTERVAL:=0,PRIORITY:=31)

        PROGRAM POU WITH POU_ACTIVE:scProgramType

    END_RESOURCE
END_CONFIGURATION

```

1.6.3.2 Automatic transfer within MXChange

By implementing MXChange exporting and importing POU blocks will be eased. The POU block is stored in the MXChange database and can be fetched by MM+ 2.40 by a user command or automatically during the normal updates.

1.7 Mapping from or to MXChange nodes

The data structures used within the MXChange server differ from the data structures used in MM+ 2.40. Some data structures are even not available in one of the communication partners. Therefore, it is necessary to convert the data structures when exchanging these data. The following chapter describes how to convert data structures.

1.7.1 PLC information

<i>MXChange type</i>	<i>MM+ 2.40 type</i>	<i>Notes</i>
ID	-	
ProjectName	Project directory name of MM+ 2.40	MM+ 2.40 project name must be limited (DOS naming conventions)
HomePC	-	Can be extracted using the WINSOCK.DLL
OnlineFlag	Project data	Must be defined in MM+ 2.40
IPLCtype	Project data	Convert from the internal PLC type setting

Conclusion

To get the necessary information, additional dialogs must be defined. The OnlineFlag and the Password can be fixed for a single project. The name of the HomePC can be extracted using the WINSOCK.DLL.

1.7.2 Serial communication card

<i>MXChange type</i>	<i>MM+ 2.40 type</i>	<i>Notes</i>
ID	-	
SlotNo	(-)	MM+ 2.40 does not provide any information about the slot numbers of serial communication modules. Only the user can define a name for the special modules, so MM+ 2.40 cannot decide which module is a serial communication card.
StationNo	-	MM+ 2.40 does not provide any information about the station number. The station number must be fixed on the module itself.

Conclusion

MM+ 2.40 does not provide any information about serial communication cards.

1.7.3 Ethernet communication card

<i>MXChange type</i>	<i>MM+ 2.40 type</i>	<i>Notes</i>
ID	TS_ETHERNET_TYPE	
SlotNo	-	MM+ 2.40 does not provide any information about the slot numbers of Ethernet communication modules. Only the user can define a name for the special modules ,so MM+ 2.40 cannot decide which module is a Ethernet communication card.
TCPaddr (Text 15)	BYTE IPAddress1 BYTE IPAddress2 BYTE IPAddress3 BYTE IPAddress4	The settings must be transformed to a string and back.
Port (long)	long PortNo	

Conclusion

These settings are included in the online setup data of MM+ 2.40, but they are only valid if the Ethernet communication card is used for program download. Otherwise these data will contain default values.

1.7.4 MELSEC-NET communication card (serial master)

<i>MXChange type</i>	<i>MM+ 2.40 type</i>	<i>Notes</i>
ID	-	
SlotNo	-	
StationNo	-	
PLCaddr	-	

Conclusion

MM+ 2.40 does not know whether the master PLC of a MELSEC NET configuration has a serial communication card.

1.7.5 MELSEC NET communication card (Ethernet master)

<i>MXChange type</i>	<i>MM+ 2.40 type</i>	<i>Notes</i>
ID	-	
SlotNo	-	
TCPaddr	-	
Port	-	
PLCaddr	-	

Conclusion

MM+ 2.40 does not know whether the master PLC of a MELSEC NET configuration has an Ethernet communication card.

1.7.6 Tag information

The tag information objects (Local-I/O tags) will be mapped to global variables in MM+ 2.40.

<i>MXChange type</i>	<i>MM+ 2.40 type (declaration)</i>	<i>Notes</i>
ID	-	
Posn	-	
CDE_name	-	
CDE_parent	-	
CDE_ID	-	-
Variable name	szName	Must be limited to 16 chars when mapping to MM+ 2.40 type
Description	pszComment	The description in MXChange is only 64 chars long. Therefore, pszComment must be limited when mapping to MXChange nodes.
FunctionBlock	-	Case of FBs
FunctionInstance	-	Case of FBs
VariableClass	pszClass	Must be transformed from string to integer and vice versa
IOtype	pszType	Must be transformed from string to integer and vice versa
IOdev IOaddr IObit	pszAddress	pszAddress must be shared to Text[2] (e.g. M), a long value, and a bit-offset Example: pszAddress: D10.1 IOdev: D IOaddr: 10 IObit: 1 (-1 if no bit addressing)
iIOelemsize	pszType	Size of a single element
iIOelems	pszType	Used for arrays.

1.7.6.1 Mapping of simple variables

A simple variable is a variable of one of the following types BOOL, INT, DINT, WORD, DWORD, REAL, or TIME. A simple variable does not include arrays of the types mentioned above nor STRING.

Mapping specials

- Posn: not used, 0 ?
- FunctionBlock always empty
- FunctionInstance always empty
- iIOelemsize size of a single element (depending on the type)
- iIOelems always 1

1.7.6.2 Mapping of strings

To have the option to include arrays of strings in later implementations, the string length will be stored in iIOelemsize rather than in iIOelems.

Mapping specials

- Posn: not used, 0 ?
- FunctionBlock always empty
- FunctionInstance always empty
- iOelemsize string length
- iOelems always 1

1.7.6.3 Mapping of arrays

The arrays can be of one of the types mentioned above.

Mapping specials

- Posn: not used, 0 ?
- FunctionBlock always empty
- FunctionInstance always empty
- iOelemsize size of a single element(depending on the element type)
- iOelems specifies the array size, must be taken off pszType maximum is 65535.

1.7.6.4 Mapping of FBs

MM+ 2.40 does not know any addresses of global FBs until the project has been compiled. Therefore, MM+ 2.40 can provide only FB addresses of compiled projects. Since the FB addresses may change after each compiling, other applications or the tag editor are not allowed to change FB I/O data or the FB structure.

The Super Project Manager displays an FB in the following way:

```
PLC
|
+-- LOCAL I/O
|
|   +-- "Name of the FB instance"
|   |
|   |   +-- "1st member of FB"
|   |   +-- ...
|   |   +-- "Last member of FB"
|   |
|   +-- "a simple variable"
```

Mapping specials

<i>Tag entry</i>	<i>FB node</i>	<i>1st FB member</i>	<i>Last FB member</i>
Parent-ID	PLC	FB node	FB node
Posn	-	0 (1)	count (count-1)
VariableName	FB instance name	name of 1 st member	name of last member
FunctionBlock	FB name	FB name	FB name
FunctionInstance	FB instance name	FB instance name	FB instance name
iIOelemsize	1	depends on the type	depends on the type
iIOelems	1	1, if no array	1, if no array

1.7.6.5 Mapping of DUTs

The Super Project Manager displays a DUT in the following way:

```

PLC
|
+-- LOCAL I/O
|
|   +-- "Name of the DUT instance"
|   |
|   |   +-- "1st member of DUT"
|   |   +-- ...
|   |   +-- "Last member of DUT"
|   |
|   +-- "a simple variable"

```

Mapping specials

<i>Tag entry</i>	<i>DUT node</i>	<i>1st member of DUT</i>	<i>Last member of DUT</i>
Parent-ID	PLC	DUT-node	DUT node
Posn	-	0 (1)	count (count -1)
VariableName	DUT instance name	name of 1 st member	name of last member
FunctionBlock	DUT name	DUT name	DUT name
FunctionInstance	DUT instance name	DUT instance name	DUT instance name
iIOelemsize	1	depends on the type	depends on the type
iIOelems	1	1, if no array	1, if no array

1.7.7 Profi-MAP POU-blocks

<i>MXChange type</i>	<i>MM+ 2.40 type FB</i>	<i>Notes</i>
ID	-	
Description (Text 127)	pszComment	
POUblock (long binary)		The binary data can be imported into MM+ 2.40 using the standard MM+ 2.40 ASCII import mechanism.

Conclusion

The binary data provided by the MXChange server can be imported into MM+ 2.40 using the standard import mechanism of MM+ 2.40. A task will be created containing the imported POU. The import mechanism is one way only (MXChange→MM+ 2.40), because it does not make sense to change anything by means of MM+ 2.40 and store it back to the MXChange server. MP+ is responsible for the POU block

1.8 User interface

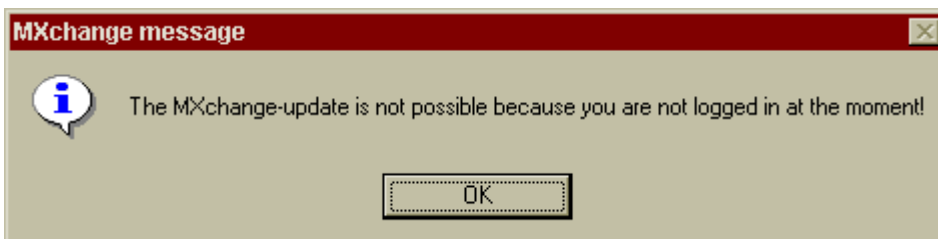
1.8.1 Marking MXChange tags in MM+ 2.40

The only tags which can be marked specially as MXChange tags are variables included in the GVL. The mark is a simple flag ("X") as known from the marking of Autoexternal variables. The column is named *MXChange* and located next to the Autoexternal column. The *MXChange* column is available only if the current project is an MXChange project (MXChange has been enabled) and within the GVL editors (not the header editors). In GVL editors of user libraries the *MXChange* column is available, too, but it does not have any function yet.

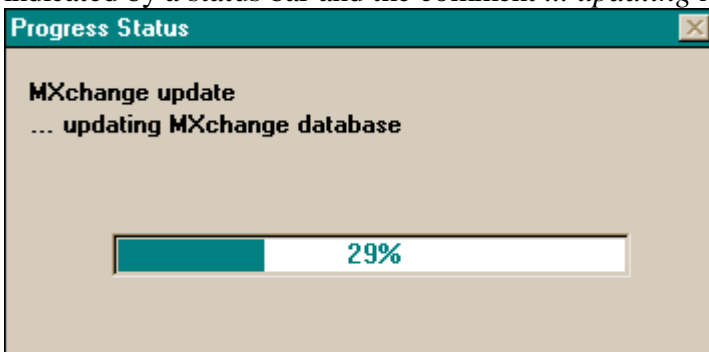
1.8.2 Update with the MXChange database

The MXChange update will be performed when saving the MM+ 2.40 project. This additional function is available only if the project currently loaded is an MXChange project. If it is not an MXChange project, no update is performed.

When you are not logged in the MXChange server, no update with the MXChange database is possible. In this case the following message appears:

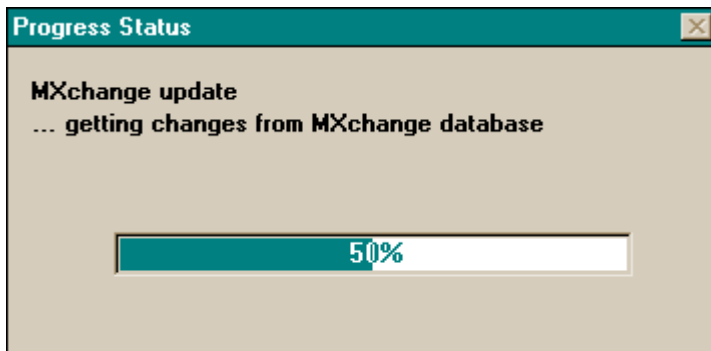


When you are logged in the MXChange server, MM+ 2.40 informs the MXChange server about relevant changes made in MM+ 2.40 since the last successful update. The progress status will be indicated by a status bar and the comment ... *updating MXChange database*.



MXChange for Melsec Medoc plus 2.40

After that MM+ 2.40 asks the server whether there are any changes relevant for the MM+ 2.40 project currently loaded. If there are relevant changes, MM+ 2.40 receives these changes and integrates them into the loaded project. The progress status will be indicated by a status bar and the comment ... *getting changes from MXChange database*.



1.8.3 Special MXChange menu commands

1.8.3.1 New menu commands

Project

...
Enable MXChange
MXChange actions → Log on
Show Changes
Call Tag-editor
Global Adjust
Change Password
Disable MXChange

Extras

...
Declare MXChange tags
Delete MXChange tags
...

1.8.3.2 Availability of the MXChange menu items

	Project closed	Non-MXChange project open	MXChange project open and logged on	MXChange project open and not logged on
Enable MXChange	D	E	D	D
Disable MXChange	D	D	D	E
Log on	D	D	E	E
Show Changes	D	D	E	D
Call MXChange-tag editor	D	D	E	D
Global Adjust	D	D	E	D
Change password	D	D	E	D

D = menu item disabled

E = menu item enabled

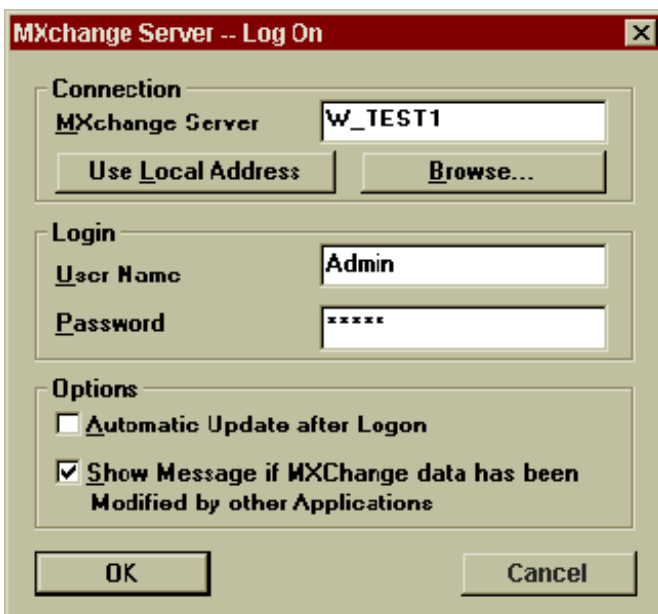
1.8.3.3 Project menu

1.8.3.3.1 Enable MXChange

The item **Enable MXChange** is available only if the current project is a non-MXChange project. The function does not work if any GVL editor is open at the moment. In this case an error message appears:

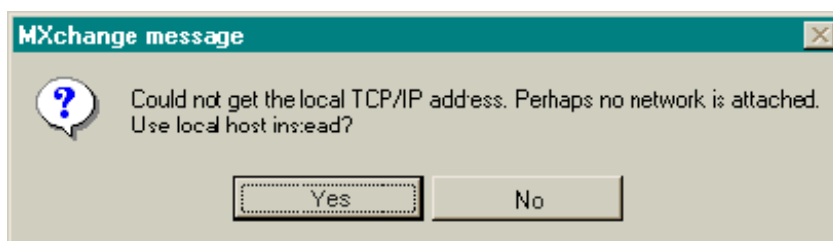


If all GVL editors are closed, you are asked to insert the TCP/IP address or the network name of the machine the MXChange server is running on. Furthermore, the login details are required.



Use Local Address

The button *Use Local Address* inserts the TCP/IP address of the PC MM+ 2.40 is running on. In some cases the TCP/IP address cannot be determined, e.g. if the PC is not running in a network. If this is the case, you are asked whether the local host address should be used instead (which is 127.0.0.1 by define).



Yes: The local host address (127.0.0.1) is inserted into the address field.

No: Nothing is inserted into the address field.

Browse

The button **Browse** opens a list box where you can select a server to connect. The timeout used to connect the servers is 3000 ms by default. This value can be modified by an entry in the **INI** file. In the following example the timeout is set to 5000 ms.

[MXChange]

...

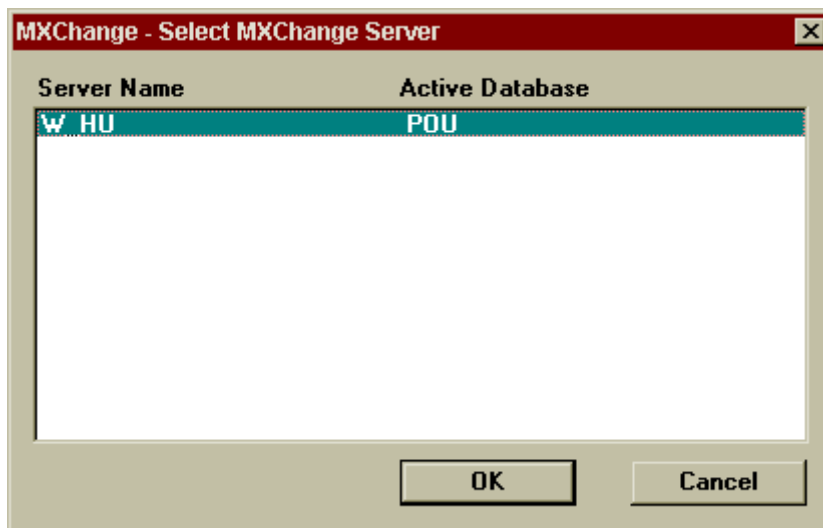
BrowseServerTimeout=5000

...

Possible error messages:

Error while looking for MXChange servers.

Too many servers found. Not all can be displayed.



OK: The dialog will be closed and the selected server address is inserted into the field **MXChange server** of the **logon** dialog.

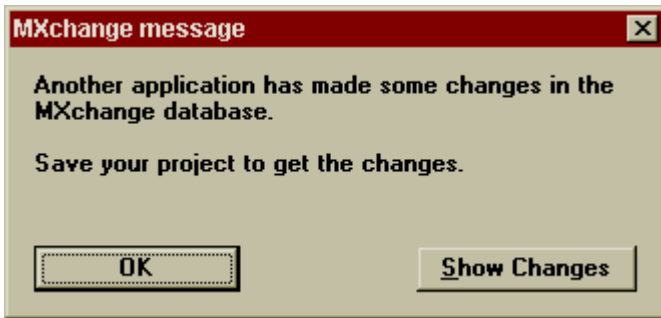
Cancel: Return to the **logon** dialog

Automatic Update After Logon

If the item **Automatic Update After Logon** is selected, the MXChange data will be updated immediately after each logon (similar to **Project->Save**).

Show Message if MXChange data has been Modified by other Applications

If this item is selected, MM+ 2.40 will display a message if any other application has changed MXChange data of the current MM+ 2.40 project. This message appears the first time when a change has been made after the last update (project save), i.e. it does not appear for each change.



OK:

The dialog will be closed and the address of the selected server is inserted into the field *MXChange server* of the **logon** dialog.

Show Changes:

A dialog will be opened to display the changes made by other applications. This dialog is described in section 1.8.3.3.3: MXChange action → Show Changes.

The last state of this flag is stored in the **INI** file of MM+ 2.40

```
...  
[MXChange]  
...  
AutomaticUpdate = TRUE  
...
```

Leaving with Cancel

The dialog will be closed.

Leaving with OK

MM+ 2.40 checks the connection to the MXChange server. If no connection is available, one of the following error messages will appear:

Possible error messages:

- The connection to the MXChange server timed out!
- The MXChange server has been disconnected!
- A not specified MXChange interface error occurred!

After the error message has been acknowledged the dialog appears again. The user can now decide to try it once again, to modify the parameters, or to cancel the action.

After a connection to the MXChange server has been established successfully there are two possible cases:

1. The PLC project does not exist in the selected MXChange server

In this case MM+ 2.40

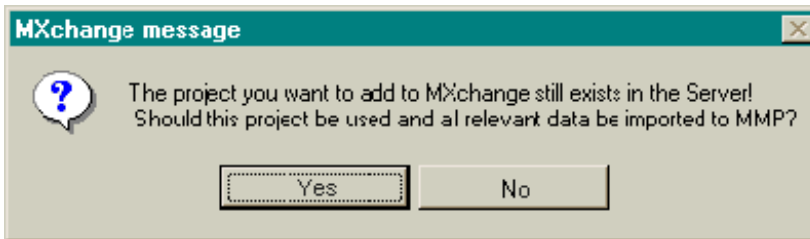
- adds a new PLC node to the MXChange database. The new PLC node is named according to the MM+ 2.40 project.
- adds a new Global Variables node below the PLC node.
- creates additional DB objects in the local MM+ 2.40 database needed for MXChange.
- inserts a new column to the GVL editors lists.
- logs on to the MXChange server.

Possible error messages:

The connection to the MXChange server timed out!
The MXChange server has been disconnected!
A not specified MXChange interface error occurred!
Failed while trying to add nodes!

2. The PLC project still exists within the MXChange server

This case occurs if the Super Project Manager includes already a PLC node which has the same name as the MM+ 2.40 project. A message appears and you can decide whether you want to use this project node and import all data into the current project.



No:

The introduction of MXChange to the MM+ 2.40 project will be cancelled and the **logon** dialog appears again. You have the opportunity to establish a connection to another server.

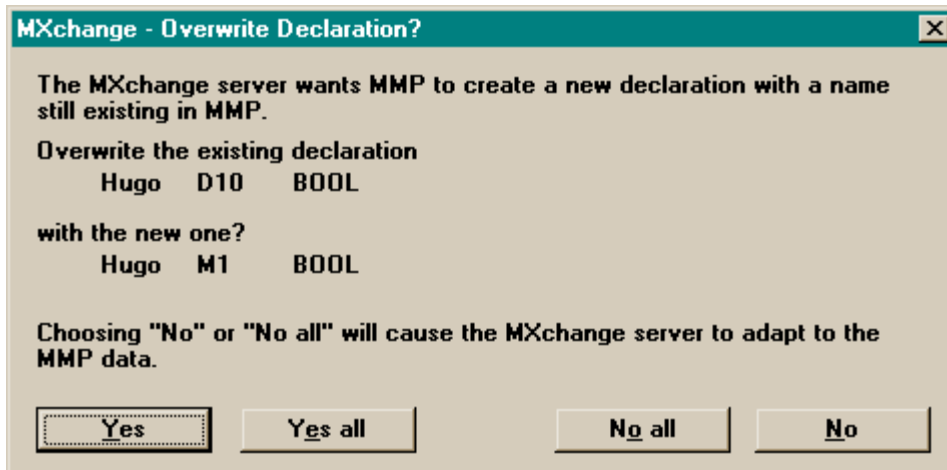
Yes:

- All MM+ 2.40 tags of the corresponding PLC node will be imported into the GVL of the current MM+ 2.40 project.
- ProfiMap POU blocks will be integrated.
- For each imported POU block a corresponding task will be created.

Possible error messages:

The connection to the MXChange server timed out!
The MXChange server has been disconnected!
A not specified MXChange interface error occurred!
The item number was outside the range of available items!
More data available!
An error occurred while importing MXChange data to MM+ 2.40!

If the MXChange database contains an MXChange tag with the same name as a declaration still existing the following message appears:



Yes: The MM+ 2.40 declaration will be overwritten.

No: The MM+ 2.40 declaration will not be overwritten, but will be declared as an MXChange tag. During the next updating the MXChange server will be forced to adapt its MXChange tag to the corresponding MM+ 2.40 declaration.

Yes all: Yes for all declarations.

No all: No for all declarations.

If the database includes empty or destroyed POU blocks, a message box appears which asks you whether you want to continue.

Yes: The POU block will be ignored and the import will be continued.

No: The import will be aborted.

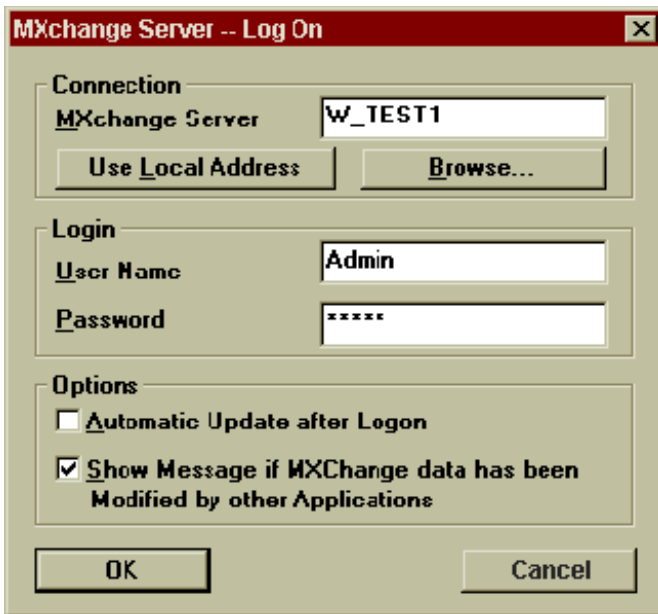
1.8.3.3.2 MXChange action → Logon

If you are logged in when selecting the menu command **logon**, a logoff is performed. If you are logged off when selecting the menu command **logon**, a logon is performed.

The logon status is shown by a **]** in front of the menu item.

The menu command **logon** is available only if an MXChange project is open.

The logon procedure



Cancel

The dialog will be closed.

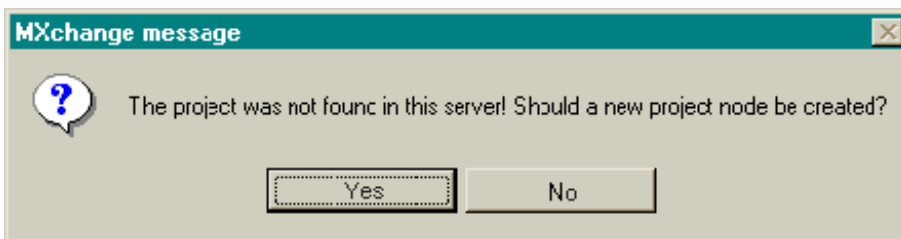
OK

MM+ 2.40 logs on to the specified MXChange server using the given user name and password. The following error messages may appear:

Possible error messages:

- The connection to the MXChange server timed out!
- The MXChange server has been disconnected!
- A not specified MXChange interface error occurred!
- The password is incorrect!
- The MXChange server has no open database.

If the logon procedure does not find the project in the selected MXChange database (for example if the PLC node has been deleted within the Super Project Manager, the following message will be displayed:



Yes:

A new project node will be created in the MXChange database. During the next update (when saving the MM+ 2.40 database) all MXChange tags stored in the MM+ 2.40 database will be exported to the new project node.

No:

The logon will be cancelled.

During each logon the MM+ 2.40 project node is updated with the current data (project path, home PC etc.).

The logoff procedure

Possible error messages:

The connection to the MXChange server timed out!

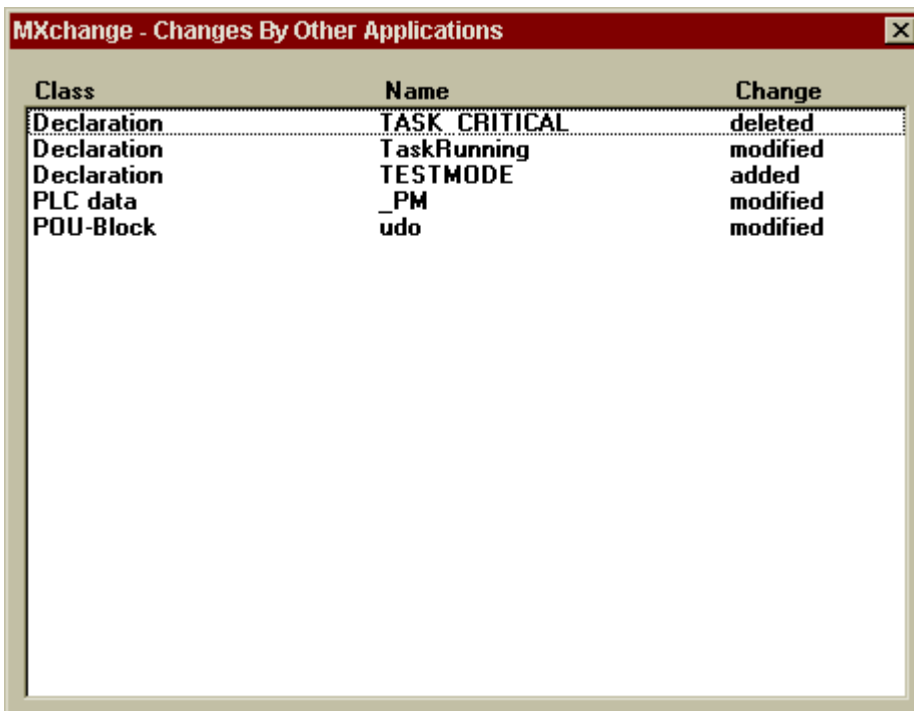
The MXChange server has been disconnected!

A not specified MXChange interface error occurred!

You are not logged in!

1.8.3.3.3 MXChange action → Show Changes

The menu item **Show Changes** opens a window indicating the changes made by other applications. The list includes all changes concerning the project node and its child nodes which have to be considered by MM+ 2.40.



Class	Name	Change
Declaration	TASK CRITICAL	deleted
Declaration	TaskRunning	modified
Declaration	TESTMODE	added
PLC data	_PM	modified
POU-Block	udo	modified

The list is sorted alphabetically and includes three different classes: declarations, POU blocks, and the PLC data. Each item has one of the three change flags: deleted, added, or modified..

1.8.3.3.4 MXChange action → Call Tag editor

The menu item **Call Tag editor** calls the global MXChange tag editor. To use this function a connection to the MXChange server must be available and the current status must be logged on.

Possible error messages:

- The connection to the MXChange server timed out!
- The MXChange server has been disconnected!
- A not specified MXChange interface error occurred!

1.8.3.3.5 MXChange action → Global Adjust

It might be that both databases (MM+ 2.40 and MXChange) do not have exactly the same tags or data or some tags differ by single items.

The following reasons are possible for that:

- using a backup version of the MXChange database
- using an older MM+ 2.40 project, i.e. you do not use the current MM+ 2.40 project
- malfunction of the normal update mechanism

The standard update mechanism does not note these cases. To realise an acceptable update duration, only data which has been changed is updated.

Starting the Global Adjustment

The global adjustment is an extra command.

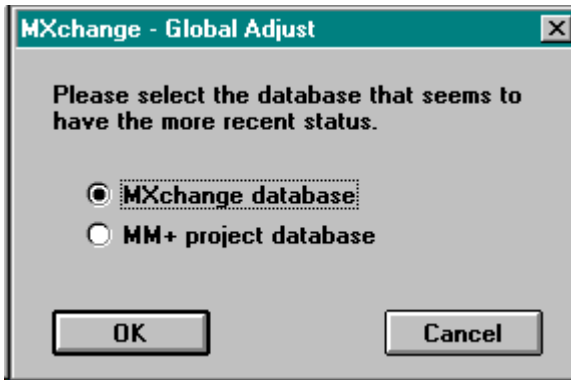
By means of the global adjustment you can select either MM+ 2.40 or MXChange to include the more recent data.

If you decide that

1. MM+ 2.40 includes the more recent data,
 - all tags not defined in MM+ 2.40 but in MXChange must be imported to MM+ 2.40.
 - all tags not defined in MXChange but in MM+ 2.40 must be exported to MXChange.
 - tags defined in MXChange and MM+ 2.40 will be adjusted to the tags defined in MM+ 2.40, i.e. the tags in MXChange will be overwritten.
 -
2. MXChange includes the more recent data,
 - all tags not defined in MM+ 2.40 but in MXChange must be imported to MM+ 2.40.
 - all tags not defined in MXChange but in MM+ 2.40 must be exported to MXChange.
 - tags defined in MXChange and MM+ 2.40 will be adjusted to the tags defined in MXChange, i.e. the tags in MM+ 2.40 will be overwritten.

After you have selected the menu item **Global Adjust**,

- update the MM+ 2.40 and MXChange databases if they are not consistent
- select the database including the more recent state of data (MM+ 2.40 project database or MXChange database)
- select Cancel or OK

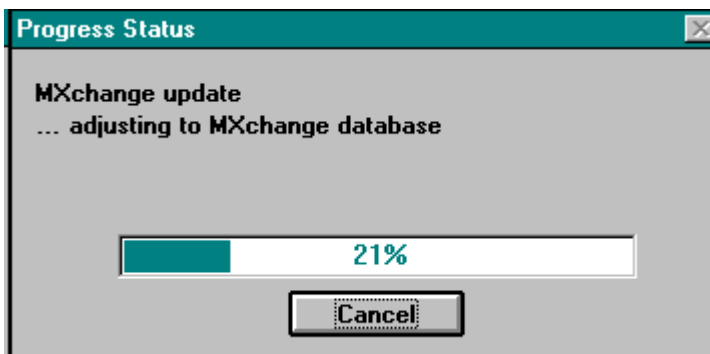


Save Project

The menu command **Save Project** includes also the standard update mechanism necessary to ensure that all changed data have been exported or imported respectively. After the update the project will be saved. Now the global adjustment may start.

Global Adjustment

The update mechanism depends on the database which has been selected: MM+ 2.40 is master or MXChange is master.



```
//----- direction MM+ → MXChange
For (all MXChange tags defined in MM+)
{
    If(Add to MXChange successful)
    {
        break;
    }
    else if (MM+ is master)
    {
        try to modify the tag
    }
}

//----- direction MXChange → MM+
Fetch all nodes defined below the MXChange project node
For (all tags)
{
    if(found in MM+)
    {
        if(MXChange is master)
            replace tag in MM+
        else
            do nothing
    }
    else //not found in MM+
    {
        import MXChange data to MM+
    }
}
```

Technical details

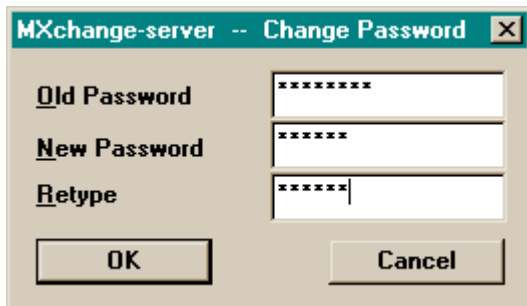
Update MM+ 2.40→MXChange

```
For(all MXChange items defined in MM+)
Item->IsDefined
|
+-- OK --- Adjust MM+ --- Item->CDEModify
|   +- Adjust MXC --- nothing |
|   |                         +-- OK           --- Adjust internal lists
|   |                         +-- No Memory    --- Error Message, Abort Global Adj
|   |                         +-- Interface error --- Error Message, Abort Global Adj
|   |                         +-- Not Found --- Err:Could not Modify, go on with next
|   |                         +-- General    --- Err:Could not Modify, go on with next
|   |
+-- Interface Error, abort Global Adjust
+-- default, do nothing
|
+-- Not Found -----|
+-- Not Supported-----+ Item->CDEAdd
|
|   +-- OK           --- Adjust internal lists
|   +-- No Memory    --- Error Message, Abort Global Adj
|   +-- Interface error --- Error Message, Abort Global Adj
|   +-- General      --- Err:Could not Add,
|                       Ask user to delete item in MM+
|                       YES: delete item
|                           remove item from list
|                       NO: do nothing
|                       go on with next
```

Update MM+ 2.40→MXChange

```
Adjust MM+ ----- ImportAllMXChangeData(Do not overwrite anything in MM+)
Adjust MXChange -- ImportAllMXChangeData(overwrite all in MM+)
```

1.8.3.3.6 MXChange action → Change Password



Cancel:

The dialog will be closed.

OK:

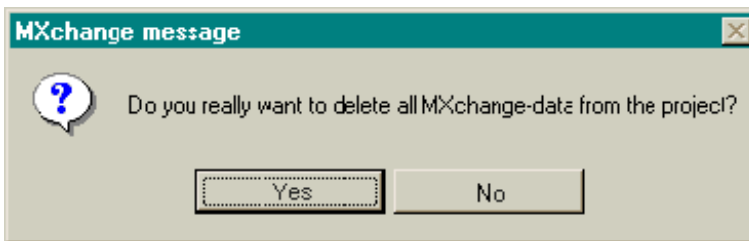
- MM+ 2.40 checks whether the new password and the retyped password are identical. If they are not identical, the following error message appears: *Re-entered password is not the same as first password!* Now you can correct the inputs.
- MM+ 2.40 tries to change the password by calling MXChange server functions. If this is not successful, one of the following error messages appears:
 - The old password is incorrect!
 - The connection to the MXChange server timed out!
 - The MXChange server has been disconnected!
 - A not specified MXChange interface error occurred!

1.8.3.3.7 MXChange actions → Disable MXChange

The menu item **Disable MXChange** is available only if the project currently loaded is an MXChange project. **Disable MXChange** removes all MXChange data from the project database and the MXChange column in the GVL. Via this menu item the PLC project will not be removed from the Super Project. This has to be done using the Super Project Manager. To disable MXChange all GVL editors must be closed. Otherwise the following message appears:



To ensure that you really want to delete all MXChange data, the following message box appears:



Yes:

All MXChange data included in the MM+ 2.40 database will be deleted.

No:

The window will be closed.

1.8.3.4 Extra menus

Declare MXChange tags

The menu item **Declare MXChange tags** is available only if the GVL editor is open and has the input focus and at least one declaration is selected. For non-MXChange projects **Declare MXChange tags** is not available.

All marked declarations will be defined as MXChange tags (see also section Declaring as MXChange tag on page 26). Declarations already defined as MXChange tags will not be influenced by this.

Delete MXChange tags

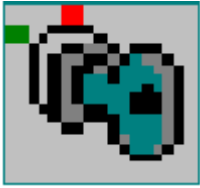
The menu item **Delete MXChange tags** is available only if the GVL editor is open and has the input focus and at least one declaration is selected. For non-MXChange projects **Delete MXChange tags** is not available.

All declarations that are defined as MXChange tags will be declared to be non-MXChange tags (see also section Deleting an MXChange tag on page 27). The other declarations will not be influenced.

1.8.4 New toolbar icon: Logon

MM+ 2.40 includes a new toolbar icon: Logon. However, this toolbar icon is visible only if MXChange is enabled via the file **MM+.INI** (see below). Otherwise it is not visible.

...
[MXChange]
MXChange = TRUE
...



The logon icon does the same as the corresponding menu item. It can have one of the following states:

Disabled if...

- no project is open.
- the open project is a non-MXChange project.

Pushed if...

- the status is logged on.

Normal if...

- the status is not logged on.

Tool tip: **Logon/Logoff to MXChange Server**

1.8.5 GVL declaration editor

If the current project is an MXChange project, another column named MXChange is added to the GVL. If the current project is not an MXChange project, the appearance will not be modified.

An X in the MXChange column indicates that this variable is declared to be an MXChange tag.

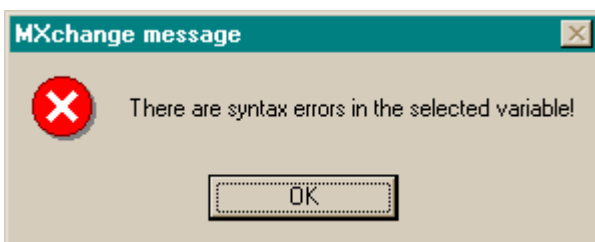
User library GVLs cannot declare variables as MXChange tags although they include also the additional column. If you click into the MXChange field, a system beep will be generated.

1.8.5.1 Declaring as MXChange tag

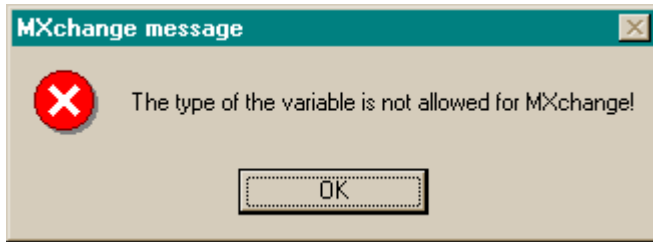
To declare a variable as MXChange tag, click into the field using the left mouse button or press **SPACE** within this field.

There are variables which cannot be declared as MXChange tags. There are the following restrictions:

1. The variable declaration must be valid. Otherwise the following message appears:



- The variable type must be of the following types: BOOL, INT, STRING, DINT, WORD, DWORD, REAL, TIME, or ARRAYS of the types mentioned before. Otherwise the following message appears:



If you try to change an MXChange tag type into a type not allowed in MXChange, the change is not accepted and the tag type which has been valid before entering the invalid type will be valid again when leaving the edit field.

1.8.5.2 Deleting an MXChange tag

To delete an MXChange tag, remove *X* from the MXChange column in the GVL. During the next update (when saving the MM+ 2.40 project) this information will be given to the MXChange server. If you delete a variable which is an MXChange tag, it will also be removed from the MXChange database.

1.8.5.3 Restrictions

It is not allowed to change the name of a, MXChange declaration in the GVL editor. If you try to open the edit field of the variable name a message beep will be generated.

1.8.6 ASCII Import and Export

1.8.6.1 Import and Export of the MXChange flag

To be compatible to lower MM+ versions '~C' is appended to the comment string of variables if it is an MXChange tag as well. The example below shows the export of a GVL containing two MXChange variables (aInt, aInt2).

```
(*SOFTCONTROL:
  VERSION:2.03.09*)
CONFIGURATION scConfiguration
  RESOURCE scResource ON scResourceType

  VAR_GLOBAL
    aInt AT %MW0.11: INT:=0;
      (*A Comment~C*)
    aInt2 AT %MW0.11: INT:=0;
      (*A Comment~C*)
  END_VAR

  END_RESOURCE
END_CONFIGURATION
```

1.8.6.2 Import of MXChange variables into an MXChange project

When importing MXChange variables into an MXChange project, the variables will be flagged as MXChange tags. During the next update with the MXChange database the MXChange variables will be transferred to the MXChange server.

1.8.6.3 Import of MXChange variables into a non-MXChange project

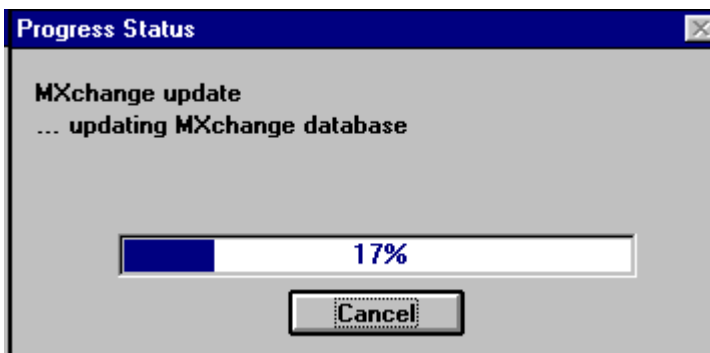
When importing MXChange variables into a non-MXChange project, the MXChange flag will be ignored.

1.8.7 User actions influenced by MXChange

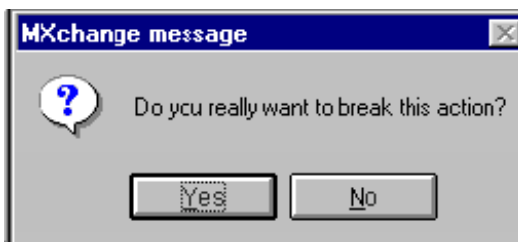
This section describes how standard MM+ 2.40 user actions will be influenced by MXChange.

1.8.7.1 Save Project

Before saving the MM+ 2.40 project database it will be updated with the MXChange server. The following dialog indicates the progress of the MXChange update.



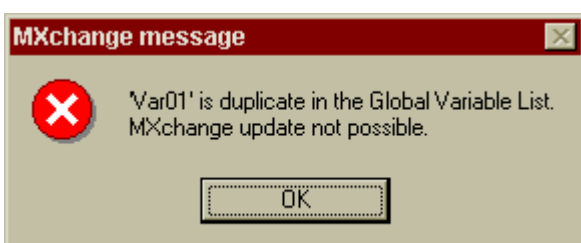
The user can cancel the update by pressing the Cancel button. To check whether the user really wants to break the update, the following message box appears:



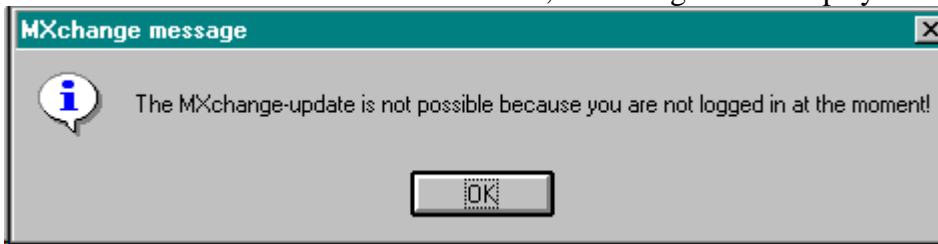
Yes: The update will be cancelled.

No: The update continues.

Before starting the update MXChange checks whether there are no duplicate names in the GVL. If there are duplicate names, the update will be cancelled and the following error message appears:



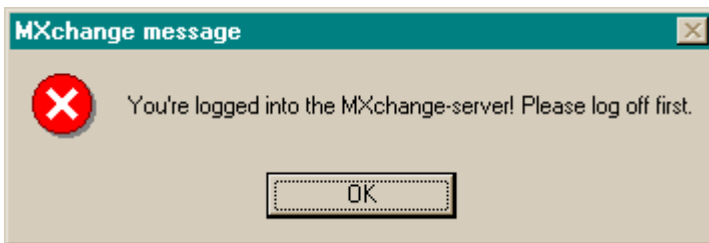
If the server is not available at the moment, a warning will be displayed.



The MM+ 2.40 project database will be updated during the next **Save Project**.

1.8.7.2 Save Project as ...

The user command **Save Project as ...** is allowed only if you are not logged in the MXChange server. **Save Project as ...** might cause problems, because the project name and its relation to the Super Project Manager will change. If you are logged in when saving the project as ..., the following message appears:



If you are not logged in and the message does not appear, the project will be saved including the new name. When you log in next time, MXChange notes the new project name and if the new project name is not available in the MXChange server, you will be asked to create a new project node.

The same will happen if an MXChange project has been renamed, restored, or copied under a new name.

1.8.7.3 ASCII import and export

It is not allowed to export data which must keep their changes made by means of MM+ 2.40 since an MXChange server connection could not be established during the last MM+ 2.40 session. These data represent a time dependent state which could be completely different when importing. The flags indicating MXChange variables are imported or exported respectively.

1.8.7.4 Change of the PLC type

The PLC type is stored in the MXChange database. Therefore, MXChange must be informed of the change of the PLC type in the current MM+ 2.40 project. This update will be done during the next logon. To ensure that MXChange is informed of the change, log off the MXChange server to change the PLC type.

If you are logged in the MXChange server when changing the PLC type, the following message will appear:



1.9 Database

1.9.1 Handling

The additional MXChange data will be added to each project that is opened by means of the MXChange version of MM+ 2.40. However, it is also possible to open an MXChange project with a non-MXChange version of MM+ 2.4023.

1.9.2 Database objects

1.9.2.1 Status information

The status information contains information about the current project status. The project name is needed to decide whether the project name has changed (**Project [] Import, Project [] Save as, Project [] Restore, Project [] Rename, Project [] Copy**).

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
flsCDEproject	Boolean	1	TRUE: project is an MXChange project FALSE: project is not an MXChange project
ICDEversion	Unsigned long	1	MXChange version of the project
SzMM+ProjectName	Text	256	Project name
SzLastUserName	Text	256	Name of the last user who has logged into the server.

1.9.2.2 Connection settings

The connections settings contain all information necessary to connect the MXChange server.

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
szCDEServerName	Text	256	Connection string
szCDEServerNetworkName	Text	256	Not used
TCP/IP-address	Unsigned long	1	Not used
Timeout	Unsigned long	1	Timeout in seconds. Not used
LFlags	Unsigned long	1	Bit 0 = 1 → online updates allowed Bit 0 = 0 → online updates not allowed

1.9.2.3 MXChange tag list

1.9.2.3.1 Frame

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
Number of elements	Long	1	Number of MXChange tags stored
Element 1			
...			
Element n			

1.9.2.3.2 Element structure

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
IMM+Id	ID	1	ID of the MM+ database object
ISPNodeId	Long	1	ID of the corresponding super project node
InodeClass	Long	1	One of the following: CDE_CLS_PLC_MM+LUS_TAG CDE_CLS_PLC_TCP_CARD CDE_CLS_PLC_SERIAL_CARD CDE_CLS_PLC_MELSEC_CARD_SERIAL CDE_CLS_PLC_MELSEC_CARD_ETH CDE_CLS_PLC_POU_BLOCK
UserModifications	Word	1	Bit 0 == 1: user wants to delete this node Bit 1 == 1: user wants to add this node Bit 2 == 1: user wants to modify this node

1.9.2.4 Clear-flags list

The clear-flag lists are arrays of Super Project node IDs. The flags of these nodes have to be cleared when the server is online again. For each deletion, addition, and modification there is a separate list. These lists will be stored in the following way.

Addition list

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
Number of elements	Long	1	ID numbers that will follow
SP IDs	Long	Number of elements	Super Project node IDs

Modification list

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
Number of elements	Long	1	ID numbers that will follow
SP IDs	Long	Number of elements	Super Project node IDs

Deletion list

<i>Field name</i>	<i>Type</i>	<i>Size</i>	<i>Description</i>
Number of elements	Long	1	ID numbers that will follow
SP IDs	Long	Number of elements	Super Project node IDs

1.10 MXChange update mechanisms

1.10.1 General behaviour of MM+ 2.40

MM+ 2.40 performs a general update during project saving. If the MXChange server is not available during the update, the changes will be stored only in the local MM+ 2.40 database. Additionally the changes are marked to be handled during next update.

Changes made by the MM+ 2.40 user will not be immediately reported to the MXChange server. They are handled during the next general update.

Changes made by other applications or the Super Project Manager will be accepted immediately if the online flag is set. Before any changes are made the MM+ 2.40 user will be informed of the change. MM+ 2.40 clears only the acknowledge flags, i.e. other flags are not cleared by MM+ 2.40. They will be cleared after the project has been saved.

1.10.2 UserModification flags

The local database in the additional MXChange data (for details see **Fehler! Verweisquelle konnte nicht gefunden werden.** on page **Fehler! Textmarke nicht definiert.**) includes information about all MXChange tags used in the MM+ 2.40 project. Each entry has three flags to indicate changes made by the MM+ 2.40 user after the last MXChange-server update. The flags indicate whether an MXChange tag was deleted, new created, or modified. Only one flag can be set at the same time.

If some changes concerning MXChange have been made within the GVL but have not been saved when closing the GVL, the changes made to the UserModification flags will not be saved neither.

The flags will be set in the following way.

The user marks a non-MXChange variable as MXChange variable

<i>Before user action</i>			<i>After user action</i>		
new	delete	modify	new	delete	modify
			X		
		X	X		
	X		X		
X			X		

The user modifies an MXChange variable

<i>Before user action</i>			<i>After user action</i>		
new	delete	modify	new	delete	modify
					X
		X			X
	X				X
X			X		

The user removes the MXChange mark from a variable

<i>Before user action</i>			<i>After user action</i>		
new	delete	modify	new	delete	modify
				X	
		X		X	
	X			X	
X				X	

The user creates a new project node of an existing MXChange project

You can create a new project node of an existing MXChange project when you establish a connection to another MXChange server. You will be asked whether you want to create a new project node. In this case the flags of all existing MXChange tags have to be adapted:

- Remove the deletion tags from the list.
- Convert all modified flags into added flags.
- Do not change the added flags.
- All other tags have to be flagged as added.

1.10.3 Order of events

1.10.3.1 General update

The general update starts integrating MM+ 2.40 changes into the MXChange server. After this the MXChange server changes will be transferred into the MM+ 2.40 database. All necessary MXChange API calls will be done using synchronous calls.

The general update will be done according to the following order:

1. MM+ 2.40→MXChange, deleted nodes
2. MM+ 2.40→MXChange, modified nodes
3. MM+ 2.40→MXChange, added nodes
4. MXChange→MM+ 2.40, deleted nodes
5. MXChange→MM+ 2.40, modified nodes
6. MXChange→MM+ 2.40, added nodes

1.10.3.2 Project save

```
Perform "General update"  
SaveProjectDB
```

1.10.3.3 Project enable MXChange

```
IF(Project still exists in MXChange-server)  
{  
    IF(user wants to use this one)  
    {  
        add new Database object to MM+-project  
        modify GVL-editor  
        add MXChange-menu to MM+  
        import all relevant MXChange-data to MM+-project  
        save project database  
    }  
    ELSE  
        do nothing  
}  
ELSE  
{
```

MXChange for Melsec Medoc plus 2.40

```
    add new Database object to MM+-project
    modify GVL-editor
    add MXChange-menu to MM+
    add new PLC-node to the super-project
    save project database
}
```

1.10.4 Update MM+ 2.40 → MXChange

MM+ 2.40 checks whether the MXChange data include local modification flags. If this is the case, MM+ 2.40 informs the MXChange server of the changes. The flags will be cleared when the MXChange server has accepted the changes.

1.10.4.1 Delete nodes

The event **Delete nodes** uses the API call **DeleteNodes**. When using this API call with numerous node IDs problems might be caused. For this reason one call can delete a maximum of 30 nodes. If more than 30 nodes have to be deleted, at least two API calls will be performed.

The starting point of this event is a list of all IDs to be deleted. As mentioned above this list must be divided into packages of 30 IDs. For these packages the following events will be done:

```
while (all Nodes handled)
  DeleteNodes (with a list of IDs)
  CDE_ERR_OK:      remove IDs from MM+ internal lists
  CDE_ERR_NOTFOUND: see in the error list which nodes returned which error code
                  CDE_ERR_NOTFOUND:      remove from MM+ internal lists
                  (assume that this node is still deleted)
                  call DeleteNodes again to avoid rollback
                  by the MXChange server
                  CDE_ERR_OK:
  CDE_ERR_XXX:     90% interface error
                  error message
                  exit update function
```

1.10.4.2 Modify nodes

```
For all Nodes to be modified
  FetchNodesList(WorkOutSize of the node)
  OK:  FetchNodesList(real fetch)
      OK:  ModifyNodes
          OK:  update MM+ internal lists
          ERR:  Interface Error?
              YES:  - "MXChange server not available any more.
                    Update was cancelled"
                    - cancel whole update
                    - exit function
              NO:  CDE_ERR_NOTFOUND || CDE_ERR_NOTEMPTY
                  YES:  - "Node '??' was not found in the MXChange
                        database.\n Delete this node in MM+ as well?
                        (YES/NO?)
                  YES:  - remove ID from MM+ internal lists
                        - delete reference in MM+ DB
                  NO:  - do nothing
              NO:  "Error while trying to modify '???' "
      ERR:  Interface Error?
          YES:  - "MXChange server not available any more.
                Update was cancelled"
                - cancel whole update
                - exit function
          NO:  - show fetch nodes error
                - cancel only for this node
                - go on with next node
  ERR:  Interface Error?
      YES:  - "MXChange server not available any more.
            Update was cancelled"
            - cancel whole update
            - exit function
      NO:  - show fetch nodes error
            - cancel only for this node
            - go on with next node
```

1.10.4.3 Add nodes

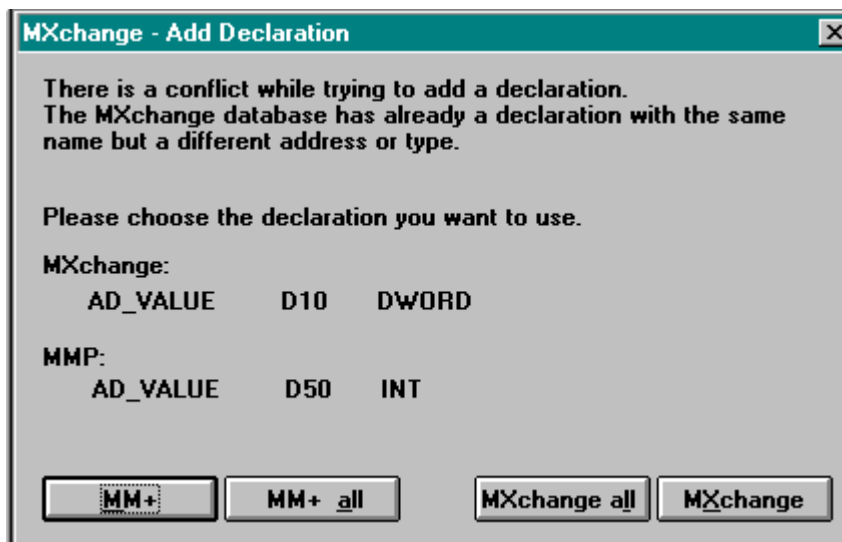
1.10.4.3.1 Restrictions

The following table shows in which cases double names of MXChange tags are allowed.

	Tags in different projects	Tags in the same project
Same addresses or types, different names	yes	yes
Different addresses or types, same names	yes	no
Same addresses or types, same name	yes	no

1.10.4.3.2 MXChange tag with same name already defined

If the MXChange database includes already a tag with the same name but a different address or type, the following dialog appears. You can select to overwrite the MM+ 2.40 declaration or to overwrite the MXChange tag.



1.10.4.3.3 Order of events

Input: list of CDETagInfo of all tags to be added. For each tag the following has to be performed.

```
LookUpCdeTags(Heuristic)
case CDE_ERR_OK
    switch(LookUpCdeTags(Name)
    case CDE_ERR_OK
        - to be sure insert new SPid into MM+-internal lists
        - update MM+ internal lists
    case CDE_ERR_NOTFOUND
        - fall through to Heuristic CDE_ERR_NOTFOUND
    default
        Interface Error ?
        YES: "MXChange server not available any more.
            Update was cancelled"
            cancel whole update
            exit function
        NO: "Error while trying to add '%s"
            Try to remove the MXChange flag
```

```
                if ok update lists
                cancel this node
                go on with next node
case CDE_ERR_NOTFOUND
- fill AddCdeTag-struct
- perform API call AddCdeTags
case CDE_ERR_OK
- create new CDETagInfo and replace this in internal list
default
Interface Error ?
YES:  "MXChange server not available any more.
      Update was cancelled"
      cancel whole update
      exit function
NO:   "Error while trying to add '%s"
      Try to remove the MXChange flag
            if ok update lists
            cancel this node
            go on with next node
default
Interface Error ?
YES:  "MXChange server not available any more.
      Update was cancelled"
      cancel whole update
      exit function
NO:   "Error while trying to add '%s"
      Try to remove the MXChange flag
            if ok update lists
            cancel this node
            go on with next node
```

A new procedure especially for GVL declarations is given in the following:

```
if(! try to add the tag)
{
    if(LookUpName == OK)
    {
        Ask user which tag should be overwritten
        case OVERWRITE MM+
            modify the MM+-tag (import MXChange tag data, save decl)
            break
        case OVERWRITE MXChange
            modify the MXChange tag (fillMXChangeTag struct, ModifyNode in MXC)
            break
        try to add the tag again
    }
    else
        error "cannot add tag to MXChange"
}
else
    everything ok
```

1.10.4.4 POU blocks

POU blocks cannot be added, modified, or deleted by MM+ 2.40.

1.10.5 Update MXChange → MM+ 2.40

1.10.5.1 Deleted nodes

1.10.5.1.1 GVL declarations

To delete GVL declarations, take the following steps:

- Get the declaration.
- Remove the MXChange flag.
- Save the declaration.
- Remove the entry of this tag from the tag list.

1.10.5.1.2 POU blocks

To delete POU blocks, take the following steps:

- Delete corresponding task and POU in the MM+ 2.40 database.
- Remove the entry from the tag list.

1.10.5.2 Added nodes

If the MXChange server reports an added node, a new GVL variable marked as MXChange tag will be created. The new variable will be inserted into the MXChange object of MM+ 2.40. New POU blocks will be handled in the same way.

1.10.5.2.1 GVL declarations

At first MM+ 2.40 searches for a declaration with the same name. If MM+ 2.40 finds a declaration with the same name, you will be asked to overwrite this declaration.

Yes: load decl, modify decl, store decl

No: create new decl, modify decl, store decl

If the type of the imported MXChange tag is not allowed in MM+ 2.40, the type will be set to "". Additionally this declaration is marked as modified to be updated during the next update MM+ 2.40→MXChange.

1.10.5.3 Modified nodes

1.10.5.3.1 GVL declarations

For modified nodes there are the following GVL declarations: load decl, modify decl, store decl

If the type of the imported MXChange tag is not allowed in MM+ 2.40, the type will be set to "". Additionally this declaration is marked as modified to be updated during the next update MM+ 2.40→MXChange.

1.10.5.3.2 POU blocks

- Delete corresponding task and POU in the MM+ 2.40 database
- ASCII import of the POU block binary data

1.10.6 Restrictions

When updating the MXChange server there is a maximum of 4500 tags which can be added, modified, or deleted at the same time.