



Excel Template

11.2

February 2026

DOCUMENT ACCESS

Public

DISCLAIMER

The contents of this document are under copyright of Critical Manufacturing S.A. It is released on condition that it shall not be copied in whole, in part or otherwise reproduced (whether by photographic, or any other method) and the contents therefore shall not be divulged to any person other than that of the addressee (save to other authorized offices of his organization having need to know such contents, for the purpose for which disclosure is made) without prior written consent of submitting company.



Excel Template

In this section we will run through the Excel template and the different ways you can navigate through it and understand the references and information contained in the template, as well as how to use it to your advantage and thus save time and effort while loading a **Master Data Package** into the MES.

Helper Sheets

The first worksheets you see in the Excel template are helper sheets that give you information and are also used as configuration for the entire template. To understand what information they hold and what they are used for, see [Help Sheets](#).

Explaining the Format

The different names and types of the sheets contained in the Excel template are described below:

Prefix	Type
No prefix	Sub-sheet of previous prefixed sheet
<DM>	Dynamic Model
<SM>	Static Model
<GT>	Generic Table
<ST>	Smart Table
<UP>	Update Model
<LOOKUP>	Reference for the longer name. 

Note

An example for `<UP>` could be the following:

Name	Team	ShiftPlan	Workgroup	Position
Operator_2	Team-E	Assembly Preparation	WeighAndDispense	WeighAndDispense Certification
Operator_3	Team-E	Assembly Preparation	Mixing & Injection	Mixing Certification

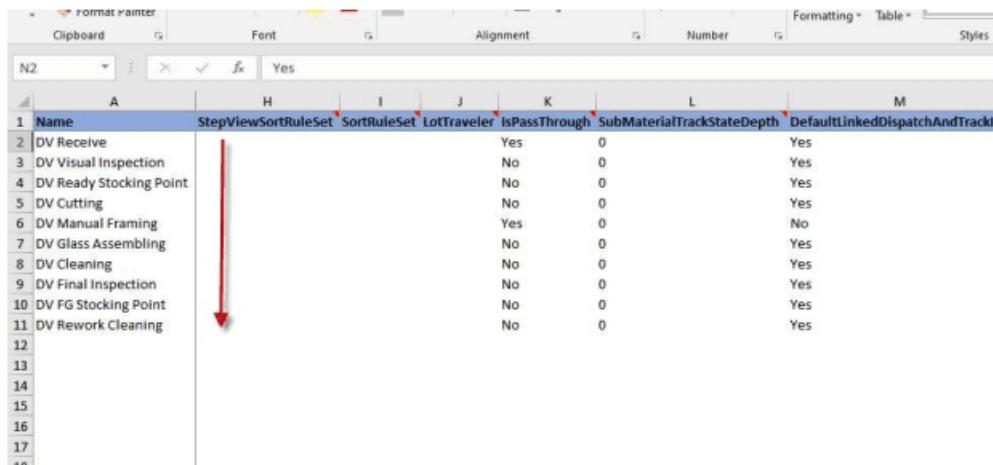
< > ... <UP>EmployeeShiftDetails

This tab is used to assign employees to a Shift Plan and then to a Workgroup of the Shift Plan.

Info

See [Help Sheets](#) for more information on the `<LOOKUP>`.

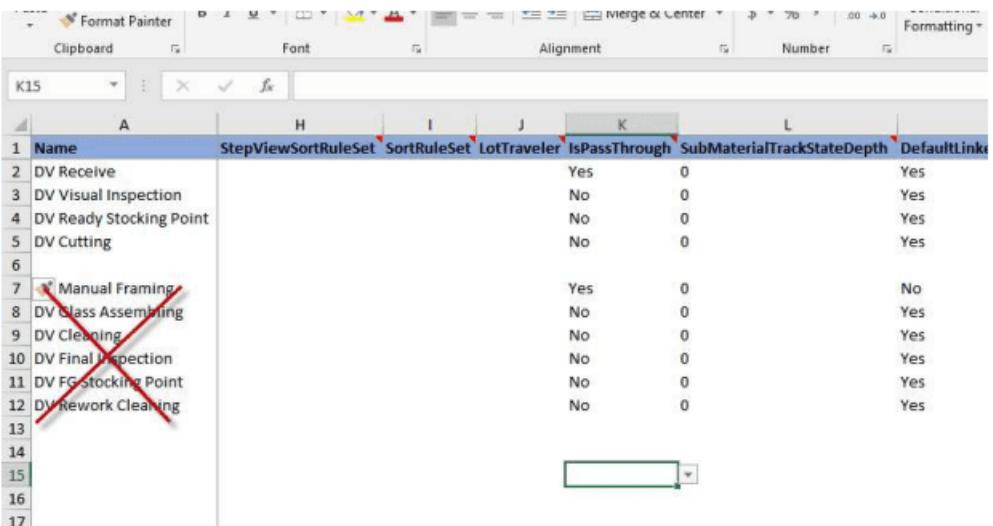
The system reads the Excel file in the order that the worksheets are placed in the file, starting from the left-most sheet and moving rightward. Each sheet is read from left to right and from top to bottom, creating fields as a linear FIFO (First-In-First-Out) sequence.



	A	H	I	J	K	L	M
1	Name	StepViewSortRuleSet	SortRuleSet	LotTraveler	IsPassThrough	SubMaterialTrackStateDepth	DefaultLinkedDispatchAndTrack
2	DV Receive				Yes	0	Yes
3	DV Visual Inspection				No	0	Yes
4	DV Ready Stocking Point				No	0	Yes
5	DV Cutting				No	0	Yes
6	DV Manual Framing				Yes	0	No
7	DV Glass Assembling				No	0	Yes
8	DV Cleaning				No	0	Yes
9	DV Final Inspection				No	0	Yes
10	DV FG Stocking Point				No	0	Yes
11	DV Rework Cleaning				No	0	Yes

Warning

The system will stop reading if a blank line appears. You can look at each sheet as a specific entity type, with each column representing an entity property or relation and each row representing a new entity.



	A	H	I	J	K	L	M
1	Name	StepViewSortRuleSet	SortRuleSet	LotTraveler	IsPassThrough	SubMaterialTrackStateDepth	DefaultLinked
2	DV Receive				Yes	0	Yes
3	DV Visual Inspection				No	0	Yes
4	DV Ready Stocking Point				No	0	Yes
5	DV Cutting				No	0	Yes
6							
7	DV Manual Framing				Yes	0	No
8	DV Glass Assembling				No	0	Yes
9	DV Cleaning				No	0	Yes
10	DV Final Inspection				No	0	Yes
11	DV FG Stocking Point				No	0	Yes
12	DV Rework Cleaning				No	0	Yes

The name of the Excel sheets must match the MES entities being loaded and the name of the columns must match the entity properties exactly. When processing the data, the system will check if the object already exists, thus creating or updating it accordingly.

Note

The template does not have all the objects and fields that are available in the UI. You can, however, add more entity types (sheets) and entity properties (columns), provided they are available in the UI and exactly match the name of the entity and respective properties.

Versioned Entities

In the case of versioned entities, a new version is always created when loaded into the system. **Change Sets** are managed automatically by the tool so there is no need to create them manually either in the Excel template or in the GUI. However, the approval process for any versioned object can only be performed using the GUI and you can access any **Change Sets** currently under approval in the **Change Sets**.

For more information on loading entity revisions and versions, see [Revision Configuration](#).

Modeling Sequence

It is highly recommended to follow an appropriate modeling sequence since there are a lot of interdependent fields that have strict rules of precedence.

Flow/Step example

A **Flow** is composed of **Flow Items** of type **Step** and **Flow**. When a **Flow** contains **Flow Items** of type **Flow**, they are denoted as **Child Flows** of the **Parent Flow** for clarity. To create a **Flow** structure, you should follow the sequence below:

1. Create the **Steps**.

A	Name	B	C	D	E	F	G	H	I	J	K	L	M
1		Description	Type	IsTemplate	DataGroup	ProcessingType	DisplayOrder	StepViewSortRuleSet	SortRuleSet	LotTraveler	MaterialLabel	IsPassThrough	IsPackingStep
2	Step_1		Process			Process						Yes	
3	Step_2		Process			Process						No	
4	Step_3		Process			Process						No	
5	Step_4		Process			Process						No	
6													
7													
8													
9													
10													
11													
12													
13													
14													
15													
16													
17													
18													
19													
20													
21													
22													
23													
24													

2. Create the **Flows**.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Name	Revision	Description	Type	DataGroup	IsNonSequentialBlock	IsAlternate	IsLineFlow	AlternateFlowSelectionType	IsEnabled						
2	Flow_A			Sequential		No	No	No		Yes						
3	Flow_1			Sequential		No	No	No		Yes						
4	Flow_2			Sequential		No	No	No		Yes						
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																
17																
18																
19																
20																
21																
22																
23																
24																

3. Utilize the **FlowItems** sheet to create the desired **Flow** structure.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Flow	Type	Target	IsOptional	Reworks	IsLine	IsSkippable	LineFlows	LogicalName	OnEnterRule	OnExitRule	ConditionType	ConditionExpression	ConditionRule	ConditionSamplingStep
2	Flow_A	Step	Step_1	No		No	No								
3	Flow_A	Step	Step_2	No		No	No								
4	Flow_A	Step	Step_3	No		No	No								
5	Flow_1	Flow	Flow_A	No		No	No								
6	Flow_1	Step	Step_4	No		No	No								
7	Flow_2	Step	Step_5	No		No	No								
8	Flow_2	Step	Step_6	No		No	No								
9															
10															
11															
12															
13															
14															
15															
16															
17															
18															
19															
20															
21															
22															
23															
24															

Resource/Service example

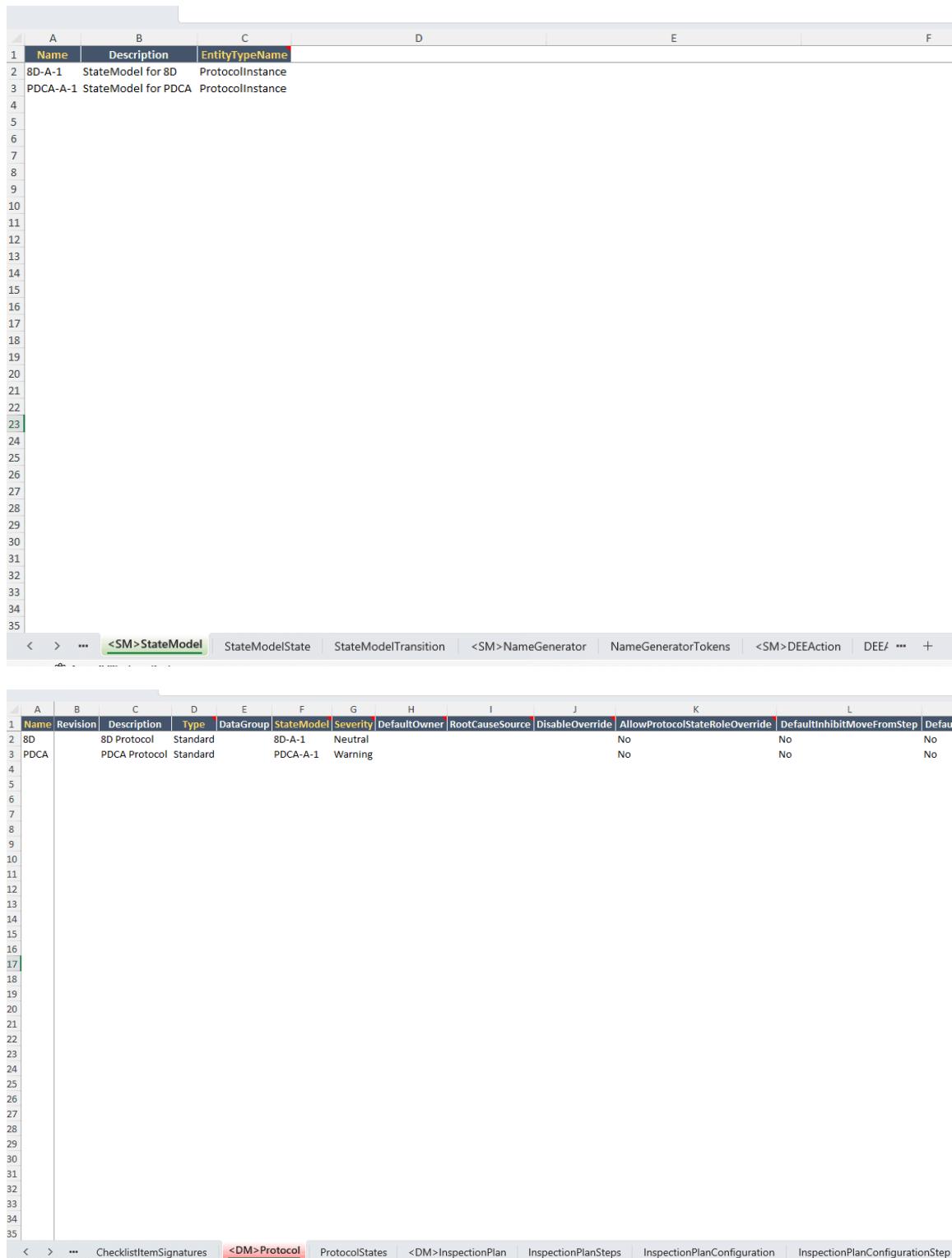
In another quick example, the **Service** must be created before the **Resource** creation and only after it is possible to fulfill the **ResourceService** sheet.

	A	B	C	D	E	F	G	H	I
1	SourceEntity	TargetEntity	Priority	IsEnabled					
2	Resource_1	Service_Resource_1		Yes					
3	Resource_2	Service_Resource_2		Yes					
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20									
21									
22									
23									
24									

<DM>ResourceMeasurementCapability <DM>ResourceService <DM>DataCollection DataCollectionParameters <DM>Dat

State Model/Protocol example

In the case of **Protocols**, you must create the **State Model** and load both the **State Model** and the **Protocol** at the same time.



The screenshot shows two tables side-by-side, each with a header row and data rows. The first table is titled 'State Model' and the second is titled 'Protocol'.

State Model Table (Top):

A	B	C	D	E	F
1	Name	Description	EntityType		
2	8D-A-1	StateModel for 8D	ProtocolInstance		
3	PDCA-A-1	StateModel for PDCA	ProtocolInstance		
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					
27					
28					
29					
30					
31					
32					
33					
34					
35					

Protocol Table (Bottom):

A	B	C	D	E	F	G	H	I	J	K	L	M	
1	Name	Revision	Description	Type	DataGroup	StateModel	Severity	DefaultOwner	RootCauseSource	DisableOverride	AllowProtocolStateRoleOverride	DefaultInhibitMoveFromStep	Default
2	8D	8D Protocol	Standard	Standard	8D-A-1	Neutral				No	No	No	No
3	PDCA	PDCA Protocol	Standard	Standard	PDCA-A-1	Warning				No	No	No	No
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													
14													
15													
16													
17													
18													
19													
20													
21													
22													
23													
24													
25													
26													
27													
28													
29													
30													
31													
32													
33													
34													
35													

Smart Table example

A custom Smart Table is often created through the CM MES interface, but it is also possible to create it through the Master Data Package. To create a Smart Table, you should follow the steps below:

1. Create the custom Smart Table using the Excel template `<ST>SmartTable` sheet:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Name	Description	DataGroup	IsHistoryEnabled	HistoryRetentionTime	IsRepeatedValuesAllowed	HasChangeControl	PreValidationRule	PostValidationRule					
2	CustomMatChecklistContext	Used to resolve the checklist for the specific context		Yes	180	No		Yes						
3														
4														
5														
6														
7														
8														
9														
10														
11														
12														
13														
14														
15														
16														
17														
18														
19														
20														
21														
22														
23														
24														
25														
26														
27														
28														
29														
30														
31														
32														
33														
34														
35														
36														
37														

2. Populate the required Properties for the custom Smart Table using the Excel template

`SmartTableProperties` sheet:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	SmartTable	Name	Description	Position	Key	Indexed	EntityType	ReferenceName	ScaleType	Size	Precision	ValidationRange	ValidationRegex	ResolutionMode					
2	CustomMatChecklistContext	Step	Reference to a Step	1	Yes	No	Entity	Step.Name	NVarChar	256									
3	CustomMatChecklistContext	LogicalFlowPath	The Logical Flow Path	2	Yes	No	None		NVarChar	512									
4	CustomMatChecklistContext	Operation	Operation Name	3	Yes	Yes	Yes	LookupValue	MaterialChecklistOperations	NVarChar	256								
5	CustomMatChecklistContext	Product	Product Name	4	Yes	No	Yes	EntityDefinition	Product.Name	NVarChar	256								
6	CustomMatChecklistContext	ProductGroup	ProductGroup Name	5	Yes	No	Yes	EntityDefinition	ProductGroup.Name	NVarChar	256								
7	CustomMatChecklistContext	Flow	Flow Name	6	Yes	No	Yes	EntityDefinition	Flow.Name	NVarChar	256								
8	CustomMatChecklistContext	Material	Material Name	7	Yes	No	Yes	EntityDefinition	Material.Name	NVarChar	256								
9	CustomMatChecklistContext	MaterialType	Material Type	8	Yes	No	Yes	LookupValue	MaterialType	NVarChar	256								
10	CustomMatChecklistContext	Resource	Resource Reference	9	Yes	No	Yes	EntityType	Resource.Name	NVarChar	256								
11	CustomMatChecklistContext	ResourceType	ResourceType Name	10	Yes	No	Yes	LookupValue	ResourceResourceType	NVarChar	256								
12	CustomMatChecklistContext	Model	Model Name	11	Yes	No	Yes	LookupValue	ResourceModel	NVarChar	256								
13	CustomMatChecklistContext	Recipe	Recipe Name	12	Yes	No	Yes	EntityDefinition	Recipe.Name	NVarChar	256								
14	CustomMatChecklistContext	Checklist	Checklist Name	13	No	Yes	No	EntityDefinition	Checklist.Name	NVarChar	256								
15																			
16																			
17																			
18																			
19																			
20																			
21																			
22																			
23																			
24																			
25																			
26																			
27																			
28																			
29																			
30																			
31																			
32																			
33																			
34																			
35																			
36																			
37																			

3. Define the required Precedence Keys for the custom Smart Table using the Excel template

`SmartTablePrecedenceKeys` sheet:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	SmartTable	PrecedenceKey1	PrecedenceKey2	PrecedenceKey3	PrecedenceKey4	PrecedenceKey5	PrecedenceKey6								
2	CustomMatChecklistContext	Step+Operation+Material	Step+Operation+Recipe	Step+LogicalFlowPath+Operation+Product+Flow+MaterialType	Step+Operation+Product+Flow+MaterialType	Step+Operation+ProductGroup+Flow+MaterialType	Step+LogicalFlowPath+Operation+Product	Step+Op							
3															
4															
5															
6															
7															
8															
9															
10															
11															
12															
13															
14															
15															
16															
17															
18															
19															
20															
21															
22															
23															
24															
25															
26															
27															
28															
29															
30															
31															
32															
33															
34															
35															
36															
37															

4. If data is to be uploaded directly into the Smart Table via Master Data, create a new Excel sheet named `<ST>ST Name`. The column names in this sheet must correspond to the properties defined in Step 2:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	Stop	LogicalKeyPath	Operation	Product	ProductGroup	Flow	Material	MaterialType	Resource	ResourceType	Model	Recipe	Checklist									
2	Inspection		After Dry		Trackin								CHK Measurement After Dry									
3																						
4																						
5																						
6																						
7																						
8																						
9																						
10																						
11																						
12																						
13																						
14																						
15																						
16																						
17																						
18																						
19																						
20																						
21																						
22																						
23																						
24																						
25																						
26																						
27																						
28																						
29																						
30																						
31																						
32																						
33																						
34																						
35																						
36																						
37																						

Moreover, because the fields to fill in the Master Data and in CM MES are the same, for more information on how to create a custom Smart Table, see [Smart Tables](#) in the Critical Manufacturing Documentation Portal.

In relation to the creation of custom Generic Tables using Master Data, you should follow steps 1, 2, and 4 above as the creation process is similar.

For Generic Tables, you can find more information in [Generic Tables](#), and your Excel template sheets would be as in the following example:

<GT>GenericTable | GenericTableProperties | <GT>GT Name

Relations

Most of the relations between objects are added automatically when two objects require contextual sharing of information due to their current operation state. You can manually add relations between objects by using the **EntityType** sheet and filling in the **SourceEntity** and the specific **TargetEntity**.

A	B	C	D	E	F	G	H	I	J	
1	Name	Description	IsRelation	SourceEntity	SourceAccessLevel	TargetEntity	TargetAccessLevel	RelationLock	IsVersioned	AllowImplicitChangeSet
2	Example_1		Yes	Resource	0	Employee	0	LockNone	Yes	No
3										
4										
5										
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										
21										

<SM>LocalizedMessageKey | <SM>EntityType | <SM>EntityTypeProperty | <SM>StateModel | StateModelState | StateModelTransition

Note

If you want to create a relation between versioned entities, use the **EntityTypeProperty** sheet to define whether the Source or Target entities belong to the version, to the definition or to the complete entity. The **Reference Type** field in that sheet defines that relationship and has three possible options:

- EntityDefinition
- EntityVersion
- Entity

Attributes

If you want to add specific attributes to an entity, you must create the attribute in the **EntityTypeProperty** sheet and then add the columns with the attribute names in the target Entity.

A	B	C	D	E	F	G	H	I	J	K	L	M	N
EntityType	Name	Description	PropertyType	DataGroup	IsEnabled	ReferenceType	ReferenceName	ScalarType	ScalarSize	ScalarPrecision	DefaultValue	IsArray	CopyOnClone
Resource	WarrantyStart		Attribute		Yes	None		DateTime	0			No	No
Resource	WarrantyEnd		Attribute		Yes	None		DateTime	0			No	No
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													
14													
15													
16													
17													
18													
19													
20													
21													
22													
23													
24													

A	B	DE	DF	DG	DH	DI	DJ
Name	Description	AllowEmployeeCheckOutOnCheckIn	LogCheckInActivity	WarrantyStart	WarrantyEnd		
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							
21							
22							
23							
24							



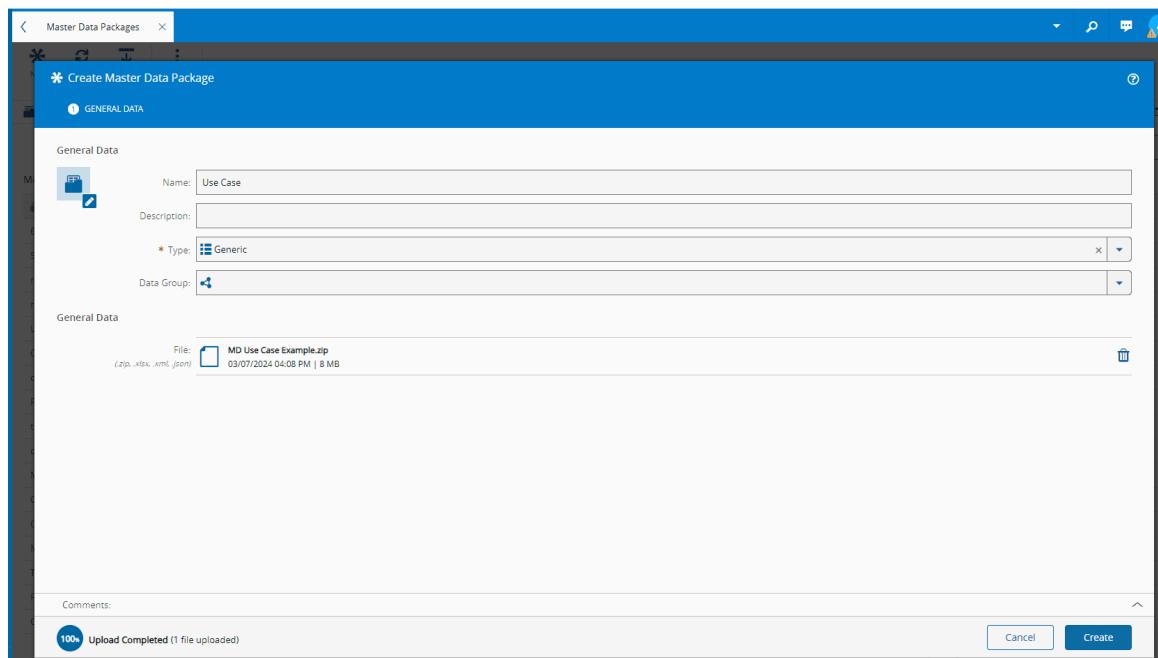
⚠ Warning

The empty spaces in the entity names are taken into consideration and can cause failures when loading the **Master Data Package**. Make sure you trim all leading and lagging spaces from the labels of any data element.

External Files

You can upload specific types of external files that will be included in the **Master Data Package**. They must be specified manually and included in a `.zip` file that will be used to upload the data into the system:

- Images - used in **Checklists**, **Data Collections** and Notes in Future Actions, among others.
- Objects - system objects like **Queries**, **Printable Documents** or **FabLives**, among others.
- Documents - files can be associated to most entity types, like PDFs or other files that can be displayed in the system.



To reference the files in a **Checklist**, you must delimit the file name and the specific format with `[[` and `]]`, like the example below:

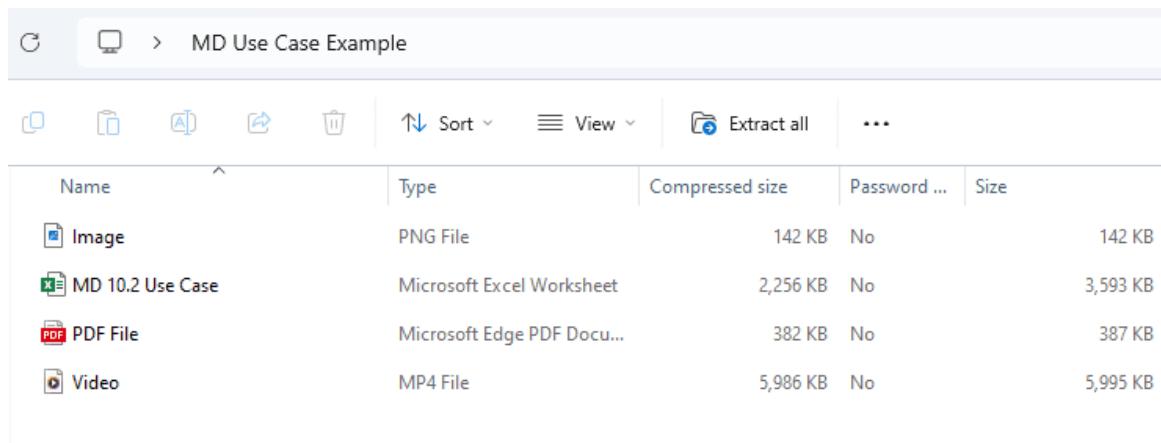


	A	B	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Checklist	Name	Group	DocumentationURL	ActivityType	TrackingType	ValidFrom	ValidTo	IsFloating	IsOptional	Rule	Instruction	DiagramFile	DiagramFileAnnotation		
2	Checklist_1	Item_1			ManualTask	End			No	No		[[Image.PNG]]				
3																
4																
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																
17																
18																
19																
20																
21																
22																
23																
24																
25																
26																

For **Document** loading the user must fill the appropriate values for `RelativeFileLocation` and `Filename`, the latter with the name of the object that will be included in the `.zip` file.

	A	B	C	D	E	F	G	H	I	J
1	Name	Revision	Description	Type	DataGroup	ContentStorageType	RelativeFileLocation	Filename	ContentURL	ChangeDescription
2	Procedure Video			Generic		Internal	Video.mp4	Video.mp4		
3	Procedure PDF File			Generic		Internal	PDF File.pdf	PDF File.pdf		
4										
5										
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										
17										
18										
19										
20										
21										
22										
23										
24										
25										
26										

The content of a `.zip` file that reflects the model shown above would look like this:



The screenshot shows a file explorer window with the following details:

Header: MD Use Case Example

Toolbar: Includes icons for Back, Forward, Refresh, Copy, Paste, Cut, Delete, Sort, View, Extract all, and a three-dot menu.

Table:

Name	Type	Compressed size	Password ...	Size
Image	PNG File	142 KB	No	142 KB
MD 10.2 Use Case	Microsoft Excel Worksheet	2,256 KB	No	3,593 KB
PDF File	Microsoft Edge PDF Docu...	382 KB	No	387 KB
Video	MP4 File	5,986 KB	No	5,995 KB



Legal Information

Disclaimer

The information contained in this document represents the current view of Critical Manufacturing on the issues discussed as of the date of publication. Because Critical Manufacturing must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Critical Manufacturing, and Critical Manufacturing cannot guarantee the accuracy of any information presented after the date of publication. This document is for informational purposes only.

Critical Manufacturing makes no warranties, express, implied or statutory, as to the information herein contained.

Confidentiality Notice

All materials and information included herein are being provided by Critical Manufacturing to its Customer solely for Customer internal use for its business purposes. Critical Manufacturing retains all rights, titles, interests in and copyrights to the materials and information herein. The materials and information contained herein constitute confidential information of Critical Manufacturing and the Customer must not disclose or transfer by any means any of these materials or information, whether total or partial, to any third party without the prior explicit consent by Critical Manufacturing.

Copyright Information

All title and copyrights in and to the Software (including but not limited to any source code, binaries, designs, specifications, models, documents, layouts, images, photographs, animations, video, audio, music, text incorporated into the Software), the accompanying printed materials, and any copies of the Software, and any trademarks or service marks of Critical Manufacturing are owned by Critical Manufacturing unless explicitly stated otherwise. All title and intellectual property rights in and to the content that may be accessed through use of the Software is the property of the respective content owner and is protected by applicable copyright or other intellectual property laws and treaties.

Trademark Information

Critical Manufacturing is a registered trademark of Critical Manufacturing.

All other trademarks are property of their respective owners.