smart factory experts

FORCE EDGE CONNECT

Machine Repository

Version 240112

Manual

Document: Manual - FORCE EDGE CONNECT Machine Repository

Release date: 2024-01-12

Document version: 1

Author: FORCAM GmbH

Content

1 About this document

This document describes how to use FORCE EDGE CONNECT Machine Repository.
The manual describes the different components and functions that are available for defining and managing templates to connect assets of different types.

Target group

FORCE EDGE CONNECT Machine Repository is an optional extension to FORCE EDGE CONNECT, therefore the present manual requires the same knowledge about machine connection as is required for that application.
If you do not have any knowledge in this area, take the time to familiarize yourself with the basics.

ⓘ We recommend that you use our Academy: **https://forcam.com/academie/**
The FORCAM Academy provides the knowledge to effectively use the methods for digital transformation and the technologies for the Smart Factory.
Based on lean manufacturing and TPM methods, our institute team will guide you to initiate changes in the company and to use the technologies correctly.

ⓣ In our **Customer Area** you can find all manuals and product descriptions as well as additional information on your release. FORCE EDGE CONNECT is a prerequisite for using EDGE CONNECT MR. The corresponding manual is also available on the customer portal.

2 Concept

ⓘ EDGE CONNECT Machine Repository is an optional extension to FORCE EDGE CONNECT. EDGE CONNECT is therefore a prerequisite to use the Machine Repository.

FORCE EDGE CONNECT (hereafter only referred to as EDGE CONNECT) Machine Repository provides the user with the ability to define templates for connecting any asset. These can either be created via the Machine Repository (MR) configuration wizard or derived from assets already connected via EDGE CONNECT. In this way, templates offer a simple solution, especially when expanding a machine park with new, similar assets. The template-supported connection of assets significantly reduces the effort required for digitization. The product enables every company to easily create, manage and use templates for the standardized connection of the same asset types.

The use of templates for connecting the same assets ensures that identical information is derived on the basis of asset signals. This creates direct comparability of assets and makes it possible to transfer asset-related measures.

As part of the traceability of individual changes to a template, a new template version is created in MR each time a change is made. The history of a template can be viewed directly in MR. Individual versions can be restored manually.

The MR's asset list provides an overview of all assets connected in the EDGE instances.
The accumulated knowledge on the MR can be distributed across plants so that all plants can have the same templates available. You can bring your plants to the same level of digitization with ease.

3 System Components

This chapter describes the following components of the FORCAM EDGE Machine Repository in more detail:

- Definition of Templates
- Overview Asset List

3.1 Templates

The template is a connection template for digitizing a specific type of asset. It does not contain any asset specific information such as IP address or serial number of an asset.

The template can contain the following asset type specific connectivity information:

- Template name and description
- Asset type and classification
- Manufacturer and model
- Controller type (PLC/PLC) and bus type description
- Signal definition
- Script for signal interpretation
- DNC configuration

By providing the general tethering information of an asset type, the effort to digitize an asset of the same type is significantly reduced. When using a template in EDGE Configuration, the tethering information is automatically applied in the Asset Configuration Wizard.

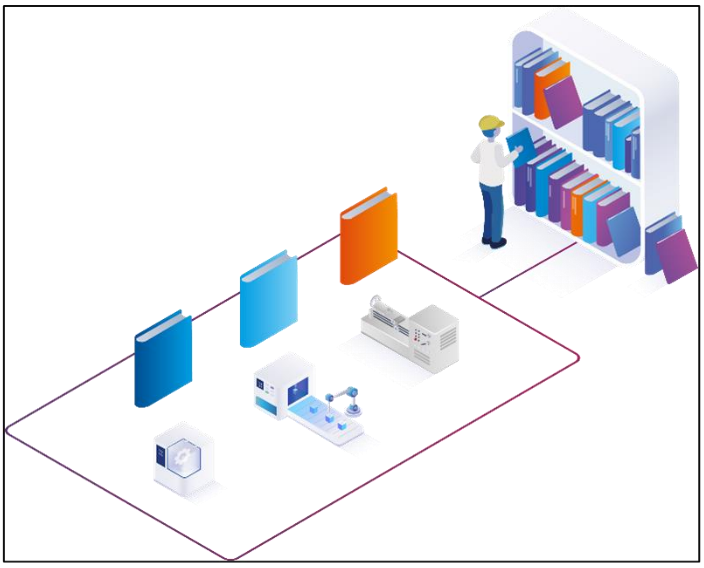


Figure 1: Template administration in the Machine Repository

3.2 Asset list

An asset is a collective term for elements that can be linked to the EDGE CONNECT (e.g., machines, sensors, databases, etc.). The asset list displays the assets of all linked EDGE instances that are connected to the EDGE CONNECT Machine Repository. In the Machine Repository, templates can be derived from the assets linked in the EDGE instances.

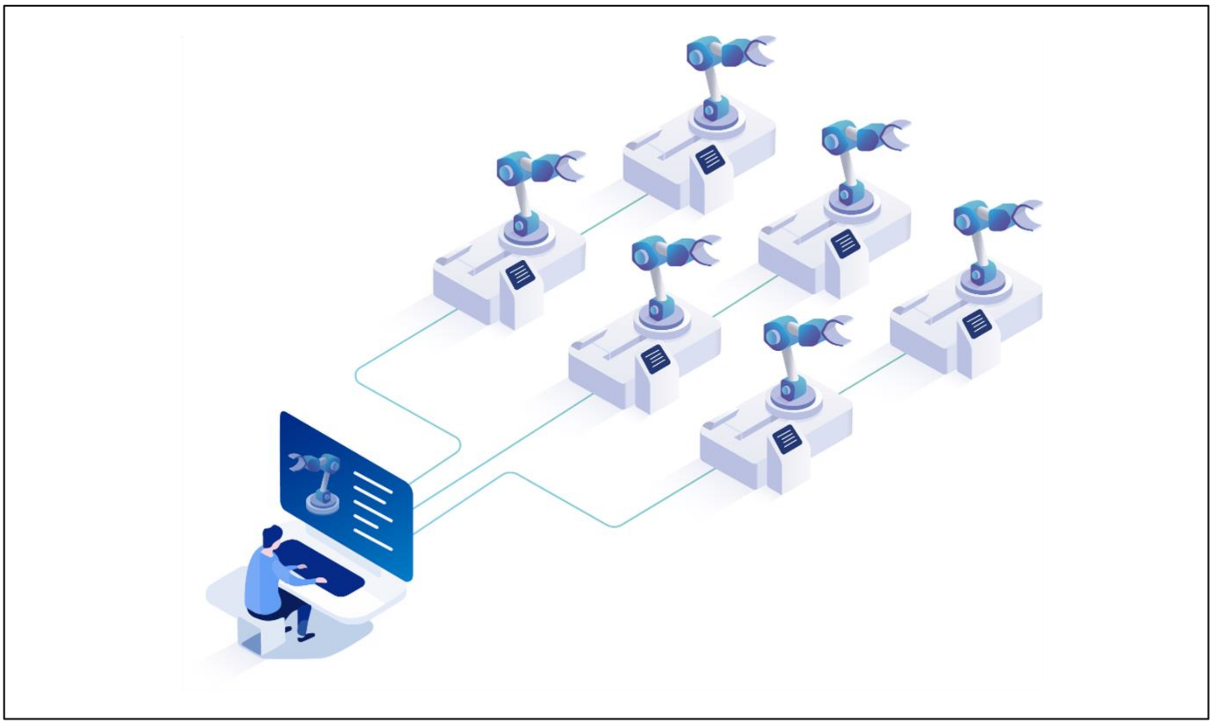


Figure 2: Overview of the asset park

4 System architecture

EDGE CONNECT Machine Repository is an optional extension to EDGE CONNECT. The MR is a standalone application that communicates with EDGE CONNECT via clearly defined interfaces. Therefore, the MR can be installed and deployed in the customer's IT infrastructure as well as in a cloud environment.
For example, several EDGE instances can be supplied by the MR. FORCAM thus makes a significant contribution to digitization in industry and focuses on the cost-efficient connection of assets across plants.

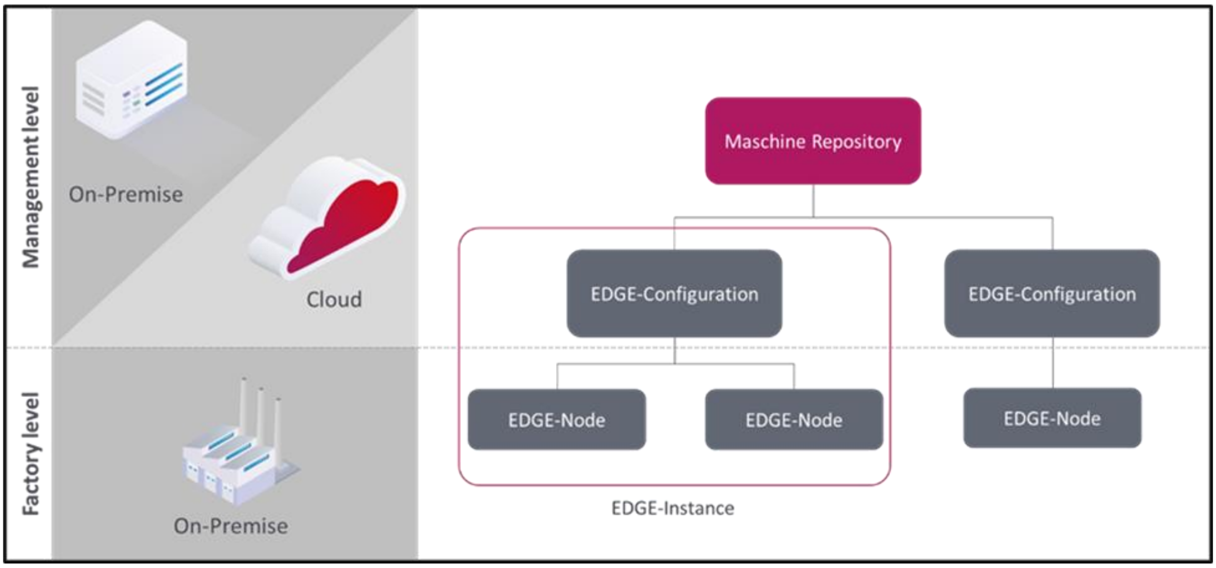


Figure 3: Architectural Structure EDGE CONNECT Machine Repository

5 Navigation Area

Templates can be managed from the menu of the MR. It is also possible to get an overview of the connected assets and the connected EDGE instances. You can also manage the users and their rights. This chapter covers **templates, assets, linked EDGE instances, user management** and **licensing** and then explains the sorting behavior of table entries.

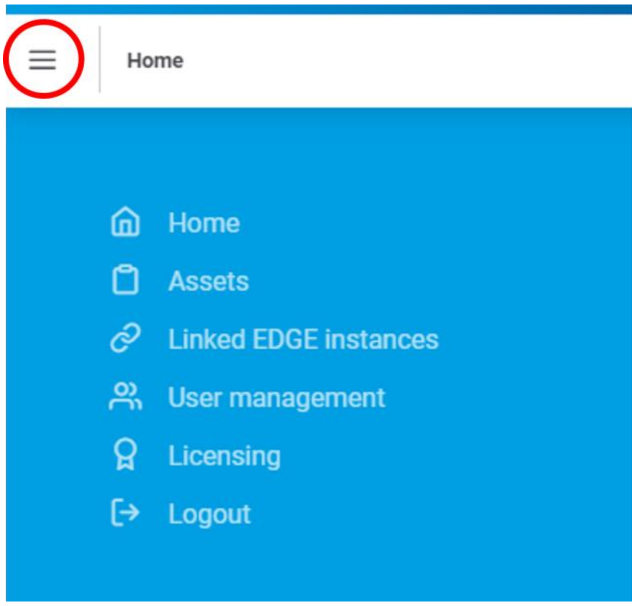
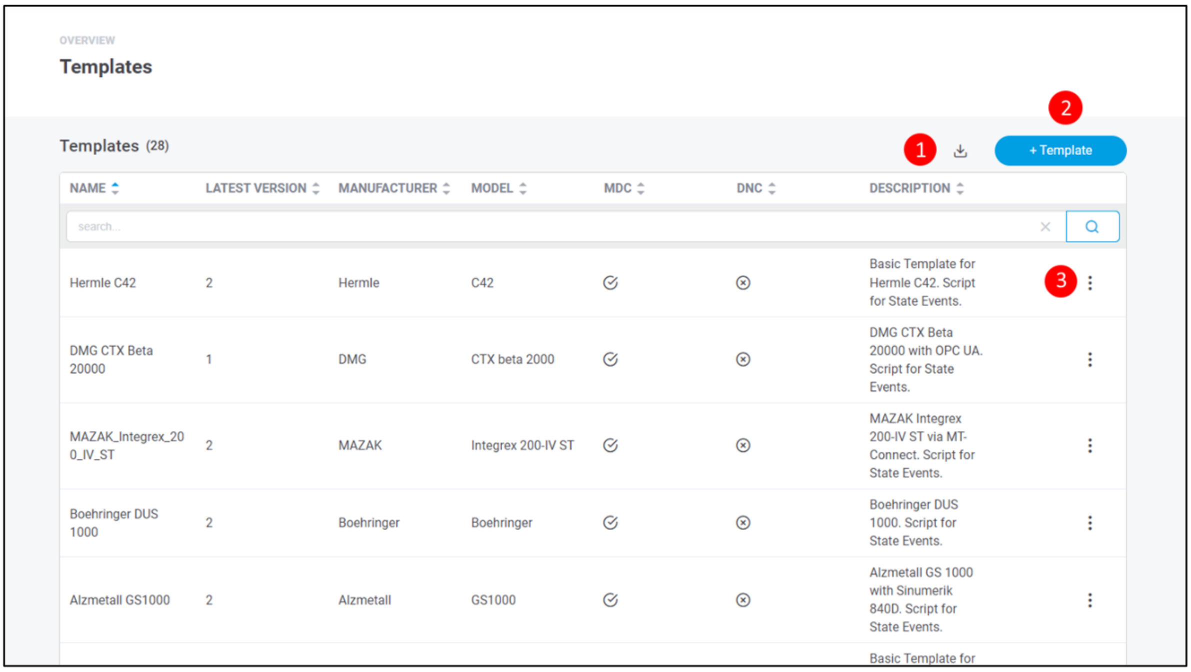


Figure 4: Calling up the menu of the Machine Repository

5.1 Template overview

On the start page of the MR, an overview of the templates already created is displayed. Among other things, templates can be created, imported, copied and deleted here. Information about the individual templates is displayed, such as the highest version number or the use of an MDC/DNC control.



NAME	LATEST VERSION	MANUFACTURER	MODEL	MDC	DNC	DESCRIPTION
Hermle C42	2	Hermle	C42	✓	⊗	Basic Template for Hermle C42. Script for State Events.
DMG CTX Beta 20000	1	DMG	CTX beta 2000	✓	⊗	DMG CTX Beta 20000 with OPC UA. Script for State Events.
MAZAK_Integrex_200_IV_ST	2	MAZAK	Integrex 200-IV ST	✓	⊗	MAZAK Integrex 200-IV ST via MT-Connect. Script for State Events.
Boehringer DUS 1000	2	Boehringer	Boehringer	✓	⊗	Boehringer DUS 1000. Script for State Events.
Alzmetall GS1000	2	Alzmetall	GS1000	✓	⊗	Alzmetall GS 1000 with Sinumerik 840D. Script for State Events.
						Basic Template for

Figure 5: Template overview

(1) Importing a template
(2) Creation of a template without attached asset (see chapter 6.1)
(3) Settings menu of the template:

- Open template
- Export template
- Copy template
- Delete

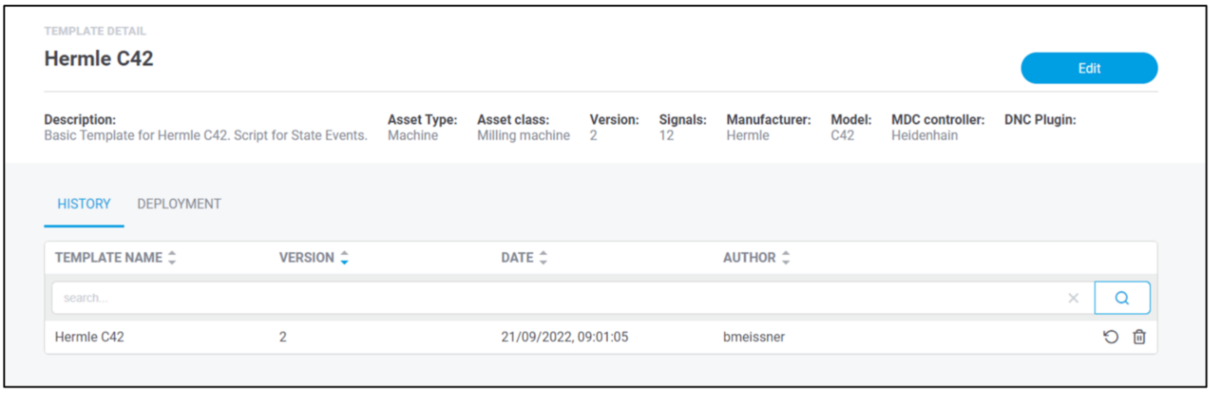
5.1.1 Open template

When opening the template, details such as asset type, version, signals, MDC control and DNC plugin are displayed. The template can be edited as desired. Each change to the template results in a new version being created. The version is incremented by 1. The history provides transparency and supports the tracking of changes. It is possible to manually reset the template to any previous version. In the **DEPLOYMENT** area, you can track for which EDGE instances the template was released with the respective version.

ⓘ The version with the highest number is the current one.

ⓘ **Edit** opens the same configuration wizard as in chapter 6.3.

ⓘ It is not possible to change the template name afterwards.



TEMPLATE DETAIL

Hermle C42

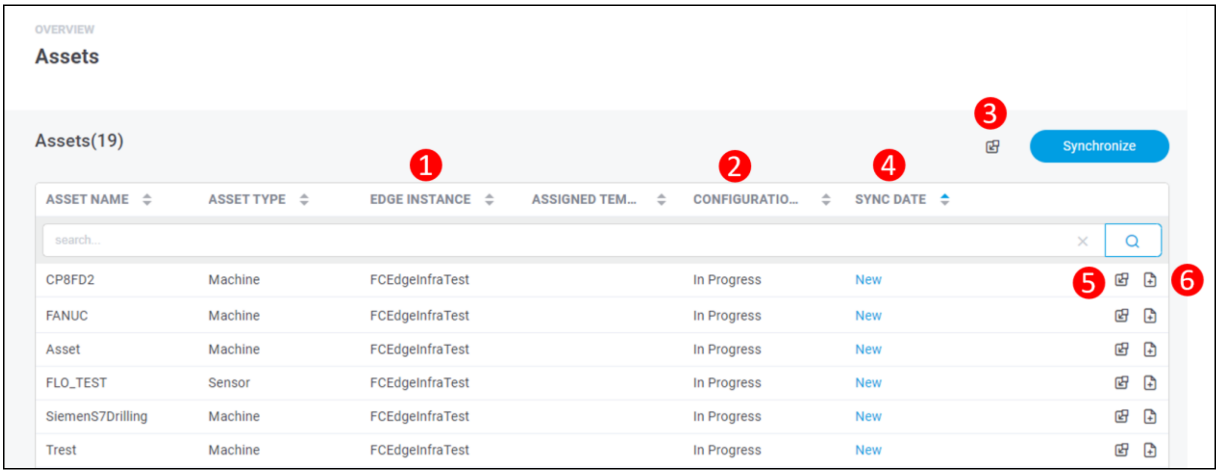
Description: Basic Template for Hermle C42. Script for State Events.
Asset Type: Machine
Asset class: Milling machine
Version: 2
Signals: 12
Manufacturer: Hermle
Model: C42
MDC controller: Heidenhain
DNC Plugin:

TEMPLATE NAME	VERSION	DATE	AUTHOR
Hermle C42	2	21/09/2022, 09:01:05	bmeissner

Figure 6: Template details

5.2 List of all assets

The list of all assets shows an overview of all assets of the EDGE instances that are connected to the MR. This gives you an overview of your machinery.



ASSET NAME	ASSET TYPE	EDGE INSTANCE	ASSIGNED TEM...	CONFIGURATIO...	SYNC DATE
CP8FD2	Machine	FCEdgeInfraTest		In Progress	New
FANUC	Machine	FCEdgeInfraTest		In Progress	New
Asset	Machine	FCEdgeInfraTest		In Progress	New
FLO_TEST	Sensor	FCEdgeInfraTest		In Progress	New
SiemenS7Drilling	Machine	FCEdgeInfraTest		In Progress	New
Trest	Machine	FCEdgeInfraTest		In Progress	New

Figure 7: Overview list of all assets

(1) Name of the EDGE instance to which the asset is attached.

(2) Configuration status of the asset:

- *In Progress*: The configuration is not yet complete and is to be continued at another time.
- *In Validation*: The configuration of the asset is to be checked for errors and consistency.
- *Completed*: The configuration is fully completed. It is recommended to generate a template from the asset only in this status.

(3) All assets that are not yet synchronized are passed to MR.

(4) Date of the last synchronization:

- New: assets that have been added but not yet transferred to MR
- Not synchronized: Configuration is not accessible, last update status is displayed

(5) Individual assets are transferred to MR

(6) Creation of a template from an asset (see chapter 6.2)

ⓘ Only when the asset is transferred to the MR, the *New* entry disappears.

5.3 Linked EDGE-Instances

This page provides an overview of the EDGE instances that are already linked to the Machine Repository as well as administration functions. Details of the EDGE instances are displayed here, such as name, status, and number of EDGE nodes. In addition, you can connect new EDGE instances to the MR here.

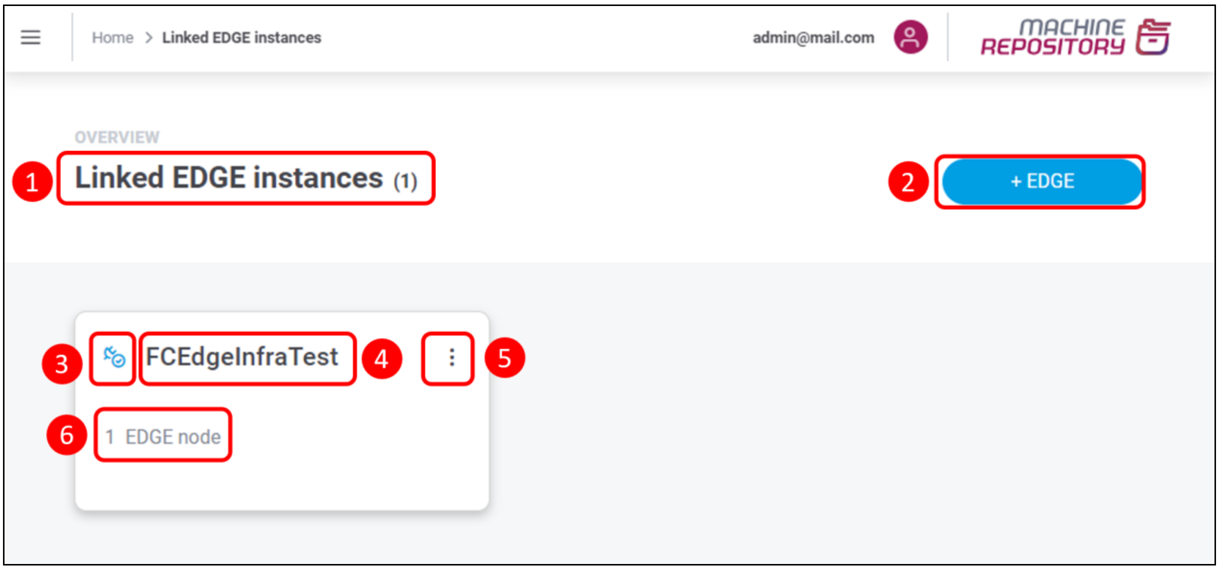


Figure 8: Overview of linked EDGE Instances

(1) Number of linked EDGE instances.
(2) Adds a new EDGE instance
(3) Status of the linked instance:

- Connected
- Disconnected

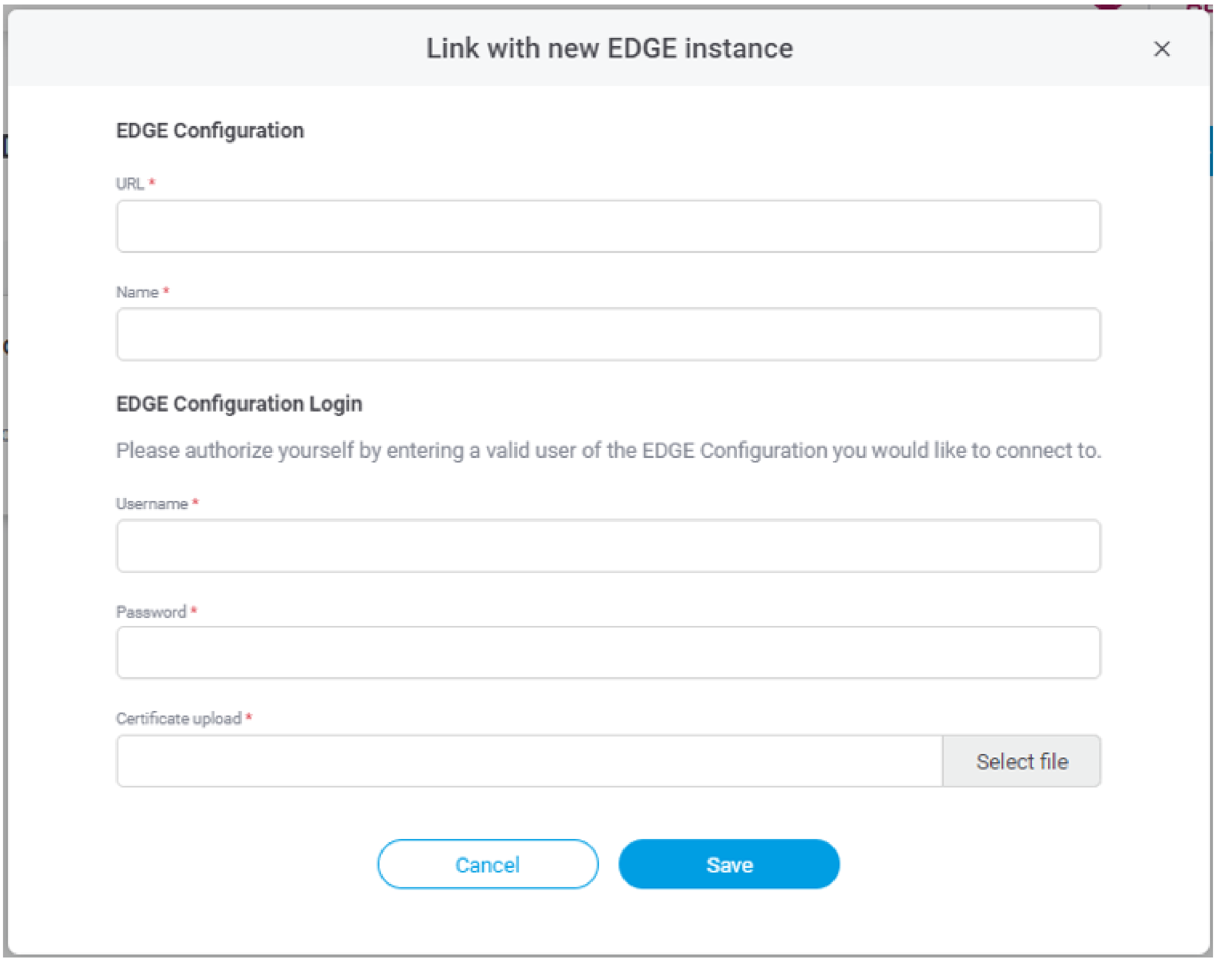
(4) Name of the EDGE instance
(5) Settings menu of the instance:

- Edit
- Delete

(6) Number of EDGE-nodes

5.3.1 Add an EDGE instance

EDGE instances can be added in the MR in a few steps. Multiple EDGE instances can be connected to the MR.



Link with new EDGE instance

EDGE Configuration

URL *

Name *

EDGE Configuration Login

Please authorize yourself by entering a valid user of the EDGE Configuration you would like to connect to.

Username *

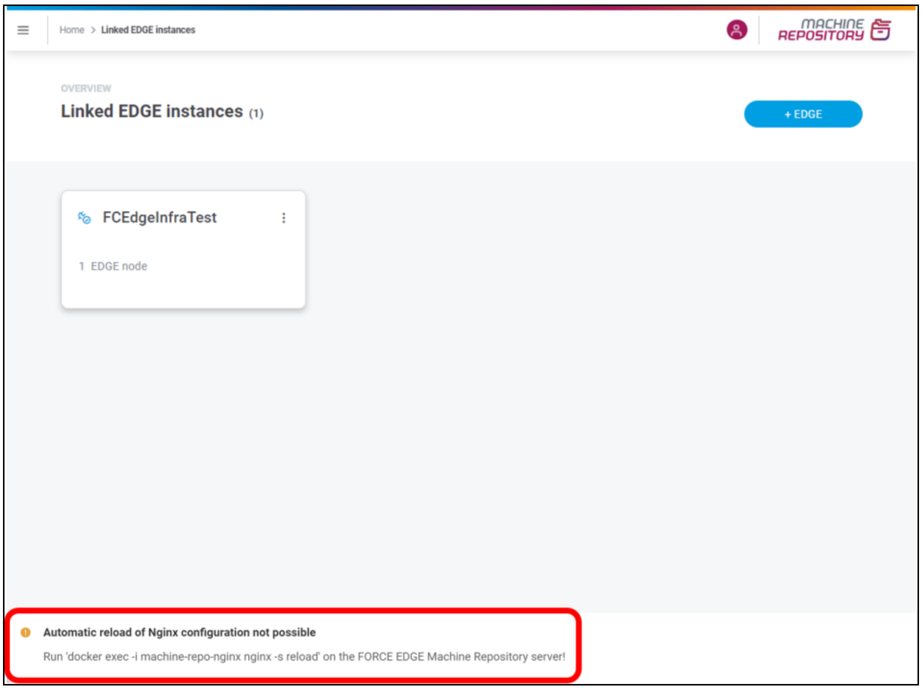
Password *

Certificate upload *

Figure 9: Dialog for adding a new instance

To add a new EDGE instance:

1. Click **+ EDGE** in the instance overview.
2. In the subsequent dialog, add all mandatory fields:
   - **URL**:
     Consist of https + IP address + port 60077. Only one EDGE instance can be added per URL.
   - **Name**: Appears in the instance overview as the title of the instance.
3. Enter **username** and **password** of the EDGE configuration that is related to the desired EDGE instance.
4. Upload certificate. (EDGE Configuration certificate that was created before the installation und used during the installation.)
5. **Save**.
   ➔ A pop-up message appears for a short time at the bottom left of the screen: To correctly connect the instance, the reload command mentioned in this message must be executed on the EDGE Machine Repository server (see next step).



Automatic reload of Nginx configuration not possible

Run 'docker exec -i machine-repo-nginx nginx -s reload' on the FORCE EDGE Machine Repository server!

Figure 10: Pop-up message with reload command

6. Execute the reload command on the EDGE Machine Repository server:
   `docker exec -i machine-repo-nginx nginx -s reload`
7. Log in to EDGE CONNECT Machine Repository again.
   ➔ After the next login, the newly created instance is correctly connected and all instance information is visible.

⚠ It is absolutely necessary to execute the command on the EDGE Machine Repository server after an instance has been created. Otherwise, the instance cannot be connected correctly and no actions can be performed on it.

5.4 User management

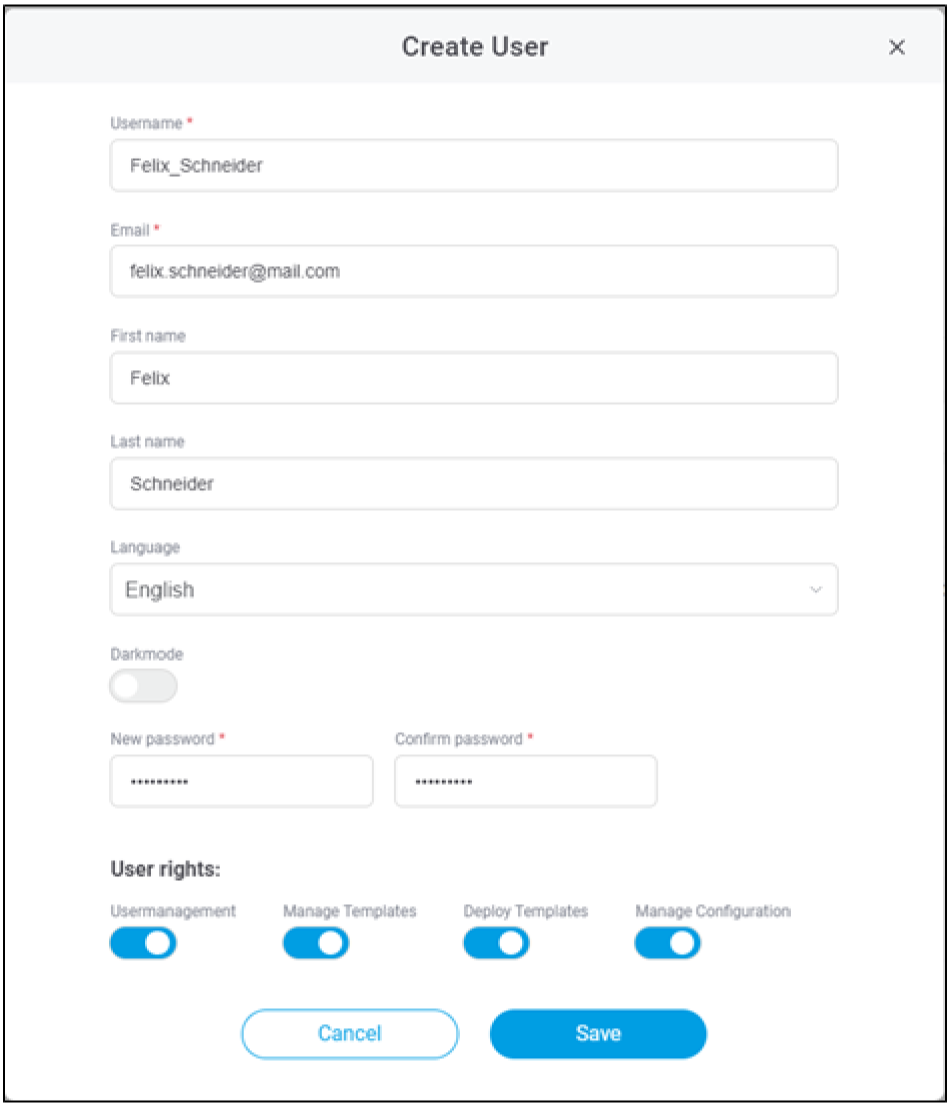
Users can be created for the EDGE CONNECT Machine Repository in the user administration. The users can be assigned appropriate rights depending on their role in the company. These are linked to different functions such as managing templates or configurations. You can also edit already existing user accounts.

ⓘ If the rights of a logged-in user have been changed, they become effective immediately after a new login. If the user does not log in again, it can take up to 30 minutes until the change is active.

To create a new user:

1. Click on **+ User**
2. Enter the **USERNAME** in the subsequent dialog.
3. Optional: enter email address, first and last name
4. Select **language**
5. Optional: activate **darkmode**
6. Set the desired password
   Must be at least 8 characters long, consist of upper- and lower-case letters and contain at least one number and one special character.
   The following special characters are allowed:
   ! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ ` { | } ~
7. Assign user rights (see Table 1).
8. **Save**.

ⓘ A user cannot be created again with the same data.



Create User

Username *
Felix_Schneider

Email *
felix.schneider@mail.com

First name
Felix

Last name
Schneider

Language
English

Darkmode
○

New password *
••••••••••

Confirm password *
••••••••••

User rights:

Usermanagement ●
Manage Templates ●
Deploy Templates ●
Manage Configuration ●

Figure 11: Dialogue for creating a new user

User right	Description
User administration	The user can call the user administration, create new users and assign/remove rights.
Manage templates	The user can call, create, copy, delete, import and export templates
Distribute templates	When configuring a template, the user can deploy the template to the different EDGE instances.
Manage configuration	The user can connect, edit and delete EDGE instances.

Table 1: User rights of EDGE CONNECT Machine Repository

5.5 Licensing

Under Licensing, licenses can be imported and viewed.

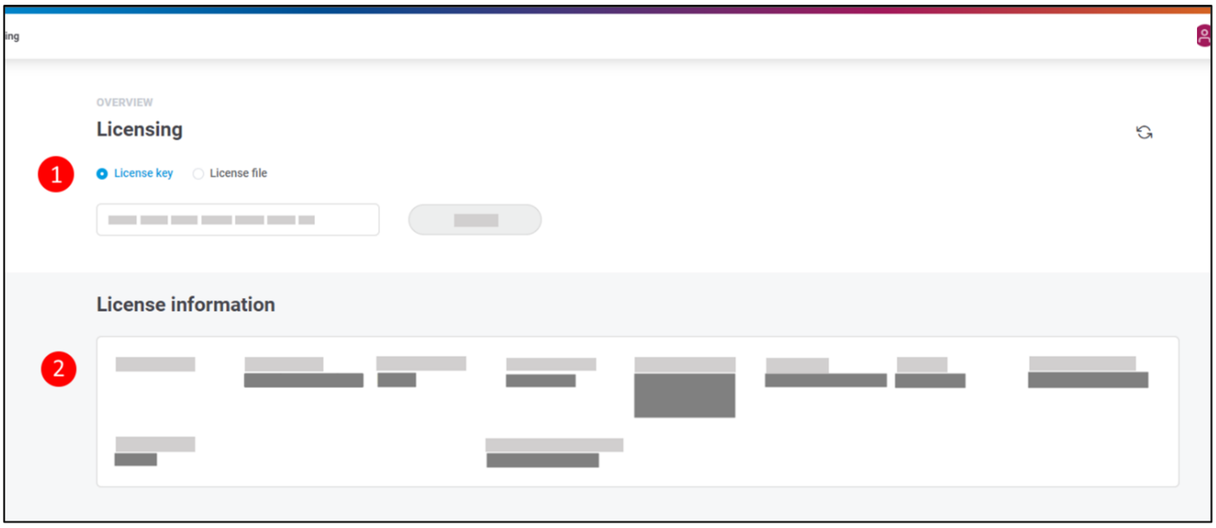


Figure 12: Licensing and overview

(1) A new license can be uploaded as a file or entered directly as a key.

(2) The license information includes type and status of the license, model, maintenance, validity and other data.

5.6 Sorting table entries

Most pages in EDGE CONNECT Machine Repository display data in the form of tables. In order to offer the familiar ease of use that you know from other table tools, the sorting function of columns has been used here as well: You can sort the columns alphabetically ascending or descending.

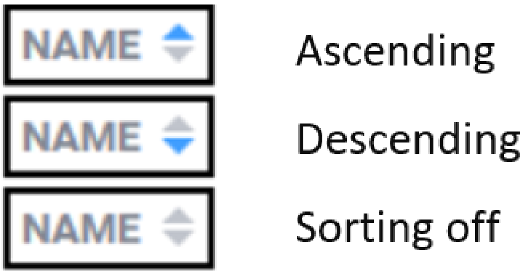


Figure 13: Alphabetical sorting of columns

6 Creation of a template

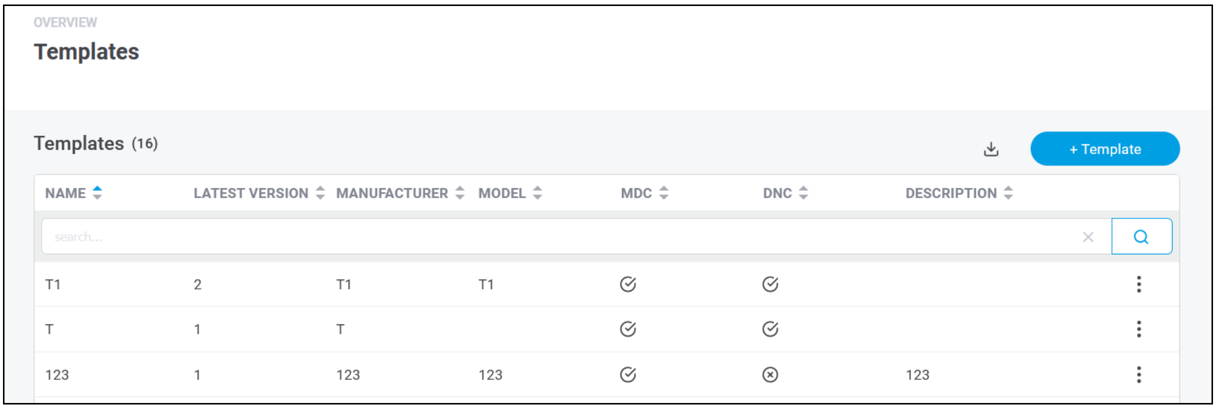
This chapter describes how to create a template. Basically, a distinction is made between two cases:

- Creation of a template without a connected asset (see chapter 6.1)
- Creation of a template from a connected asset (see chapter 6.2)

6.1 Creating a template without connected asset

You can freely configure a template for an asset type. A guided configuration wizard allows you to create a template in a few steps (see chapter 6.3). In this wizard, MDC/DNC controls are configured, machine signals are defined and the deployment for a template is specified.

ⓘ Chapter 6.3 describes the individual steps of the configuration wizard.



OVERVIEW

Templates

Templates (16)

NAME	LATEST VERSION	MANUFACTURER	MODEL	MDC	DNC	DESCRIPTION	
T1	2	T1	T1	✓	✓		⋮
T	1	T		✓	✓		⋮
123	1	123	123	✓	⊗	123	⋮

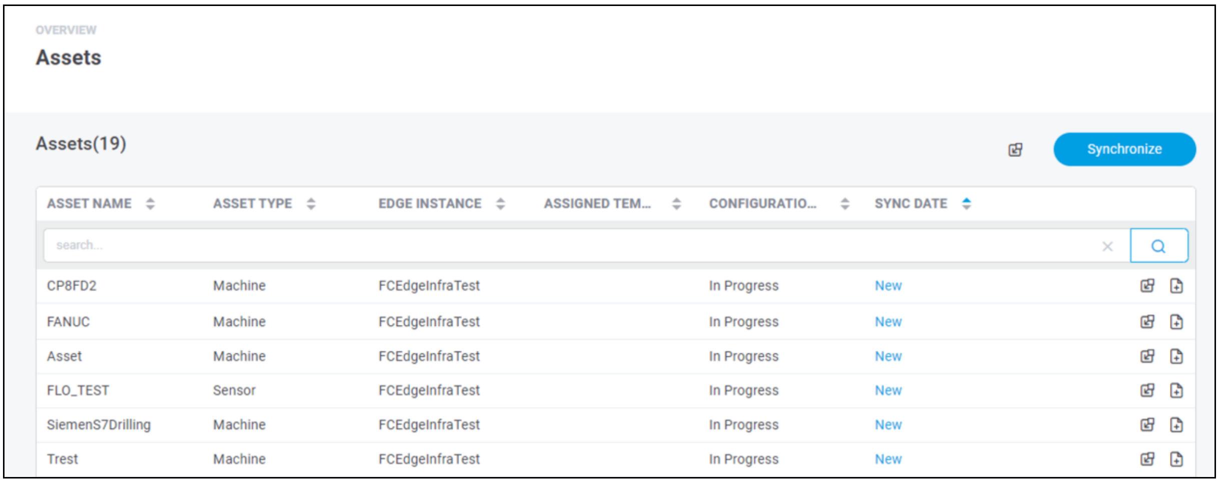
Figure 14: Create template

To add a new template:

1. Click on **Home** in the upper area.
   Or
   Clock on **Home** in the navigation area.
2. Click on **+ Templates**.
   ➔ The following dialog guides you through the next seven steps to configure a template (see chapter 6.3).
3. Click **Save** to finish.

6.2 Creating a template from a connected asset

The creation of a template by deriving an already connected asset is done with little effort. The mask of the configuration wizard is preset with the information of the selected asset. The asset-specific data, such as IP address or serial number, is not transferred.



ASSET NAME	ASSET TYPE	EDGE INSTANCE	ASSIGNED TEM...	CONFIGURATIO...	SYNC DATE
CP8FD2	Machine	FCEdgeInfraTest		In Progress	New
FANUC	Machine	FCEdgeInfraTest		In Progress	New
Asset	Machine	FCEdgeInfraTest		In Progress	New
FLO_TEST	Sensor	FCEdgeInfraTest		In Progress	New
SiemenS7Drilling	Machine	FCEdgeInfraTest		In Progress	New
Trest	Machine	FCEdgeInfraTest		In Progress	New

Figure 15: Create a template from a connected asset

To add a new template based on an asset:

1. Select **Assets** in the navigation pane**.**
2. Click the template icon in the right column for the desired asset).

ⓘ The steps in the Configuration Wizard are the same as in chapter 6.3.

ⓘ The pre-filled configuration fields in the template can be customized manually as desired.

ⓘ FORCAM recommends generating a template from the asset when the configuration status is completed.

6.3 Configuration Wizard

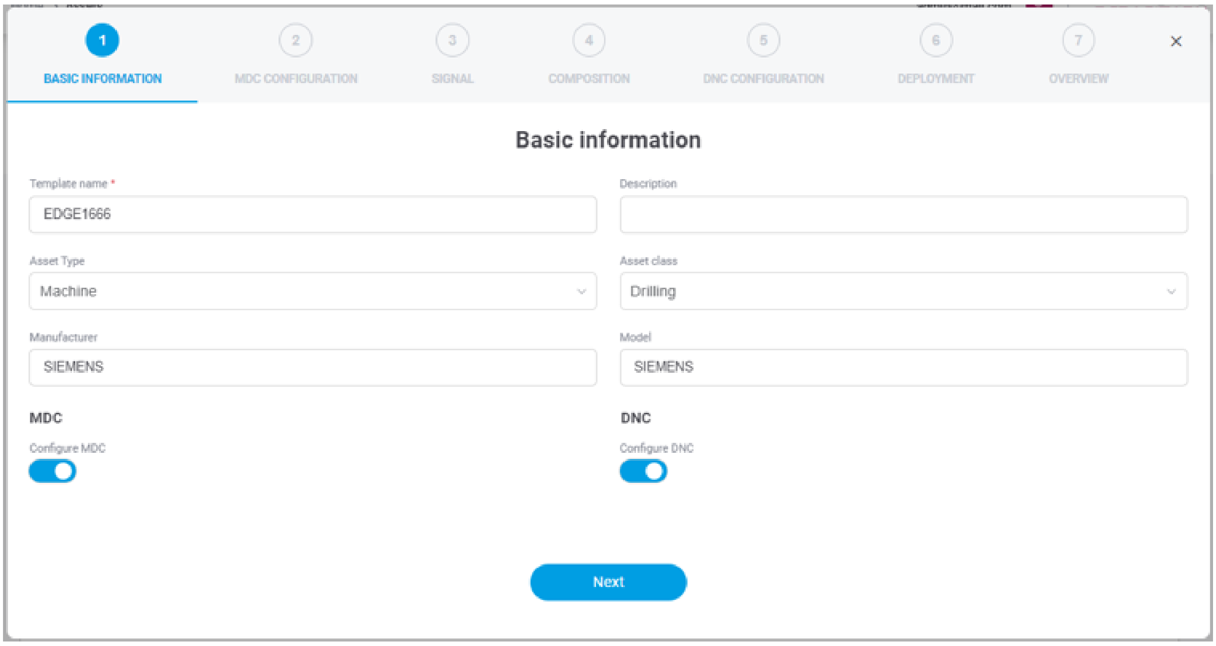
This section describes the individual steps of the configuration wizard.

ⓘ If a step is completed, it is highlighted in blue in the top bar. Clicking on a completed step returns to it.

6.3.1 ① Basic information

Basic information about the template is specified here, such as name or asset type. In addition, it is determined whether an MDC or a DNC control is to be configured, or both.

⚠ The template name cannot be changed after the complete configuration.



Basic information

Template name * EDGE1666

Description

Asset Type Machine

Asset class Drilling

Manufacturer SIEMENS

Model SIEMENS

MDC

Configure MDC ●

DNC

Configure DNC ●

Figure 16: Configuration Wizard - Basic Information

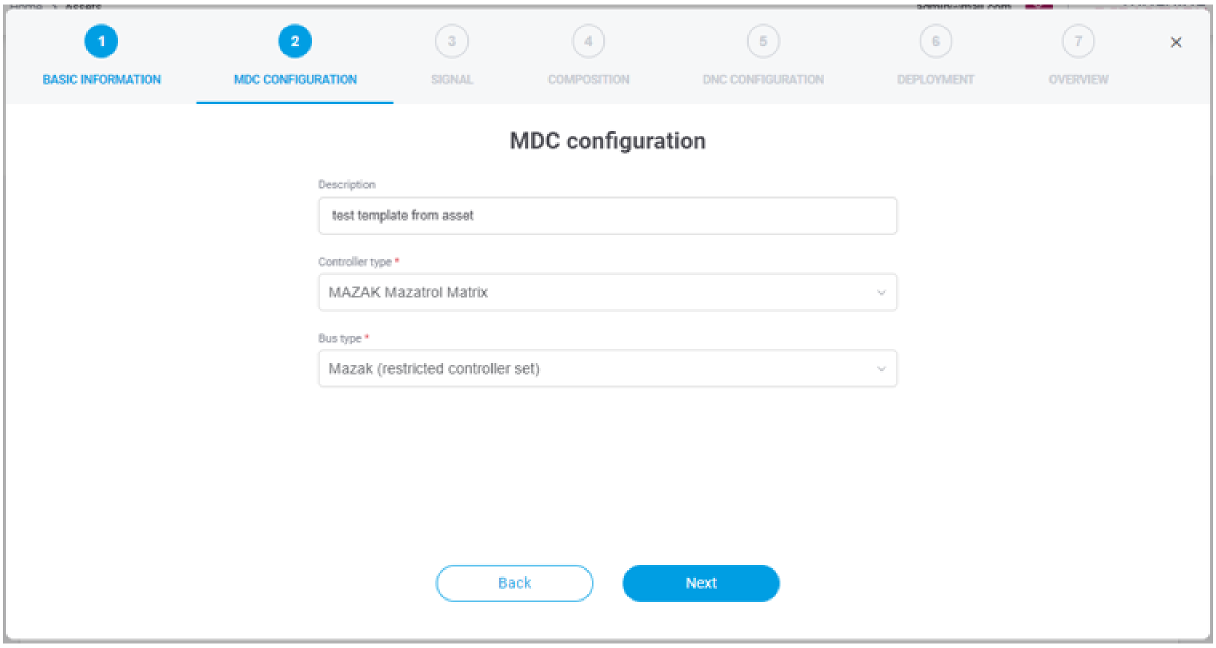
Entry field	Description
Template name	Mandatory field
Description	Free text, optional field
Asset type	Selection: – Machine – Sensor – IT System
Asset class	Available options depend on the selected asset type
Manufacturer	Free text, optional field
Model	Free text, optional field

Entry field	Description
Configure MDC	Activate the switch if you want to configure MDC.
Configure DNC	Activate the switch if you want to configure DNC.

6.3.2 ② MDC Configuration

Describes the way in which communication with the asset takes place. The **controller type** defines the type of controller to communicate with. The **bus type** is a specific communication protocol of the controller type. An overview of the current FORCAM plug-ins is listed in section 7.3.

ⓘ This step is only available if **Configure MDC** was activated in step ①.



MDC configuration

Description: test template from asset

Controller type *: MAZAK Mazatrol Matrix

Bus type *: Mazak (restricted controller set)

Figure 17: Configuration Wizard - MDC configuration

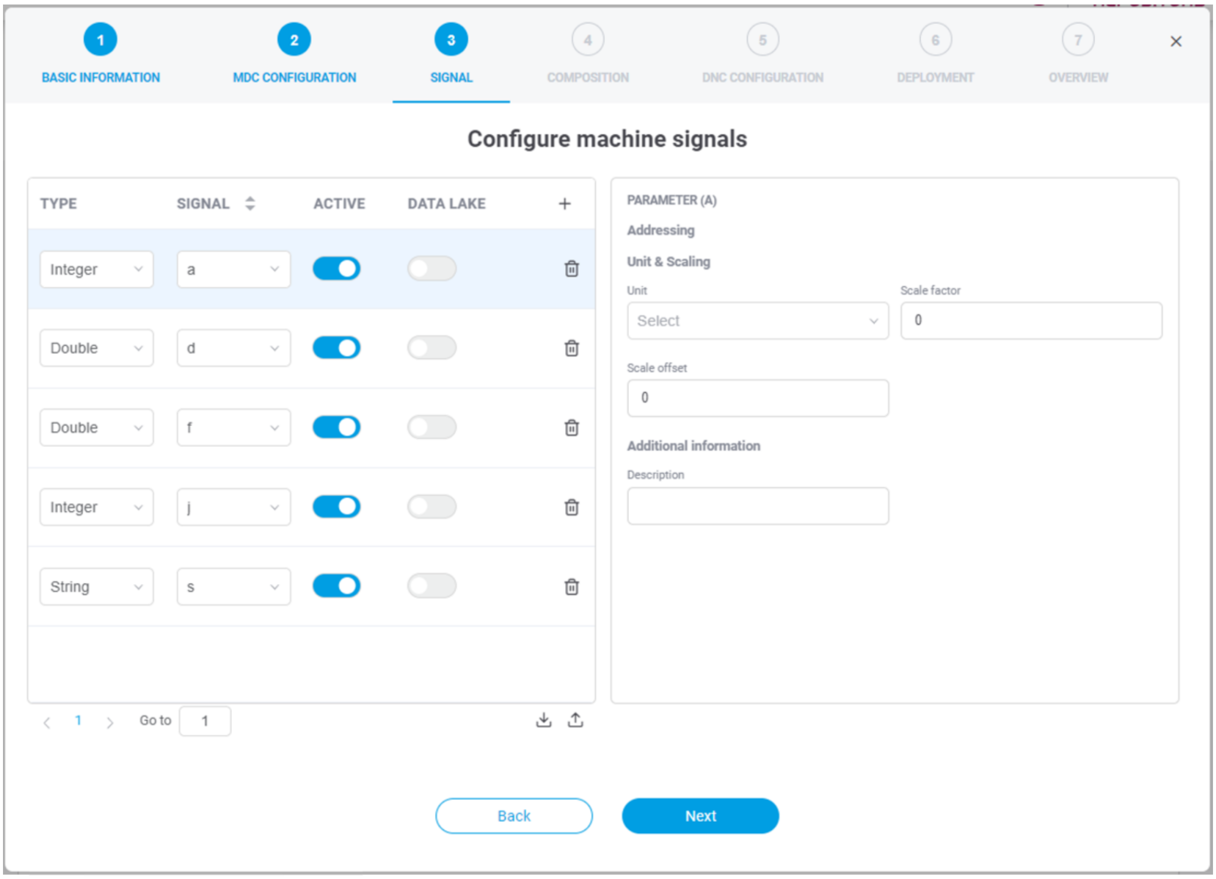
Entry field	Description
Description	Free text, optional field
Controller type	Select the controller you want to configure
Bus type	Available options depend on the selected controller type

6.3.3 ③ Signal

This step defines which signals are read out from the controller. Depending on the configuration of the MDC controller (step ②), different listings of the signal types are displayed. The Data Lake can be used to record and save all data. Data Lake storage can be switched on and off as per signal. Units can be specified for the individual signals (e.g., degrees Celsius or liters per minute), and scaling factors can also be set. The scaling factor makes it possible, for example, to infer temperature by the resistance recorded on an asset.

ⓘ Signals can be imported or exported by means of a CSV file.

ⓘ If the **ACTIVE** switch is deactivated for a signal, the signal cannot be used in step ④ **COMPOSITION**.



Configure machine signals

TYPE	SIGNAL	ACTIVE	DATA LAKE
Integer	a	●	○
Double	d	●	○
Double	f	●	○
Integer	j	●	○
String	s	●	○

PARAMETER (A)

Addressing

Unit & Scaling

Unit: Select

Scale factor: 0

Scale offset: 0

Additional information

Description:

Figure 18: Configuration Wizard – Signal

Input field	Description
TYPE	Data type of the signal; available options depend on the variant used for asset connection
SIGNAL	Free text, optional field
ACTIVE	Switch must be activated if you want to use the signal in step ④
DATA LAKE	The Data Lake component must be purchased separately.
PARAMETER	

Input field	Description
Addressing	available fields depend on the selected TYPE
Unit and Scaling	Unit: Select the unit Scaling factor: Scale offset:
Additional information	Free text, optional field

6.3.4 ④ Composition

In this step, the received signals are interpreted and interpretation conditions are defined. As a result, measurement values, maintenance information and production states are available. This makes it possible to draw logical conclusions about the behavior of the asset.

ⓘ We recommend creating internal company guidelines for signal composition. This creates a uniform data model across all assets, which forms the basis for comparative evaluations.

There are two ways to implement this signal interpretation: In the **SCRIPT** section, text-based code is displayed and edited (see Figure 20), whereas in the **GRAPHIC** section graphical blocks can be used (Figure 19).

⚠ Simultaneous editing in **SCRIPT** and **GRAPHIC** is not possible, neither can you switch from **SRIPT** to **GRAPHIC**.
Once changes are made in the script editor, further editing must be done there, and the code will no longer be displayed under **GRAPHICS**.

Graphical editor

The blocks in the graphical editor are programming blocks/modules that can be put together and connected, similar to the individual pieces of a puzzle. The advantage of this modular system is that you can create the required commands even if you are new to programming in general.

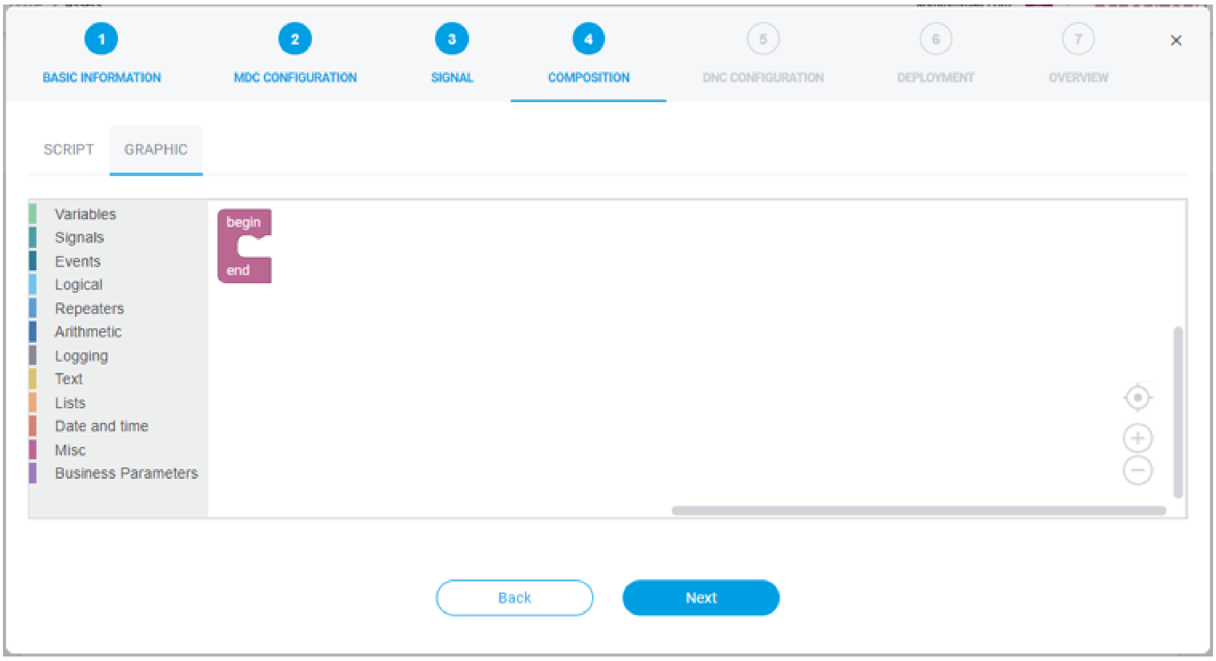


Figure 19: Graphical editor

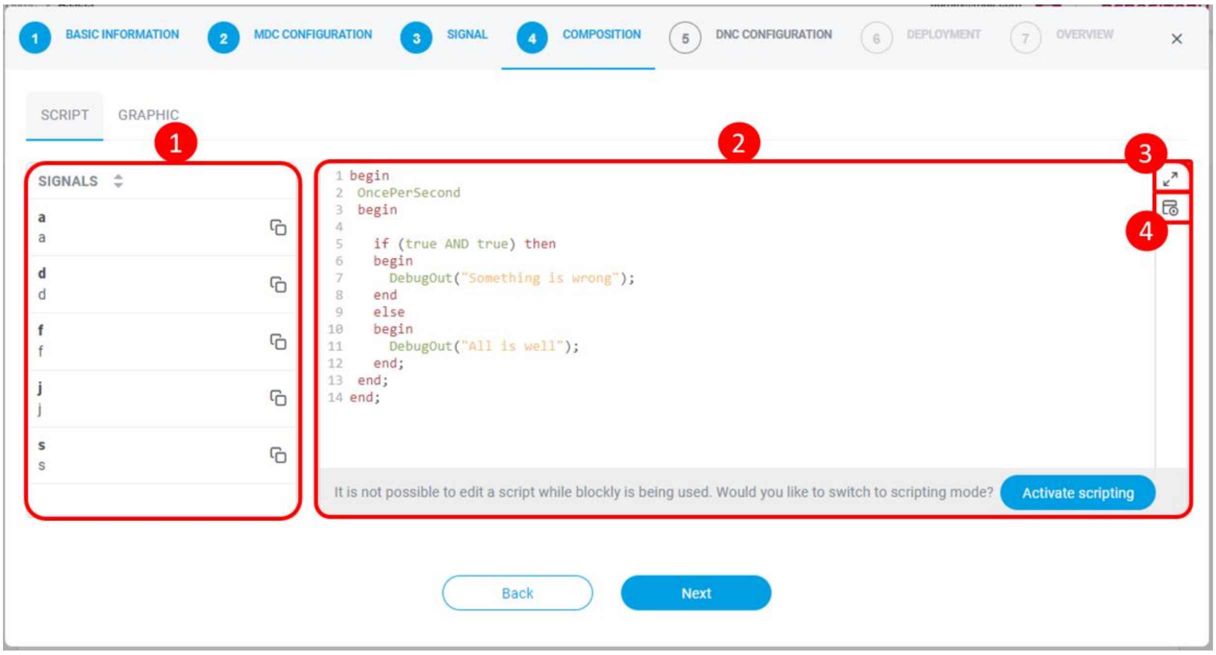
On the left side of the screen, all available function categories are listed, divided, and sorted by color. Drag-and-drop can be used to move the required blocks to the editing area on the right and place them in the correct order. This is where the actual asset logic is defined.

ⓘ For a detailed description of the individual function categories of the blocks, see the **Graphical Composition** manual.

Script editor

The annex of this manual contains sample scripts and script functions.

⚠ Only users that are familiar with programming should work in the **SCRIPT** editor.



```
begin
 OncePerSecond
 begin

   if (true AND true) then
   begin
     DebugOut("Something is wrong");
   end
   else
   begin
     DebugOut("All is well");
   end;
 end;
end;
```

It is not possible to edit a script while blockly is being used. Would you like to switch to scripting mode?

Figure 20: Script editor

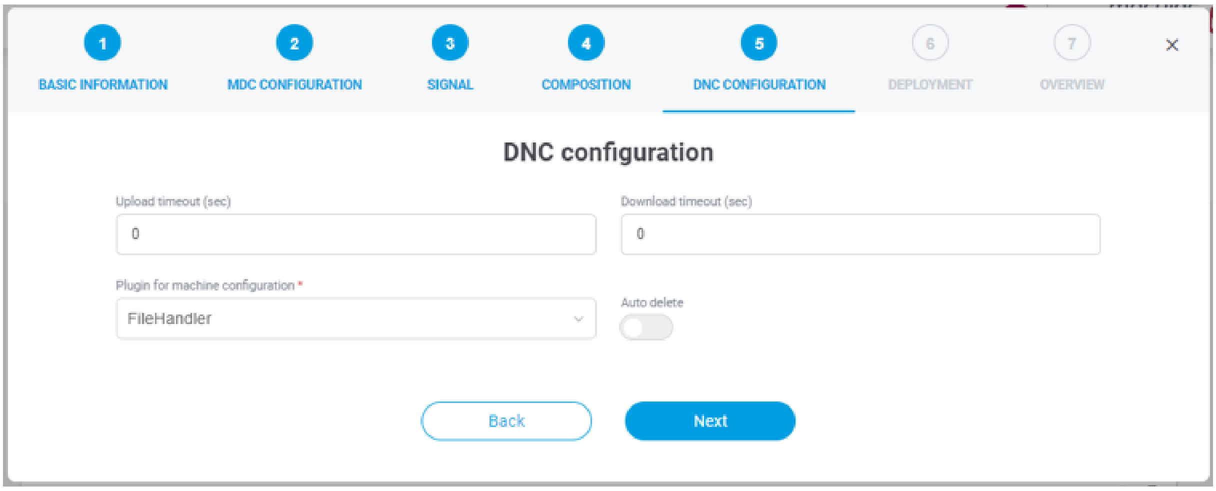
(1) Shows the signals that were added in step ③ of the Configuration Wizard
(2) Editing area with current script
(3) Zoom in/out view (full screen)
(4) Check the current script for validity

⚠ The script must be error-free. You can only proceed to the next configuration step if the script has no errors.

6.3.5 ⑤ DNC configuration

In this step, a DNC control can be configured. The DNC determines the way an NC file is transferred to the asset.
Section 7.3 provides an overview of currently available plug-ins.

ⓘ This step is only available if **Configure DNC** has been activated in step ①.



DNC configuration

Upload timeout (sec): 0

Download timeout (sec): 0

Plugin for machine configuration *: FileHandler

Auto delete

Figure 21: Configuration Wizard – DNC Configuration

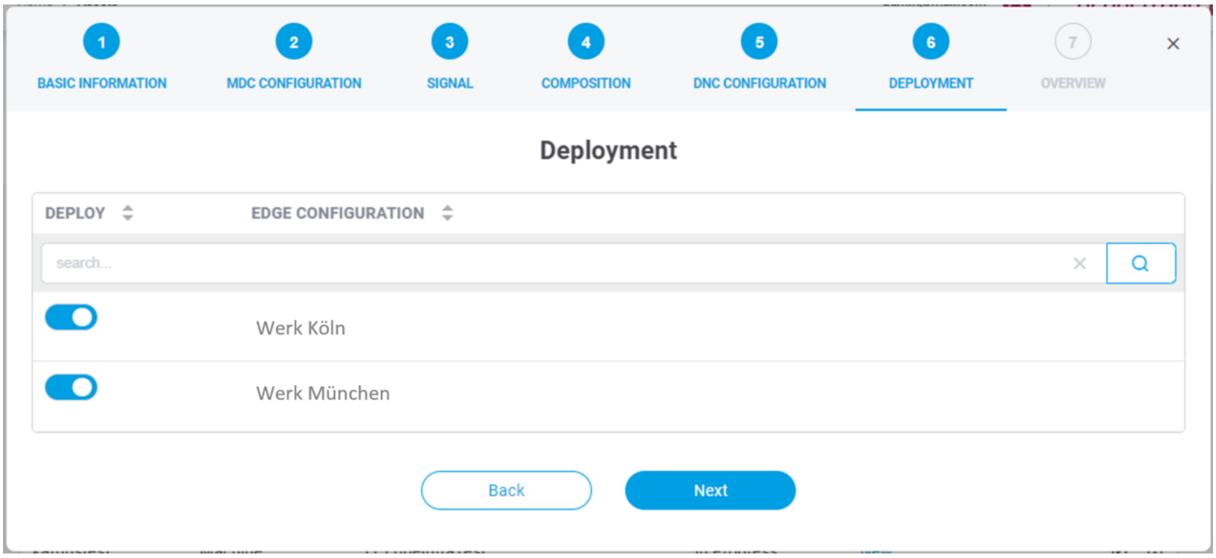
d	Description
Upload timeout (sec)	Defined in seconds
Download timeout (sec)	Defined in seconds
Plugin for machine configuration	Selection of all available plug-ins, mandatory field
Auto delete	Activate the switch to automatically delete the NC file after reading it from the asset

ⓘ If the switch for Automatic deletion is active, the file is automatically deleted after the asset is read out. Otherwise, the files accumulate on the asset and manual deletion is required.

6.3.6 ⑥Deployment

The deployment specifies the EDGE instances to which the template is to be released.

ⓘ Only the EDGE instances that are connected to the MR are listed.



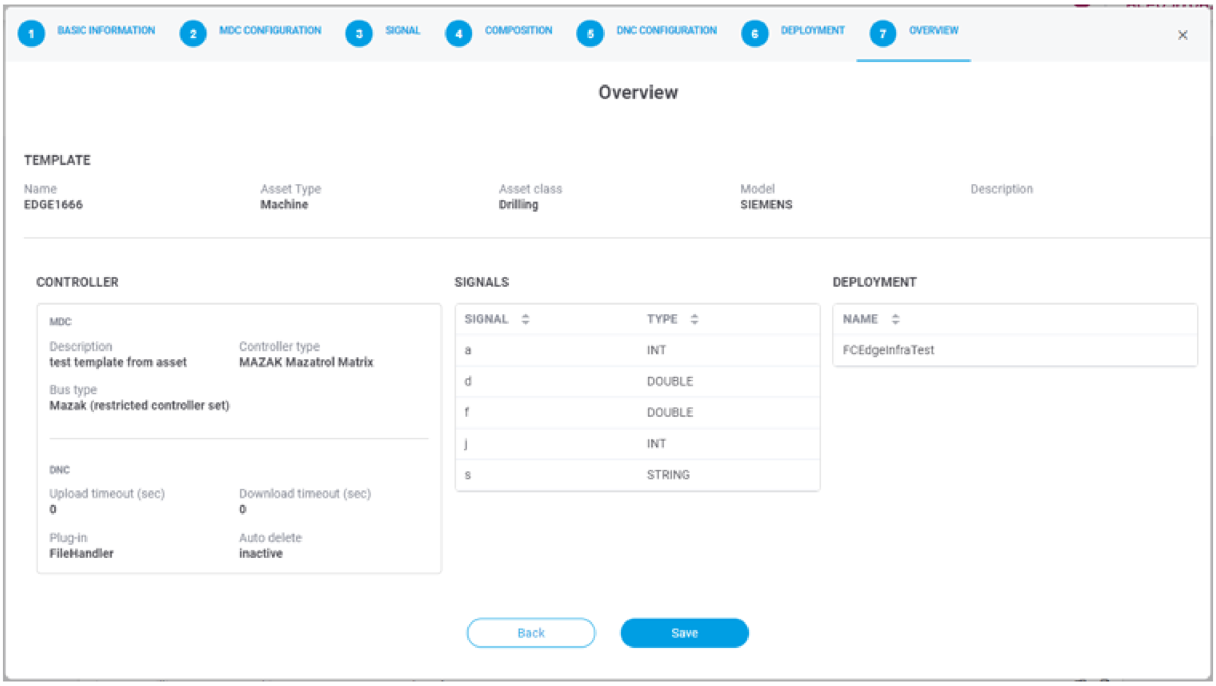
Deployment

DEPLOY	EDGE CONFIGURATION
●	Werk Köln
●	Werk München

Figure 22: Configuration Wizard – Deployment

6.3.7 ⑦ Overview

This step is a summary of the configuration settings from all previous steps and lists the MDC/DNC controls, signals and deployment. Confirmation creates the template, with the specified configurations, and displays it on the home page as a table view.
Consequently, the template is deployed to the corresponding EDGE instances. From the template, a concrete asset can now be connected in the EDGE Configuration.



Overview

TEMPLATE

Name	Asset Type	Asset class	Model	Description
EDGE1666	Machine	Drilling	SIEMENS	

CONTROLLER

MDC

Description: test template from asset
Controller type: MAZAK Mazatrol Matrix
Bus type: Mazak (restricted controller set)

DNC

Upload timeout (sec): 0
Download timeout (sec): 0
Plug-in: FileHandler
Auto delete: inactive

SIGNALS

SIGNAL ⇕	TYPE ⇕
a	INT
d	DOUBLE
f	DOUBLE
j	INT
s	STRING

DEPLOYMENT

NAME ⇕
FCEdgeInfraTest

Figure 23: Configuration Wizard – Overview

7 Appendix

7.1 Document-Convention

Conventions	Description
Boldface	Buttons and options names are written in boldface.
Italics	Highlighted words are in italics.
Icons	For a function that is represented by an icon, the icon is referenced as the object.
Action result	Action results are indicated by ➔.
Prerequisites	Prerequisites are indicated by ✓.
Warnings	Warnings are indicated by ⚠.
Notes	Notes are indicated by ⓘ.
Tips	Tips are indicated by ⓣ.

Table 2: Fonts, formatting and characters used

7.2 Abbreviations and terms

Abbreviations	Explanation
Apache Kafka	Apache Kafka is a distributed messaging system that uses the publish-subscribe method.
Asset	Generic term for all objects that EDGE CONNECT can connect (e.g. machines, sensors, databases, IT systems, etc.).
Brownfield	An existing factory or manufacturing facility that has already been built and in operation for some time. The Brownfield approach in the context of Industry 4.0 means the digital transformation of an existing manufacturing plant.
CP	Communication processor
DNC	Distributed Numerical Control: NC systems that are connected to a computer. The individual systems can be centrally supplied with NC programs and then coordinated.
IT	Information technology
Asset	In EDGE CONNECT, the asset is a partial plant according to ISA 95. If there are no other partial plants (i.e. additional physical controls), we refer to it as a plant.
MDC	Asset data connection (asset data collection)
MQTT	Message Queuing Telemetry Transport: Open network protocol for asset-to-asset (M2M) communication that enables transmitting telemetry data in the form of messages between devices, despite high delays or network limitations.

MR	Asset Repository
Northbound	A northbound interface communicates with a higher level element in a particular network component.
OT	Operational technology
POST	POST is a method which is supported by HTTP and means that a web server accepts the data contained in the body of the message requested.
PUT	The PUT method is used to update a resource available on the server. Typically, it replaces everything existing at the target URL with something else.
REST	Programming paradigm for distributed systems (collection) of independent computers that present themselves to the user as a single system.)
RESTful API	API for data exchange based on HTTP requests via GET, PUT, POST and DELETE, which is subject to the requirements or restrictions of the REST architecture.
Signal	Values read from the asset control, such as temperature, pressure or certain statuses.
Southbound	An equivalent to the northbound interface, a southbound interface communicates with lower level components.
SPS	Programmable Logical Control
UTC	Coordinated Universal Time
Wildcard	Placeholder for other characters.

Table 3: Abbreviations and terms used

7.3 List of supported plug-ins

MDC Plug-ins

MDC plug-in	Read	Write	Transmission: Polling/Event-based
CSV File Reader	X		X/
Database Exchange	X		X/
Euromap 63	X		X/
FANUC	X	X	X/
FORCAM I/O Controller	X	X	/X
Heidenhain	X	X	X/
MAKINO Pro 3/Pro 6	X		X/
MAZAK Mazatol Fusion M640M	X	X	/X
MAZAK Mazatol Fusion M640MTPro	X	X	/X

MDC plug-in	Read	Write	Transmission: Polling/Event-based
MAZAK Mazatol Matrix	X	X	/X
MAZAK Mazatol Smart	X	X	/X
MAZAK Mazatol Smooth	X	X	/X
Mitsubishi	X		X/
Modbus	X		X/
MQTT	X	X*	/X
MT Connect	X		X/
Node-RED	X	X	/X
Okuma	X		X/
Omron CS/CJ	X	X	X/
Omron CV	X	X	X/
OPC DA	X	X	X/
OPC XML-DA	X		X/
OPC UA	X	X	/X
Rockwell / Allen Bradley	X	X	X/
Schneider Electric iEM3000 Schneider Electric Pm3000/Pm5000 Schneider Electric Power Tag Energy F160 and Rope Schneider Electric Power Tag Energy M250/M630 Schneider Electric Power Tag Energy X63	X		X/
Siemens S5	X		X/
Siemens S7 (200, 300, 400, 1200, 1500)	X	X	X/
Siemens LOGO!	X		X/
WAGO 750	X		X/
Weihenstephan	X		X/

MDC plug-in	Read	Write	Transmission: Polling/Event-based
Wiesemann & Theis (WUT)	x		x/

* - Writes only to the related queue. No verification of writing to the asset.

DNC Plug-ins

DNC plug-in	Read	Write
COM	x	x
External program file transfer	x	x
FANUC	x	x
FileHandler	x	x
FileHandlerServer	x	x
File system copy	x	x
File system FTP	x	x
Heidenhain	x	x
MAZAK	x	x
Mitsubishi	x	x
MOXA	x	x
Wiesemann & Theis (WUT)	x	x

7.4 Script functions

Application	Script function Parameters in [..] are optional	Description	Output event
Default	SendImpulse(ImpulseCount, [Reference])	Sends impulse	Impulse
Default	SendQuantity(Quantity, [Unit], [QualityDetail], [Reference])	Sends quantity.	Quantity
Custom	SendState(State, [StatusCodesListName], [Reference])	Sends status.	State
Default	SendStateProduction([StatusCodesListName], [Reference])	Sends production status.	State
Default	SendStateStoppage([StatusCodesListName], [Reference])	Sends the stop state.	State
Default	SendSignalValue(SignalName, Value, [Unit], [Reference], [CustomerSpecificSetting], [Timestamp])	Sends the value of a signal. Data type "Long" (L) must be used for the timestamp list.	SignalValue
Default	SendSignalPackage(SignalNamesListName, ValuesListName, [UnitsListName], [Reference], [CustomerSpecificSetting], [TimestampsListName])	Sends signal values as a package. Data type "Long" (L) must be used for the timestamp list.	SignalPackage
Custom	SendGenericInformation(ParamName, ParamValue, [Reference])	Sends generic information.	GenericInformation
Helper	ListNew(ListName, DataType)	Creates a new list with the name ListName and list elements of the data type DataType (S for string, B for boolean, N for number).	-
Helper	ListAdd(ListName, Value)	Adds an element to the list.	-
Helper	ListClear(ListName)	Empties the list.	-
Helper	ListDelete(ListName)	Deletes the list.	-
Helper	GetMachinestatus()	Indicates the asset status.	-
Helper	GetMachineData(ParameterName)	Indicates asset data for the specified parameter.	-
Helper	SetParameter(ParameterName, ParameterValue)	Sets a new value for the specified parameter.	-
Helper	GetParameter(ParameterName)	Fetches the value for the specified parameter.	-
Helper	DeleteParameter(ParameterName)	Deletes the parameter.	-
Helper	DeleteAllParameters()	Deletes all parameters.	-
Helper	OFFLINE	Indicator whether the controller is offline or not.	-
Helper	IPADDRESS	The IP address of the Composition.	-

Application	Script function Parameters in [..] are optional	Description	Output event
Helper	HOSTNAME	Hostname of the Composition.	-
Helper	SQRT(args)	Root function MATH.	-
Helper	SIN(args)	Sine function MATH.	-
Helper	COS(args)	Cosine function MATH.	-
Helper	TAN(args)	Tangent function MATH	-
Helper	RISINGEDGE(args)	At the beginning the variable is FALSE, the EDGE checks if the values have changed. If this is the case, the variable is corrected to TRUE.	-
Helper	FALLINGEDGE(args)	Checks if the last inspected value was true and if it is false.	-
Helper	SUBSTRING(str, startIndex[, endIndex])	Substring of the specified string.	-
Helper	TONUMBER(str)	String to number (double), replaces comma to period in string.	-
Helper	TOSTRING(str or number[, formatSpecifier])	Specifies the format of the form width. The default formatting is used for empty strings. Width is the minimum length of the result string. Precision is the number of decimal places. If not specified, 0 is used. If the format specification starts with 0, the result string is prefixed with filled zeros. If the format specification ends with X, the number is converted to hexadecimal, using upper or lower case letters with upper or lower case x. In this case, the decimal places are always cut off.	-
Helper	LENGTH(obj)	The length of an object as a string value.	-
Helper	FORMATTIME(timeformatStr, timeOffset, [, timeunit])	Formats the current time with the time unit as one of the following: MILLISECOND SECOND MINUTE HOUR TAG MONTH YEAR MSABSOLUTE (current time) "R" at Format is specified as a number in milliseconds, otherwise the format is used and the offset and time unit are used to calculate the time.	-

Applicat ion	Script function Parameters in [..] are optional	Description	Output event
Helper	STDLOG(ignored, logLevel, suffixNumber, logText)	The first parameter is ignored. The log level should be W = warning, C or F = error and everything else for the debug level. The suffix number, if not 0, is added to the end of the log text as "(<SuffixNumber>)" with script loggers.	-
Helper	DEBUGOUT(text)	Logs the text at debug log level with parser logger.	-
Helper	COPYFILE(inFile, outFile)	Copies data from in-file to out-file. Arguments can be file paths. If successful, the last modified out-file is also updated as in-file.	-
Helper	COPYREPLACE(inFile, outFile, searchStr, replaceStr)	Copies from in-file to out-file as with function COPYFILE, replacing all incidences of search-string with replace-string.	-
Helper	ATTIME(seconds, obj)	Calculates the object every day at specified times in seconds (seconds represents time fraction of the current day in seconds).	-
Helper	FROMASCII(num)	Returns a string that has the numeric value specified as num.	-
Helper	SLEEP(ms)	Pauses the current thread for a specified time in milliseconds (ms).	-