



# MxLOADER USER GUIDE

Extract and load data in Maximo with ease

*Author: Bruno Portaluri*

*Updated on: 2019-04-03*

# 1 CONTENTS

<b>2</b>	<b>Introduction.....</b>	<b>4</b>
2.1.1	System Requirements.....	4
<b>3</b>	<b>User Guide .....</b>	<b>5</b>
<b>3.1</b>	<b>Basic Usage .....</b>	<b>5</b>
3.1.1	Insert, Update and Delete Records.....	7
3.1.2	Child Objects.....	8
3.1.3	Change Status of Objects.....	8
3.1.4	Various Tips .....	8
<b>3.2</b>	<b>Advanced Topics .....</b>	<b>9</b>
3.2.1	Update Data Dictionary .....	9
3.2.2	Using Web services .....	9
3.2.3	Web Authentication .....	10
3.2.4	Querying object IDs .....	10
3.2.5	Uploading attachments (DOCLINKS).....	11
3.2.6	Finding and fixing data inconsistencies .....	12
3.2.7	Excel Tips and Tricks .....	13
3.2.8	Securing Maximo server .....	14
<b>4</b>	<b>MxLoader Templates and Examples.....</b>	<b>15</b>
<b>4.1</b>	<b>Administration.....</b>	<b>16</b>
4.1.1	People.....	16
4.1.2	Person Groups .....	16
4.1.3	Labor.....	16
4.1.4	Classification Hierarchy .....	17
4.1.5	Specifications Attributes.....	17
4.1.6	Classification Attributes.....	18
<b>4.2</b>	<b>Assets .....</b>	<b>19</b>
4.2.1	Assets.....	19
4.2.2	Assets Spare Parts .....	19
4.2.3	Asset Classification and Attributes .....	19
4.2.4	Locations.....	20
4.2.5	Locations Hierarchy .....	20
4.2.6	Failure Codes .....	22
<b>4.3</b>	<b>Financial .....</b>	<b>24</b>
4.3.1	Currency Codes and Exchange Rates.....	24
<b>4.4</b>	<b>Inventory .....</b>	<b>25</b>
4.4.1	Items.....	25
4.4.2	Tools .....	25
4.4.3	Item/Organization association .....	25
4.4.4	Storerooms .....	26
4.4.5	Inventory .....	26

4.4.6	Inventory Balance .....	26
4.4.7	Units of Measure .....	27
4.4.8	Units of Measure Conversion .....	27
4.4.9	Commodities.....	28
<b>4.5</b>	<b>Purchasing .....</b>	<b>29</b>
4.5.1	Companies .....	29
<b>4.6</b>	<b>Work.....</b>	<b>29</b>
4.6.1	Work Orders .....	29
<b>4.7</b>	<b>Planning.....</b>	<b>29</b>
4.7.1	Job Plans .....	29
4.7.2	Routes.....	30
<b>4.8</b>	<b>Security.....</b>	<b>31</b>
4.8.1	Users.....	31
4.8.2	Users/Groups Association .....	32

## 2 INTRODUCTION

MxLoader is a Microsoft® Excel® spreadsheet that allows to quickly and easily query and load data into IBM Maximo, IBM SmartCloud Control Desk (SCCD) and other TPAE based applications.

It brings together the power of the Maximo Integration Framework and the flexibility of Excel allowing to quickly manipulate and import data into any Maximo database.

This document helps you getting familiar with MxLoader and use it at its full potential. Chapter 3 contains a quick tutorial of the tool describing its basic features and some and advanced techniques that can be used. Chapter 4 contains a set of useful templates and examples to be used in many common data loading scenarios.

*MxLoader is provided "as is" and, subject to any statutory warranties which can not be excluded, the author makes no warranties or conditions either express or implied, including but not limited to, the implied warranties or conditions of merchantability, fitness for a particular purpose, and non-infringement, regarding it.*

*The author will not be liable for any direct, indirect, special, incidental or consequential damages arising out of or relating to any use or distribution of MxLoader.*

- ✓ **To download MxLoader, ask for help or suggest improvements join the [MxLoader community](#).**
- ✓ **The latest version of this document is available in the [Files section](#) the MxLoader community.**

---

### 2.1.1 System Requirements

MxLoader runs on any Windows PC with a recent version of Office installed. No particular add-on or plug-ins are required.

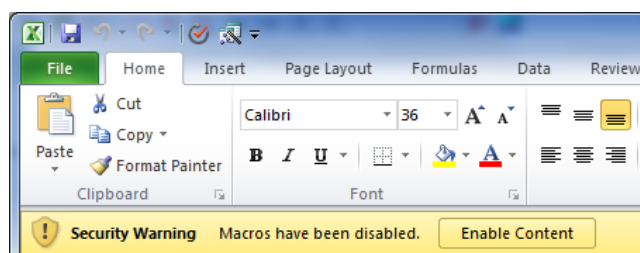
- Microsoft Excel 2007/2010/2013
- Microsoft XML Core Services (MSXML) - Included in Windows Vista, Windows 7 and Windows XP Service Pack 3. It can also be [downloaded from Microsoft](#).

## 3 USER GUIDE

Before starting to work with MxLoader ensure you have the latest version by connecting to the [MxLoader community](#).

### 3.1 BASIC USAGE

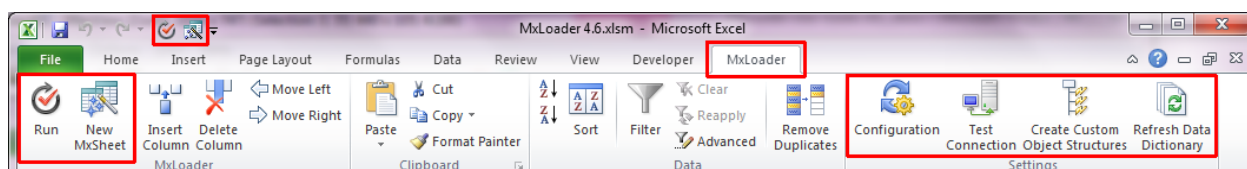
Open the MxLoader file with Microsoft Excel and enable macros if you receive a prompt like this.



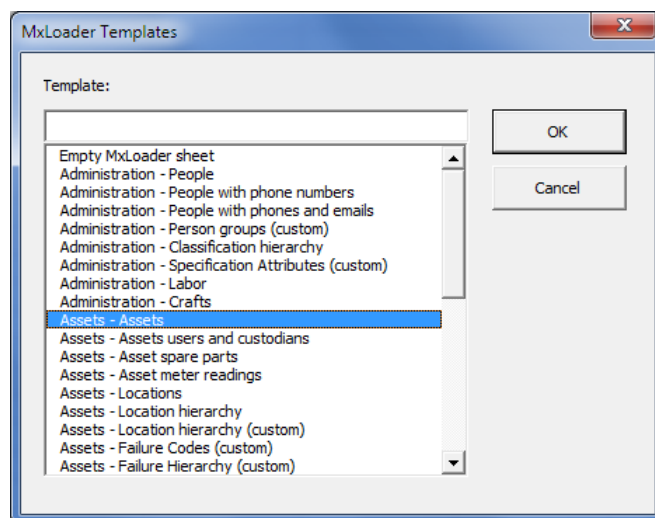
After that, go to *Config* sheet and update the configuration based on your environment. The most common parameters to be set are the 'Server address' and authentication settings if needed. Use the 'Test Connection' button to check your configuration.

	A	B	C	D	E
1	Configuration				
2					
3	Connection				
4	Server address	<a href="http://[MXSERVER]/meaweb/os/">http://[MXSERVER]/meaweb/os/</a>			
5	Authentication	None			
6	User	wilson			
7	Password				
8	Proxy				
9					
10	Resolve timeout	5			
11	Connect timeout	10			
12	Send timeout	10			
13	Receive timeout	120			

MxLoader has a custom Excel ribbon that provides access to MxLoader functions and other useful shortcuts to Excel data manipulation tools.



The *New MxSheet* button will open a dialog that allows creating a new worksheet from a set of predefined templates.



Pick a simple template (e.g. Asset) and press the OK button.

A new worksheet will be created with a predefined header. The first row points to the MIF Object Structure and the second row contains the list of attributes to work with.

To query data from Maximo you just have to press the *MxRun* button (the rounded arrow) on the Quick Access Toolbar. The worksheet will be cleaned from previous data and populated with information retrieved from your server.

	A	B	C	D	E	F	G
1	MXASSET	ASSET	Query				
2	ASSETNUM	SITEID	DESCRIPTION	STATUS	LOCATION	PARENT	
3	10001	BEDFORD	3MG1E201800160	OPERATING	IT-015192-006		
4	10002	BEDFORD	5DTDC14UR03126	OPERATING	IT-015192-006		
5	10003	BEDFORD	5DTDGUWK3210	OPERATING	IT-015192-006		
6	10004	BEDFORD	AASTRAIPBX1001	OPERATING	IT-006003-003		
7	10005	BEDFORD	AASTRAIPBX1002	OPERATING	IT-027025-001		
8	10006	BEDFORD	AASTRAIPBX1004	OPERATING	IT-078045-007		
9	10007	BEDFORD	AASTRAIPBX1031	OPERATING	IT-089001-004		
10	10008	BEDFORD	AASTRAIPBX1034	OPERATING	IT-072006-034		
11	10009	BEDFORD	AASTRAIPBX1035	OPERATING	IT-037006-055		
12	1001	BEDFORD	Fire Extinguisher	ACTIVE	MOFLOOR1		
13	10010	BEDFORD	AASTRAIPBX1037	OPERATING	IT-037006-044		
14	10011	BEDFORD	AASTRAIPBX1039	OPERATING	IT-001063-002		
15	10012	BEDFORD	AASTRAIPBX1040	OPERATING	IT-011011-001		

To filter data you can type any SQL *where* clause in cell D1.

	A	B	C	D	E	F
1	MXASSET	ASSET	Query	description like '%Ext%'		
2	ASSETNUM	SITEID	DESCRIPTION	STATUS	LOCATION	PARENT
3	1001	BEDFORD	Fire Extinguisher	ACTIVE	MOFLOOR1	
4	1002	BEDFORD	Fire Extinguisher	ACTIVE	MOFLOOR2	
5	1003	BEDFORD	Fire Extinguisher	NOT READY	MOFLOOR3	
6	1004	BEDFORD	Fire Extinguisher	NOT READY	MOFLOOR4	
7	1005	BEDFORD	Fire Extinguisher	NOT READY	CONF400	
8	1006	BEDFORD	Fire Extinguisher	NOT READY	OFF401	
9	1007	BEDFORD	Fire Extinguisher	NOT READY	OFF402	

To sort data you can type any SQL *order by* clause in cell E1.

	A	B	C	D	E	F
1	MXASSET	ASSET	Query		assetnum desc	
2	ASSETNUM	SITEID	DESCRIPTION	STATUS	LOCATION	PARENT
3	V1050	BEDFORD	MS Visio	OPERATING	OFF104	A7704
4	V1044	BEDFORD	MS Visio	OPERATING	FIELDSTAFF	7120
5	V1043	BEDFORD	MS Visio	OPERATING	FIELDSTAFF	7124
6	V1042	BEDFORD	MS Visio	OPERATING	FIELDSTAFF	7122
7	V1041	BEDFORD	MS Visio	OPERATING	FIELDSTAFF	7131
8	V1040	BEDFORD	MS Visio	OPERATING	FIELDSTAFF	7129
9	V1039	BEDFORD	MS Visio	OPERATING	FIELDSTAFF	7130
10	V1038	BEDFORD	MS Visio	OPERATING	FIELDSTAFF	7128

### 3.1.1 Insert, Update and Delete Records

Cell C1 allows specifying a different action. The **Sync** action allows inserting and updating records. By using Query and Synchron actions it is quite easy to perform massive updates and inserts.

The available actions are:

- **Query:** Query the Maximo server and fills the active sheet with the received data.
- **Sync-AddChange:** Add or update records depending on whether the primary record exists in the database. The child objects will be created or updated to match the input data. It is similar to the 'Sync' action but child objects will never be deleted.
- **Sync:** Add or update records depending on whether the primary record exists in the database. The child objects will be created, updated or deleted to match the input data.
- **Sync-Add:** Add records to the target system. Fails if any of the records already exist.
- **Sync-Change:** Update existing records in the target system. Fails if any of the records do not already exist.
- **Delete:** Delete existing records in the target system.

Refer to the [Action attributes paragraph](#) of Maximo Integration Manual for more details.

### 3.1.2 Child Objects

Create a new worksheet from the **People with phone numbers** template. Look at cells E2:G2 to understand how to specify child objects.

	A	B	C	D	E	F	G	H
1	MXPERSON	PERSON	Query					
2	PERSONID	FIRSTNAME	LASTNAME	LOCATIONSITE	PHONE. PHONENUM	PHONE. TYPE	PHONE. ISPRIMARY	
3	REVIS	Akash	Kumar		(617) 745-0879	WORK	1	
4	STANLEY	Frederick	Stanley		(613) 643-1933	WORK	1	
5	MILLS	Keith	Mills		(617) 734-7965	WORK	1	
6	JACOBY	Mark	Jacoby		(617) 594-0249	WORK	1	
7	BPILLA	Bob	Pilla	BEDFORD	781-335-9234	WORK	1	
8	BPILLA	Bob	Pilla	BEDFORD	781-335-0002	WORK	0	
9	BPILLA	Bob	Pilla	BEDFORD	781-335-0432	HOME	0	
10	BOUDREAU	Jane	Boudreau	BEDFORD	781-335-0002	WORK	1	
11	TORY	Tory	Little		011-04-866-896-6775	WORK	1	
12	GANESH	Ganesh	Venka	BEDFORD	781-335-6655	WORK	1	
13	CROCKER	Robert	Crocker		617-555-9846	WORK	1	

✓ MxLoader compares the values of the attributes in the main object to group child objects when syncing data.

### 3.1.3 Change Status of Objects

Many record types in TPAE applications include a special STATUS field. MxLoader uses a standard MIF interface so you have to take care of MIF rules when trying to update this field.

Refer to the following IBM docs to understand how to modify the status of an object:

- [Change status using the Integration Framework \(Maximo wiki\)](#)
- [Changing status with an inbound IF or MEA transaction \(TechNote 1407820\)](#)

### 3.1.4 Various Tips

**Null values** can be specified using ~NULL~ key.

If the DESCRIPTION\_LONGDESCRIPTION non-persistent field is enabled on the Object Structure you can use it to read and set object's **long descriptions**. Here is an [IBM Tech Note](#) that may help.

To **interrupt long operations** you may try pressing CTRL+BREAK once or ESC key repeatedly.

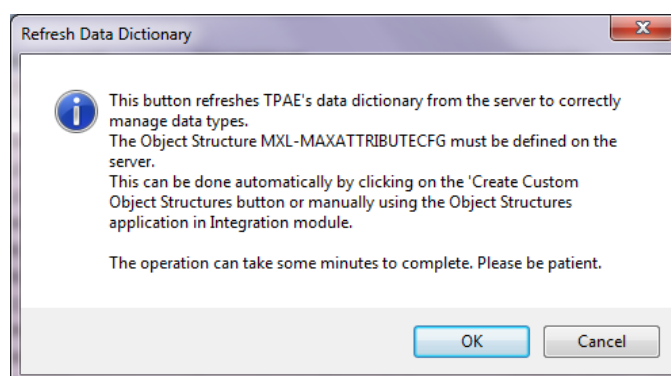


## 3.2 ADVANCED TOPICS

### 3.2.1 Update Data Dictionary

MxLoader is aware of the Maximo attributes data types. This is important to correctly manage number and dates formats.

MxLoader has a preloaded dictionary from a vanilla Maximo 7.5. The 'Update Data Dictionary' button on the MxLoader ribbon allows loading the data dictionary from your server.



### 3.2.2 Using Web services

By default MxLoader uses plain Object Services through HTTP calls. These services are automatically exposed when an Object Structure is created in Maximo. Using Web Services is possible to have more control over data defining processing classes, user exit classes, XSL maps, and processing rules.

To enable MxLoader to work with Web Services you need two settings in the Config sheet:

- **Server address** must be changed to `http://[MXHOST]/meaweb/services/`
- **Use Web Service** option must be set to 'True'

When a WebService is defined from an Object Structure you can just specify the WS name in cell A2 as you normally do. However, when the WebService is defined from an Enterprise Service you have to specify the web service name with a dotted notation in cell A2 with format `[EXTSYSNAME].[ESNAME].[OSNAME]`. For example, if your External System is 'MYEXTSYS' your Enterprise Service is named 'MYESVC' and based on 'MXASSET' you have to specify MYEXTSYS\_MYESVC.MXASSET in cell A2.

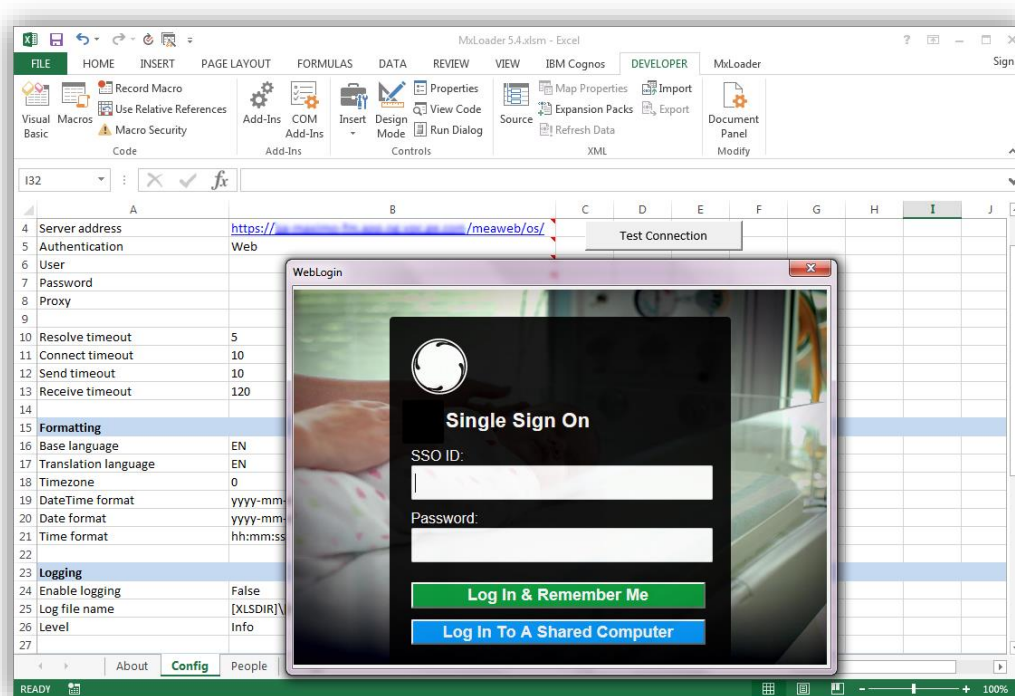
	A	B	C	D
1	MYEXTSYS_MYESVC.MXASSET.MXASSET	ASSET	Query	
2	ASSETNUM	SITEID	DESCRIPTION	STATUS
3				
4				
5				

### 3.2.3 Web Authentication

MxLoader 5.4 introduces a web login feature to support SSO web authentication technologies like OAuth or SAML.

To use the web authentication you must install the *WebLogin.dll* component on your Windows client. You can [download the WebLogin component](#) and install it following the instructions in the readme file.

By setting the *Authentication* property to 'Web' in the *Config* worksheet, MxLoader will launch the web browser in a popup window and display the page specified in the Server Address setting. When a web authentication is configured on Maximo MIF servlet (meaweb), this will display the login page where the user has to enter its credentials.



After successful authentication, the server will reply with a session ID in a secure cookie. This cookie can be captured by the WebLogin component and used by MxLoader in any subsequent call to the Maximo MIF interface.

To find out the name of the cookie in your environment you may need to install a plugin like [Firefox SAML Tracer](#) to inspect HTTP packets. Enabling the HTTP inspection and performing the authentication with the web browser you should be able to identify the name of the session cookie.

Session cookie name and other related settings like the size of the popup window are located in the 'Web Authentication' section of the Config sheet.

### 3.2.4 Querying object IDs

MxLoader is able to fill key attributes of synched objects.

To identify the key attributes you use square brackets: [WONUM]

This is particularly useful when the autonumbering feature is enabled and you want to retrieve the generated ID of the created objects.

	A	B	C	D	E	F
1	<b>MXWO</b>	<b>WORKORDER</b>	<b>Sync</b>			
2	<b>[WONUM]</b>	<b>SITEID</b>	<b>DESCRIPTION</b>	<b>STATUS</b>	<b>ASSETNUM</b>	
3		BEDFORD	Test1	WAPPR	1001	
4		BEDFORD	Test2	WAPPR	1002	
5		BEDFORD	Test3	WAPPR	1003	
6		BEDFORD	Test4	WAPPR	1004	
7		BEDFORD	Test5	WAPPR	1005	
8						

- ✓ To use this feature you must set the *mxe.int.keyresponse* to '1'. This will include the key attributes of the updated objects in each response. This feature is available starting from TPAE 7.1.1.6 as described in [IBM TechNote 1414088](#).

### 3.2.5 Uploading attachments (DOCLINKS)

Starting from Maximo 7.1.1.6 it is possible to import attachments to an object through MIF using Base64 encoding. MxLoader has the capability to read files from the filesystem, encode them in Base64 and upload them on Maximo. This can be done on any Maximo object that supports attachments but the integration object structure has to be modified to include the DOCLINK object.

In this example the MYASSET object structure has been created with the DOCLINKS child object.

Object Structure:

MYASSET

Consumed By: INTEGRATION  Integration Application

Authorized Application:

Outbound Definition Class:

Inbound Processing Class:

Query Only? ☐

Self Reference? ☐

Support Flat Structure? ☐

User Defined? ☒

Configurable? ☒

Alias Conflict? ☐

Source Objects for MYASSET  Filter  1 - 2 of 2

Object	Parent Object	Object Location Path	Relationship	Object Order	User Defined?
ASSET		ASSET		1	<input checked="" type="checkbox"/>
DOCLINKS	ASSET	ASSET/DOCLINKS	DOCLINKS	1	<input checked="" type="checkbox"/>

New Row

To specify a file to be uploaded you have to prefix the full path of the file with the 'file:/// ' string. Look at the screenshot below for an example of how to add two attachments to an asset.

MYASSET	ASSET	Sync-AddChange	DOCLINKS.	DOCLINKS.	DOCLINKS.	DOCLINKS.	DOCLINKS.
ASSETNUM	SITEID	ADDINFO	URLTYPE	DOCTYPE	DESCRIPTION	DOCUMENTDATA	
21000	BEDFORD		1 FILE	Attachments	test.txt	file:///C:/MyTemp/test.txt	
21000	BEDFORD		1 FILE	Attachments	logo.png	file:///C:/MyTemp/MxLoaderLogo.png	

### 3.2.6 Finding and fixing data inconsistencies

If you are receiving error messages from the server when syncing data you can try to set the 'Batch size' setting to 1 to send each object in a separate transaction. This will allow identifying the wrong data in your sheet.

MxLoader will stop and notify you of the error so you can fix the data. When clicking on the Run button again it will ask if you want to restart from the failed row ('Continue sync' setting).

	A	B	C	D	E	F
1	MXASSET	ASSET	Sync			
2	ASSETNUM	SITEID	DESCRIPTION	STATUS	LOCATION	PARENT
3	T2010	FLEET	1997 Great Dane 4	NOT READY	CALGARY	
4	TRST152	FLEET	TRAILER STARTING CIRC	NOT READY	OILFLD	SPF01602
5	13143	BEDFORDXXX	Chain Wash Assembly	ACTIVE	PLANT-W1	
6	12510	BEDFORD	Brake System- Overhe	NOT READY	SHIPPING	12500
7	T-B13180	TEXAS	WASH BARREL	NOT READY	CLEAN	
8	61001	BEDFORD	Freightliner #61001	NOT READY	FLTGAR	
9						
10						
11						
12						
13						

Microsoft Excel  
Continuing syncing from row 5  
Yes No Cancel

When you have to load data that has some inconsistencies you can use another technique. Do a first pass to load all the correct rows and then handle the wrong rows with a second pass.

First of all set the 'Batch size' to 1 and the 'Stop on error' to False. Then add the object ID in the first column. For example if you are loading Assets you should add the [ASSETID] field.

If you run a Sync action, MxLoader will go through all the rows and set the object ID for the successfully imported rows. The 'bad' rows will have blank IDs.

	A	B	C	D	E	F	G	H
1	MXASSET	ASSET	Sync					
2	[ASSETUID]	ASSETNUM	SITEID	DESCRIPTION	STATUS	LOCATION	PARENT	
3	25	T2010	FLEET	1997 Great Dane 48ft Tr	NOT READY	CALGARY		
4	26	TRST152	FLEET	TRAILER STARTING CIRC	NOT READY	OILFLD	SPF01602	
5		13143	BEDFORDxxx	Chain Wash Assembly	ACTIVE	PLANT-W1		
6	28	12510	BEDFORD	Brake System- Overhe	NOT READY	SHIPPING	12500	
7	29	T-B13180	TEXAS	WASH BARREL	NOT READY	CLEAN		
8	90	61001	BEDFORD	Freightliner #61001	NOT READY	FLTGAR		
9								

If you now copy all the rows with blank IDs in a separate MxSheet and set the 'Stop on error' to True you can fix the inconsistencies for each row one by one and finish importing your data.

Another technique is to use the [ERROR] virtual attribute. If you add such attribute in the header row, MxLoader will fill the corresponding cells with the returned error. This, used in conjunction with the 'Stop on error' setting set to False, can be used to load large batch of data, collecting and handling errors separately.

	A	B	C	D	E	F
1	MXASSET	ASSET	Sync-AddChange			
2	ASSETNUM	SITEID	DESCRIPTION	STATUS	LOCATION	[ERROR]
3	1001	BEDFORD	Fire Extinguisher	ACTIVE	MOFLOOR1	
4	1002	BEDFORDxx	Fire Extinguisher	OPERATING	MOFLOOR2	BMXAA4153E - BEDFORDxx is not a valid site. Enter a valid Site value as defined in the Organization Application.
5	1003	BEDFORD	Fire Extinguisher	ACTIVE	MOFLOOR3	
6	1004	BEDFORD	Fire Extinguisher	ACTIVE	MOFLOOR4xx	BMXAA2659E - To Location MOFLOOR4XX is not a valid location.
7	1005	BEDFORD	Fire Extinguisher	ACTIVE	CONF400	

### 3.2.7 Excel Tips and Tricks

One of the great features of MxLoader is that it runs in Excel. It is great because:

- You probably already have asset data in Excel spreadsheets.
- You have plenty of features to manipulate existing data before loading it.

A useful technique is to have your reference data in one worksheet and reference it from a separate MxLoader worksheet. This allows to maintain your reference data in your original spreadsheet and use Excel formulas and drag-and-drop features to transform data into MxLoader sheets.

Let's make an example. You have the following asset data in your source spreadsheet.

Asset	Description	Location
AS_P100	PUMP 100	BLK1000
AS_P101	PUMP 101	BR210
AS_P102	PUMP 102	CSBRK571

The MxLoader template to load assets has different columns in different order. You also want to adjust the asset identifier and capitalize the description. Here is how the Maximo data should be.

ASSETNUM	SITEID	DESCRIPTION	STATUS	LOCATION
PUMP100	BEDFORD	Pump 100	OPERATING	BLK1000
PUMP101	BEDFORD	Pump 101	OPERATING	BR210
PUMP102	BEDFORD	Pump 102	OPERATING	CSBRK571

The formulas used to obtain such transformations are.

Column	Formula	Notes
A	=CONCATENATE("PUMP", RIGHT('RefData'!A2,3))	Concatenate the word 'PUMP' with the last 3 characters of the previous asset ID.
B	BEDFORD	Constant value.
C	=PROPER('RefData'!B2)	Capitalize the description.
D	OPERATING	Constant value.
E	= 'RefData'!C2	Copies the location

The most useful Excel formulas are:

- **VLOOKUP**: Looks at a value in one column, and finds its corresponding value on the same row in another column. Using VLOOKUP is similar to looking up a person's name in a telephone book to get a telephone number.

- **INDEX/MATCH:** is a good alternative to the VLOOKUP function. This [article](#) explains other alternatives and how to use them.
- **PROPER:** Capitalizes the first letter of each word in a text string. This can help with a list of names and addresses that was in all uppercase or lowercase.
- **LEFT, RIGHT, MID:** extract a substring (part of a text entry), of a length you specify, from text in a cell.
- **REPLACE:** replaces a sequence of characters in a string with another set of characters.
- **CONCATENATE:** The CONCATENATE function joins text strings into one.

---

### 3.2.8 Securing Maximo server

User's authentication and authorization management is important also for integration interfaces. Inbound and outbound communication with external systems should be authenticated too to prevent unwanted system access.

The TPAAE's Integration Framework provides several handlers for external systems: web (HTTP, WebServices, REST, OSLC), files, database interface tables, JMS queues, EJBs and command lines.

Maximo 7.6 manual has an [Integration security section](#) that discusses both authentication and authorization issues and configurations.

These are standard integration interfaces but other techniques must be also be analyzed if used.

It is important to notice that **Integration Framework web interfaces (HTTP, Web Service, REST) are not secured by default**. This could allow a skilled guy to have read/write access to all TPAAE's objects through HTTP calls. Follow the [Configuring J2EE security](#) page to enforce authentication of incoming requests.

Another critical aspect is the role of the MIF default user. When anonymous access is possible the user specified in **mxe.int.dfltuser** system property is used and it may be dangerous to have this user in an administrative security group.

## 4 MxLOADER TEMPLATES AND EXAMPLES

In this chapter are described some MxLoader templates together with small examples of how to use them. Sometimes they are related one each other.

1. Locations
2. Items
3. Tools
4. Assets or Rotating Assets
5. People
6. Labor
7. Companies
8. Users (if you are not using LDAP)

The samples will be based on the Maximo demo database that can also be viewed on the [Maximo Asset Management 7.5 Preview Site](#).

A small extract of the Maximo demo data has been used and is centered on the following locations and assets.

- Location: NEEDHAM - Needham Site
  - Location: BOILER - Boiler Room
  - Location: OFFICE - Office Building
  - Location: PLANT-P1 - Materials Processing Plant #1
    - Location: MTP100 - Materials Transfer Pipe
      - Asset: 26000 - Motor Controlled Valve
        - Asset: 26020 - 24 Volt-DC Motor

✓ All the examples in this doc are collected in separate Excel worksheets that can be [downloaded from the community](#).

## 4.1 ADMINISTRATION

### 4.1.1 People

People application (Administration > Resources > People) stores information about individuals, such as users, laborers, asset owners, and supervisors who receive workflow notifications.

The **People** template allows you to load *Person* records using the predefined **MXPERSON** Object Structure.

PERSONID	FIRSTNAME	LASTNAME	LOCATIONSITE	STATUS
ADAMS	Hank	Adams	BEDFORD	ACTIVE
HUNTER	John	Hunter		ACTIVE
WILSON	Mike	Wilson	BEDFORD	ACTIVE

### 4.1.2 Person Groups

The **Person groups (custom)** template allows you to load *Person Groups* using the custom **MXL\_PERGRP** Object Structure.

PERSONGROUP	PERSONGROUPTEAM. RESPARTY	PERSONGROUPTEAM. RESPARTYGROUP	PERSONGROUPTEAM. RESPARTYGROUPSEQ	PERSONGROUPTEAM. GROUPDEFAULT	PERSONGROUPTEAM. USEFORORG	PERSONGROUPTEAM. USEFORSITE
MAINT	HUNTER	HUNTER	1	1	~NULL~	~NULL~
MAINT	WILSON	WILSON	2	0	~NULL~	~NULL~

### 4.1.3 Labor

The **Labor** template allows you to load *Labor* records using the predefined **MXLABOR** Object Structure.

LABORCODE	ORGID	PERSONID	WORKSITE
WILSON	EAGLENA	WILSON	BEDFORD
SCHAFER	EAGLENA	SCHAFER	BEDFORD



---

#### 4.1.4 Classification Hierarchy

The **Classification hierarchy** template allows you to load classification hierarchy using the predefined **MXCLASSIFICATION** Object Structure.

The following example will load a small extract of the [UNSPSC classification codes](http://www.ibm.com/support/docview.wss?uid=swg21516924).

CLASSIFICATIONID	HIERARCHYPATH	DESCRIPTION	CLASSUSEWITH. OBJECTNAME	CLASSUSEWITH. OBJECTVALUE
26	26	Power Generation and Distribution Machinery and Accessories	ASSET	ASSET
2610	26 \ 2610	Power sources	ASSET	ASSET
261012	26 \ 2610 \ 261012	Electric direct current DC motors	ASSET	ASSET
40	40	Distribution and Conditioning Systems and Equipment and Components	ASSET	ASSET
4014	40 \ 4014	Fluid and gas distribution	ASSET	ASSET
401416	40 \ 4014 \ 401416	Valves	ASSET	ASSET
40141602	40 \ 4014 \ 401416 \ 40141602	Needle valves	ASSET	ASSET
40141604	40 \ 4014 \ 401416 \ 40141604	Safety valves	ASSET	ASSET
40141609	40 \ 4014 \ 401416 \ 40141609	Control valves	ASSET	ASSET
4015	40 \ 4015	Industrial pumps and compressors	ASSET	ASSET
401515	40 \ 4015 \ 401515	Pumps	ASSET	ASSET
40151501	40 \ 4015 \ 401515 \ 40151501	Air pumps	ASSET	ASSET
40151502	40 \ 4015 \ 401515 \ 40151502	Vacuum pumps	ASSET	ASSET
40151503	40 \ 4015 \ 401515 \ 40151503	Centrifugal pumps	ASSET	ASSET

<http://www.ibm.com/support/docview.wss?uid=swg21516924>

---

#### 4.1.5 Specifications Attributes

The **Specification Attributes** template allows you to load specifications attributes using the custom **MXL\_ASSETATTRIBUTE** Object Structure.

The following example will load few sample specifications attributes.

ASSETATTRID	DESCRIPTION	MEASUREUNITID	DATATYPE
MAX WGT	Max Weight Allowed	KG	NUMERIC
HEIGHT	Height	CM	NUMERIC
WIDTH	Width	CM	NUMERIC
DIAM	Pipe Diameter	CM	NUMERIC
OP_PRES	Operating Pressure	PSI	NUMERIC
OP_TEM	Operating Temperature	DEG C	NUMERIC
MAX_PRES	Maximum Allowable Pressure	PSI	NUMERIC

#### 4.1.6 Classification Attributes

The **Classification Hierarchy** template allows you to load a classification hierarchy using the predefined **MXCLASSIFICATION** Object Structure.

CLASSIFICATIONID	HIERARCHYPATH	CLASSSPEC. ASSETATTRID	CLASSSPEC. SECTION
261012	26 \ 2610 \ 261012	HEIGHT	~NULL~
261012	26 \ 2610 \ 261012	WIDTH	~NULL~
40151502	40 \ 4015 \ 401515 \ 40151502	OP_PRES	~NULL~
40151502	40 \ 4015 \ 401515 \ 40151502	OP_TEM	~NULL~

## 4.2 ASSETS

### 4.2.1 Assets

The **Assets** template can be used to load *Asset* records using the predefined **MXASSET** Object Structure.

The following example will create two assets. Note that the first one is a subassembly of the second.

- Location: MTP100 - Materials Transfer Pipe
  - Asset 26000 - Motor Controlled Valve
    - Asset: 26020 - 24 Volt-DC Motor

ASSETNUM	SITEID	DESCRIPTION	STATUS	LOCATION	PARENT
26000	BEDFORD	Motor Controlled Valve	OPERATING	MTP100	
26020	BEDFORD	24 Volt-DC Motor	OPERATING	MTP100	26000

### 4.2.2 Assets Spare Parts

The **Asset spare parts** template allows you to load spare parts associated to assets using the custom **MXL\_ASSETSP** Object Structure.

The following example will specify the rotating item for assets 26000 and 26020. For asset 26000 four spare parts will be created.

ASSETNUM	SITEID	ITEMNUM	SPAREPART. ITEMNUM
26000	BEDFORD	MCV-60	78354
26000	BEDFORD	MCV-60	S-546
26000	BEDFORD	MCV-60	VB-512
26000	BEDFORD	MCV-60	28149
26020	BEDFORD	MOTR-24	

Note that we don't have to load the entire set of attributes but just specify the primary columns (ASSETNUM, SITEID) with the additional fields we want to set. All the other information will remain unchanged.

### 4.2.3 Asset Classification and Attributes

Once the classification hierarchy has been loaded, it is possible to classify assets, items and other objects. The described procedure is for assets but can be used for other objects as well.

ASSETNUM	SITEID	HIERARCHYPATH	ASSETSPEC. ASSETATTRID	ASSETSPEC. ALNVALUE	ASSETSPEC. NUMVALUE	ASSETSPEC. SECTION	ASSETSPEC. LINEARASSETSPECID
26000	BEDFORD	40 \ 4014 \ 401416 \ 40141609	OP_PRES		20	~NULL~	0
26000	BEDFORD	40 \ 4014 \ 401416 \ 40141609	OP_TEM		60	~NULL~	0
26020	BEDFORD	26 \ 2610 \ 261012	HEIGHT		35	~NULL~	0
26020	BEDFORD	26 \ 2610 \ 261012	WIDTH		42	~NULL~	0

The following fields seem to be mandatory:

- ASSETSPEC.SECTION
- ASSETSPEC.LINEARASSETSPECID

#### 4.2.4 Locations

The **Locations** template allows you to load locations using the predefined **MXOPERLOC** Object Structure.

The following example will create five operating locations in the BEDFORD site.

LOCATION	SITEID	DESCRIPTION	STATUS	TYPE
BOILER	BEDFORD	Boiler Room	OPERATING	OPERATING
OFFICE	BEDFORD	Office Building	OPERATING	OPERATING
PLANT-P1	BEDFORD	Materials Processing Plant #1	OPERATING	OPERATING
MTP100	BEDFORD	Materials Transfer Pipe	OPERATING	OPERATING

#### 4.2.5 Locations Hierarchy

Locations can be grouped in hierarchical structures called *Systems*. Systems must be created manually. The root location must be also associated to the system manually.

The **Location hierarchy** template allows you to load locations hierarchy using the predefined **MXOPERLOC** Object Structure.

The example described hereafter created this location hierarchy:

- Location: NEEDHAM - Needham Site
  - Location: BOILER - Boiler Room
  - Location: OFFICE - Office Building
  - Location: PLANT-P1 - Materials Processing Plant #1
    - Location: MTP100 - Materials Transfer Pipe

LOCATION	SITEID	DESCRIPTION	STATUS	TYPE	PARENT	SYSTEMID
NEEDHAM	BEDFORD	Bedford plant	OPERATING	OPERATING		PRIMARY
BOILER	BEDFORD	Boiler Room	OPERATING	OPERATING	NEEDHAM	PRIMARY
OFFICE	BEDFORD	Office Building	OPERATING	OPERATING	NEEDHAM	PRIMARY
PLANT-P1	BEDFORD	Materials Processing Plant #1	OPERATING	OPERATING	NEEDHAM	PRIMARY
MTP100	BEDFORD	Materials Transfer Pipe	OPERATING	OPERATING	PLANT-P1	PRIMARY

Note that you need to manually create NEEDHAM location and set it as the PRIMARY system root location.

Unfortunately the location hierarchy cannot be set once the locations are already created using MXOPERLOC Object Structure. You have to delete previous locations and import them back using this information.

To update the location hierarchy for existing locations you have to update the LOCHIERARCHY table.

The **Location hierarchy (custom)** template allows you to update locations hierarchy using the custom **MXL\_LOCATION** Object Structure.

LOCATION	SITEID	DESCRIPTION	STATUS	TYPE	LOCHIERARCHY. PARENT	LOCHIERARCHY. SYSTEMID
NEEDHAM	BEDFORD	Bedford plant	OPERATING	OPERATING		PRIMARY
BOILER	BEDFORD	Boiler Room	OPERATING	OPERATING	NEEDHAM	PRIMARY
OFFICE	BEDFORD	Office Building	OPERATING	OPERATING	NEEDHAM	PRIMARY
PLANT-P1	BEDFORD	Materials Processing Plant #1	OPERATING	OPERATING	NEEDHAM	PRIMARY
MTP100	BEDFORD	Materials Transfer Pipe	OPERATING	OPERATING	PLANT-P1	PRIMARY

On old version of Maximo it is important to correctly set the LOCHIERARCHY.CHILDREN attribute to avoid the problem described in [TechNote 1613050](#).

## 4.2.6 Failure Codes

To load failure codes you first have to load the failure codes descriptions and then the failure class hierarchy.

The **Failure Codes** template allows you to load failure codes using the custom **MXL\_FAILURECODE** Object Structure.

The following example will load one failure class, two problems, two causes and three remedies.

FAILURECODE	ORGID	DESCRIPTION	LANGCODE
CL1	EAGLENA	Test Class 1	EN
P1	EAGLENA	Test Problem 1	EN
P2	EAGLENA	Test Problem 2	EN
C1	EAGLENA	Test Cause 1	EN
C2	EAGLENA	Test Cause 2	EN
R1	EAGLENA	Test Remedy 1	EN
R2	EAGLENA	Test Remedy 2	EN
R3	EAGLENA	Test Remedy 3	EN

The **Failure Hierarchy** template allows you to load failure class hierarchy using the custom **MXL\_FAILURELIST** Object Structure.

The following example loads a small failure hierarchy using the failure codes defined previously.

ORGID	[CLASSID]	CLASS	[PROBLEMID]	PROBLEM	[CAUSEID]	CAUSE	[REMEDYID]	REMEDY
EAGLENA		CL1		P1		C1		R1
EAGLENA		CL1		P1		C2		R2
EAGLENA		CL1		P1		C2		R3
EAGLENA		CL1		P2		C1		R2
EAGLENA		CL1		P2		C2		R1

Note how the failure hierarchy is specified in a flat table.

The object IDs in the square brackets will be generated dynamically by MxLoader during the import process. Do not enter those IDs manually.

This is how this failure class will look in Maximo.

List View

Failure Codes

Failure Class:

CL1

Test Class:1

Organization:

EAGLENA

Attachments

Problems

Filter

1 - 2 of 2

Failure Code	Description
P1	Test Problem 1
P2	Test Problem 2

New Row

Causes for P1

Filter

1 - 2 of 2

Failure Code	Description
C1	Test Cause 1
C2	Test Cause 2

New Row

Remedies for C1

Filter

1 - 1 of 1

Failure Code	Description
R1	Test Remedy 1

New Row

Note that the **Failure Hierarchy** template does not allow querying existing failure hierarchy. This can be a limitation if you want to copy a failure class hierarchy from one organization to another or from one server to another. To extract an existing failure hierarchy you have use an SQL query like this.

```
SELECT f.orgid, f.failurelist classid, f.failurecode class,
       p.failurelist problemid, p.failurecode problem,
       c.failurelist causeid, c.failurecode cause,
       r.failurelist remedyid, r.failurecode remedy
FROM failurelist f
LEFT OUTER JOIN failurelist p ON p.parent=f.failurelist AND p.type='PROBLEM'
LEFT OUTER JOIN failurelist c ON c.parent=p.failurelist AND c.type='CAUSE'
LEFT OUTER JOIN failurelist r ON r.parent=c.failurelist AND r.type='REMEDY'
WHERE f.parent IS NULL
ORDER BY f.failurecode, p.failurecode, c.failurecode, r.failurecode;
```

MxLoader 4.7.5 has introduced the capability to upload custom levels of failure hierarchy. For example, if you have a custom level called **SYSTEM**, you can simply add it to the columns and MxLoader will handle it.

ORGID	[CLASSID]	CLASS	[SYSTEMID]	SYSTEM	[PROBLEMID]	PROBLEM	[CAUSEID]	CAUSE	[REMEDYID]	REMEDY

## 4.3 FINANCIAL

### 4.3.1 Currency Codes and Exchange Rates

The **Currency Codes** template allows you to load currency codes using the custom **MXL\_CURRENCY** Object Structure.

CURRENCYCODE	DESCRIPTION	ACTIVE
AUD	Australian Dollar	1
CAD	Canadian Dollar	1
EUR	Euro	1
GBP	British Pound Sterling	1
USD	United States of America Dollar	1
YEN	Japanese Yen	1

The **Exchange Rates** template allows you to load currency exchange rates using the custom **MXL\_EXCGRATES** Object Structure.

ORGID	CURRENCYCODE	CURRENCYCODETO	EXCHANGERATE	ACTIVEDATE	EXPIREDATE
EAGLENA	YEN	USD	0.0095800	2000-01-01 23:00:00	2022-12-31 23:00:00
EAGLENA	EUR	USD	1.0713000	2000-01-01 23:00:00	2022-12-31 23:00:00



## 4.4 INVENTORY

### 4.4.1 Items

The **Items** template allows you to load inventory items using the predefined **MXITEM** Object Structure.

The following example will create four items in the ITEMSET1 item set. Note that the first two items are defined as 'rotating' while the other two are defined as 'non rotating'.

ITEMNUM	ITEMSETID	DESCRIPTION	STATUS	ROTATING
MCV-60	ITEMSET1	Motor Controlled Valve	ACTIVE	1
MOTR-24	ITEMSET1	24 Volt-DC Motor	ACTIVE	1
78354	ITEMSET1	Motor Support Brackets	ACTIVE	0
S-546	ITEMSET1	Stem Coupling	ACTIVE	0

✓ Use 'Sync-AddChange' operation to avoid deleting entries in the ITEMORGINFO table.

### 4.4.2 Tools

The **Tools** template allows you to load inventory tools using the predefined **MXTOOLITEM** Object Structure. Here is an example.

ITEMNUM	ITEMSETID	DESCRIPTION	STATUS	ROTATING	ISSUEUNIT
PUNCH	ITEM1	BLACK & DECKER 24C49487 PNEU. PUNCH	ACTIVE	0	EACH
TORQUE	ITEM1	TORQUE WRENCH-20 FT-LB CAPACITY	ACTIVE	0	EACH

### 4.4.3 Item/Organization association

The **ItemOrg** template allows you to activate items on specific organization using the predefined **MXITEM** Object Structure.

✓ NOTE: Use 'Sync-Change' operation to avoid deleting entries in the ITEMORGINFO table.

The following example will activates item 28149 in the EAGLENA organization and set the same item as 'non stockable' for EAGLEUK organization.

ITEMNUM	ITEMSETID	ITEMMORGINFO.ORGID	ITEMMORGINFO.STATUS	ITEMMORGINFO.CATEGORY
28149	SET1	EAGLENA	ACTIVE	STK
28149	SET1	EAGLEUK	PENDING	NS

For details on stock categories refer to [IBM TechNote 1651038](#).

#### 4.4.4 Storerooms

The **Storerooms** template allows you to load storerooms using the predefined **MXSTORELOC** Object Structure.

The following example creates three storerooms.

LOCATION	SITEID	DESCRIPTION	STATUS	ISDEFAULT	USEINPOPR
CENTRAL	BEDFORD	Central Storeroom	OPERATING	1	1
PKG	BEDFORD	Packaging Dept. Storeroom	OPERATING	0	1
HORN	BEDFORD	Christine Horn	OPERATING	0	0

#### 4.4.5 Inventory

After having created Items and Storerooms you have to associate them to specify in which storeroom each of the items can be stored.

The **Inventory** template allows you to assign items to inventory using the predefined **MXINVENTORY** Object Structure.

The following example will load items in the CENTRAL storeroom.

ITEMNUM	ITEMSETID	SITEID	LOCATION	ISSUEUNIT
78354	SET1	BEDFORD	CENTRAL	EACH
S-546	SET1	BEDFORD	CENTRAL	EACH

#### 4.4.6 Inventory Balance

The **Inventory balances** template allows you to load inventory balances using the predefined **MXINVBAL** Object Structure.

The following example will load actual inventory balances for two items in the CENTRAL storeroom.

ITEMNUM	ITEMSETID	SITEID	LOCATION	BINNUM	CURBAL
78354	SET1	BEDFORD	CENTRAL	A112	0
S-546	SET1	BEDFORD	CENTRAL	A113	8

#### 4.4.7 Units of Measure

The ***Units of Measure*** template allows you to load units of measures using the custom ***MXL\_MEASUREUNIT*** Object Structure.

The following example will loads few sample units of measure.

MEASUREUNITID	DESCRIPTION	ABBREVIATION
EACH	Each	
BOX	Box	
DEG F	Degree Fahrenheit (temperature)	°F
DEG C	Degree Celsius (temperature)	°C
LTRS	Liters (volume)	l
SECS	Seconds (time)	s
HOURS	Hours (time)	h
CM	Centimeters (length)	cm
METER	Meter (length)	m
KM	Kilometer (length)	Km
MILES	Miles (length)	Mi
MPH	Miles per hour (speed)	
PSI	Pounds per square inch (pressure)	PSI
BAR	Bar (pressure)	BAR
RPM	Revolutions per minute	RPM
VOLTS	Volts (voltage)	V

#### 4.4.8 Units of Measure Conversion

The ***Units of Measure conversions*** template allows you to load units of measure conversion factors using the custom ***MXL\_CONVERSION*** Object Structure.

The following example will loads few sample units of measure conversion factors.

<u>FROMMEASUREUNIT</u>	<u>TOMEASUREUNIT</u>	CONVERSION
MILES	FEET	5280.00
KM	METER	1000.00
FEET	CM	30.48

#### 4.4.9 Commodities

The **Commodities** template allows you to load commodity codes using the custom **MXL\_COMMODITIES** Object Structure.

COMMODITY	ITEMSETID	DESCRIPTION	PARENT	ISSERVICE
26101500	SET1	Engines		0
26101501	SET1	Hydraulic engines	26101500	0
26101503	SET1	Gas engines	26101500	0
26101504	SET1	Diesel engines	26101500	0

## 4.5 PURCHASING

### 4.5.1 Companies

The **Companies** template allows you to load Companies using the custom **MXL\_COMPANIES** Object Structure.

COMPANY	ORGID	NAME	TYPE
EVERGR	EAGLENA	Evergreen Lawn Maintenance	V
GST	EAGLENA	Gast Pumps	M

- ✓ You may need to activate the 'Automatically Add Companies to Company Master' option in your Organization's Company Set (Administration > Sets application).

## 4.6 WORK

### 4.6.1 Work Orders

The **Work Orders** template allows you to load work orders using the predefined **MXWO** Object Structure.

WONUM	SITEID	DESCRIPTION	STATUS	ASSETNUM	LOCATION
1000	BEDFORD	Relocate Guard Rails Around Compressor	WAPPR	11300	BR300
1022	BEDFORD	Electric Cart Tune-Up	WSCH	12300	SHIPPING
2005	BEDFORD	Check-out Leaking	WAPPR	13145	BPM3100

## 4.7 PLANNING

### 4.7.1 Job Plans

The **JobPlans** template allows you to load Job Plans using the custom **MXL\_JOBPLAN** Object Structure.

Refer to [this article](#) for more details.

## 4.7.2 Routes

The **Routes** template allows you to load Routes using the custom **MXL\_ROUTES** Object Structure.

<b>ROUTE</b>	<b>SITEID</b>	<b>DESCRIPTION</b>	<b>ROUTESTOPSBECOME</b>	<b>ROUTE_STOP. ROUTESTOPID</b>	<b>ROUTE_STOP. ASSETNUM</b>
1002	BEDFORD	Inspect Fire Extinguishers	MULTI	1	1002
1002	BEDFORD	Inspect Fire Extinguishers	MULTI	2	1003
1002	BEDFORD	Inspect Fire Extinguishers	MULTI	3	1004
1002	BEDFORD	Inspect Fire Extinguishers	MULTI	4	1005
1002	BEDFORD	Inspect Fire Extinguishers	MULTI	5	1006
1002	BEDFORD	Inspect Fire Extinguishers	MULTI	6	1007

## 4.8 SECURITY

### 4.8.1 Users

The **Users** template allows you to load users using the predefined **MXPERUSER** Object Structure.

In order to be able to set passwords, all the attributes related to the password on MAXUSER object must be set as "not restricted".

- Go to > System Configuration > Integration > Object Structures > MXPERUSER
- Action Menu > Inbound Setting Restrictions > MAXUSER
- For PASSWORDINPUT, PASSWORDCHECK, EMAILPSWD, FORCEEXPIRATION attributes check the box in the Override column, then clear the box in the Restricted column.

Inbound Setting Restrictions

Object Structure: MXPERUSER Person User Definition

Source MBOs for MXPERUSER

Object: PERSON, MAXUSER

Inbound Setting Restrictions For MBO MAXUSER

Field	Restricted?	Override?
DEFAULTREPFAC	<input type="checkbox"/>	<input type="checkbox"/>
DATABASEUSERID	<input type="checkbox"/>	<input type="checkbox"/>
FORCEEXPIRATION	<input type="checkbox"/>	<input checked="" type="checkbox"/>
EMAILPSWD	<input type="checkbox"/>	<input checked="" type="checkbox"/>
PASSWORDCHECK	<input type="checkbox"/>	<input checked="" type="checkbox"/>
PASSWORDINPUT	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Here is an example.

PERSONID	FIRSTNAME	LASTNAME	LOCATIONSITE	STATUS	MAXUSER. USERID	MAXUSER. PASSWORDINPUT	MAXUSER. PASSWORDCHECK	MAXUSER. EMAILPSWD	MAXUSER. FORCEEXPIRATION
BPORTALU	Bruno	Portaluri	BEDFORD	ACTIVE	bportalu	Pwd123	Pwd123	0	0

---

## 4.8.2 Users/Groups Association

The ***Users/Group*** template allows you to load associations between Users and Security Groups using the custom ***MXL\_USERGRP*** Object Structure.

USERID	GROUPUSER.GROUPNAME
WILSON	STDSVCMGR
WILSON	TOOLMGR
WILSON	ITEMMGR

- ✓ Use 'Sync-AddChange' operation to avoid deleting existing associations.