



FORCE EDGE CONNECT

Version: 231009

Product Description



Document: Product Description - FORCE EDGE CONNECT



Release date: 09.10.2023



Document version: 1



Author: FORCAM GmbH

Product Overview

FORCE EDGE CONNECT (hereafter simply referred to as EDGE CONNECT) offers manufacturing companies a solution for digitally connecting their heterogeneous machinery. Almost all machines can be digitized with EDGE CONNECT, regardless of age or technical status. Thus, FORCAM supports the digital transformation of a manufacturing plant in the Brownfield environment.

FORCAM therefore delivers a product that solves the main requirement of Industry 4.0 by extracting digital information from the production machinery. This closes the gap between IT (information technology) and OT (operative technology).

EDGE CONNECT offers a variety of possible methods to connect assets, it sends the asset signals to superordinate systems by means of standardized events. These can be ME (Manufacturing Execution) or MOM (Manufacturing Operation Management) systems such as SAP DM/ME or MII, among others. FORCAM can thus reduce the time and effort required for digitization and create a standardized interface to the machine park.

The machines are connected via an innovative plug-in concept. Common manufacturer-specific (proprietary) protocols are supported (such as HEIDENHAIN, Siemens S7 or FANUC & Co) as well as common communication standards like MTConnect, OPC UA or MQTT. The FORCAM I/O Controller is available as separate hardware for digitizing machines that are not network-capable. EDGE CONNECT is continually expanded with plug-ins to meet the challenge of digitally mapping every type of machine via the EDGE solution.

The asset connections are used to obtain a wide variety of information. This includes information about the current status of the assets, or their sensor readings such as temperatures, pressures or energy consumption. In the Brownfield as well as in the Greenfield environment, it is important not only to read the signals and pass them on, but also to interpret them for further processing. This task is performed by the Signal Composition component. This makes it possible, for example, to find out whether an asset is currently in production or at a standstill. Another essential part of the solution is the handling of NC programs and the possibility to transfer them to and from asset.

The modern and cleanly structured menu navigation of EDGE CONNECT makes it possible to digitally connect machines in a quick and efficient way using the available control and signal information.

The Machine Repository (optional extension) makes it easy to create and use templates. Templates are used for recurring settings when connecting assets. They already contain all the important general information. Only individual information, such as IP address or serial number, must be added. With the MR, you can either define templates for asset connections or derive a template from an existing connection and use it to connect further assets of the same type. The template structure ensures a standardized connection of identical assets, thus enabling the comparison of assets of the same type. This further reduces the individual effort required to connect an asset, enabling the time- and resource-efficient implementation of digitization projects. FORCAM provides standard templates for common machines.

EDGE CONNECT is flexible and can be applied to any manufacturing company. The individual components of the solution can be positioned in different areas and provide benefits at each level.

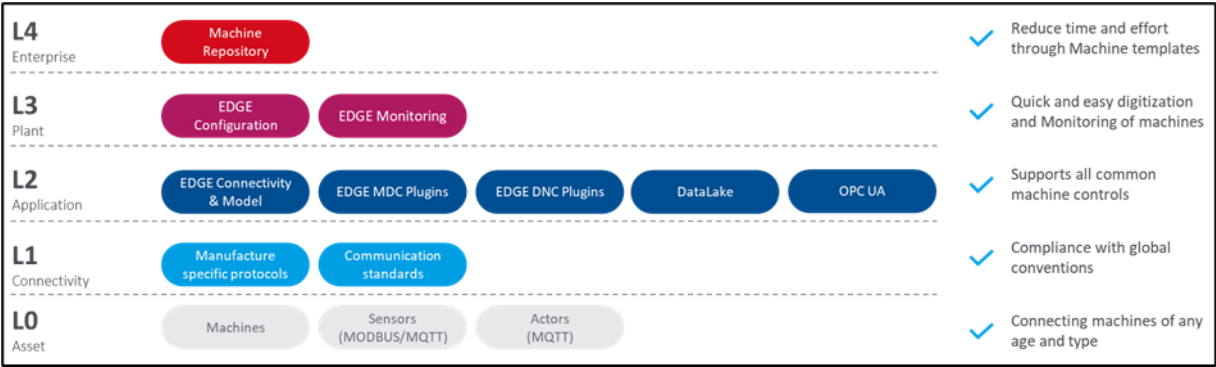


Fig. 1: Location of the EDGE CONNECT solution components

The figure shows the reference architecture of the Open Industry 4.0 Alliance, which is also the basis of the EDGE CONNECT architecture. FORCAM contributes significantly to digitalization in industry and focuses on customer benefits. The connectivity of hardware through intuitive and user-friendly software is what makes EDGE CONNECT stand out.

System Components

This chapter describes the individual EDGE CONNECT components and their functions.

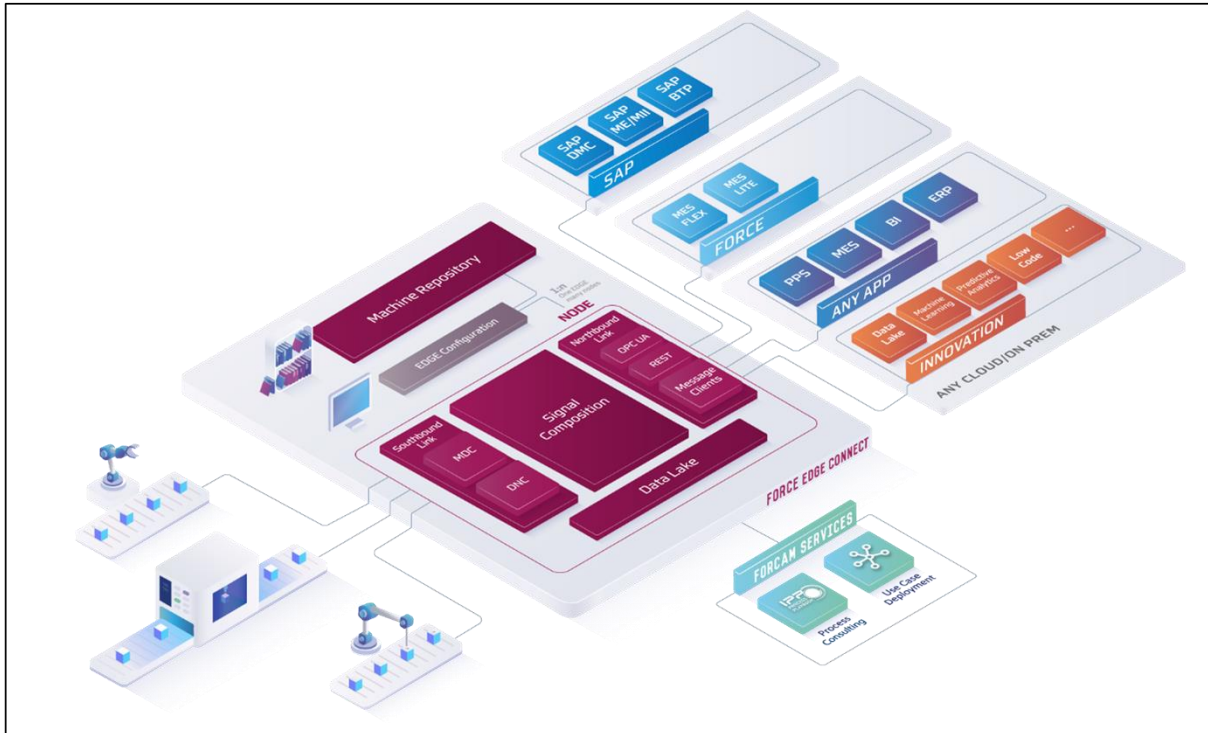


Fig. 2: Schematic structure of EDGE CONNECT

EDGE Node

The EDGE node is the central element of EDGE CONNECT when it comes to connecting assets. It consists of the following subcomponents:

Southbound Link

The Southbound Link component is responsible for the communication between EDGE CONNECT and the asset. In terms of infrastructure, EDGE CONNECT is located above the asset level (shopfloor). This is why we refer to the communication between assets and EDGE CONNECT as “southbound” communication.

Plug-ins

The plug-ins used in the EDGE CONNECT establish communication links with specific machine controllers. They also standardize the data, thus making evaluations more comparable.

They allow for direct communication with various asset controllers, but also cover modern communication protocols such as MQTT, OPC UA and many more.

The plug-ins are divided into those for Machine Data Collection (MDC) and for Distributed Numerical Control (DNC).

- MDC plug-ins for machine data collection

These include plug-ins designed for unidirectional reading of machine signals as well as plug-ins for bidirectional signal transmission, i.e., for reading and writing back signals.

- DNC plug-ins for transferring and reading NC files
These plug-ins can be used to transfer NC programs to the machine's file system or to query the program that is active on the machine.

⚠️ EDGE CONNECT is not intended to be used for providing, editing, or managing NC programs.

For the most common control types, a set of plug-ins is included in EDGE CONNECT by default. The annex contains a list of all FORCAM plug-ins that are currently available.

Signal Composition

This component is used to derive logical asset states. This allows standardized events to be derived from signal combinations. Events are messages that are sent to a third-party system. Signal composition also makes it possible to react to events and to write values to the control unit of the asset (if this is supported by control unit and protocol). Such a composition can be implemented in EDGE CONNECT either via a script or using a graphical solution. The graphical composition provides an easy introduction into the world of signal composition.

Northbound Link

- ⚠️ The Northbound Link component is responsible for the communication between EDGE CONNECT and any 3rd party system. In terms of infrastructure, the 3rd party system is located above EDGE CONNECT. This is why we refer to the communication between EDGE CONNECT and 3rd party systems as “northbound” communication.

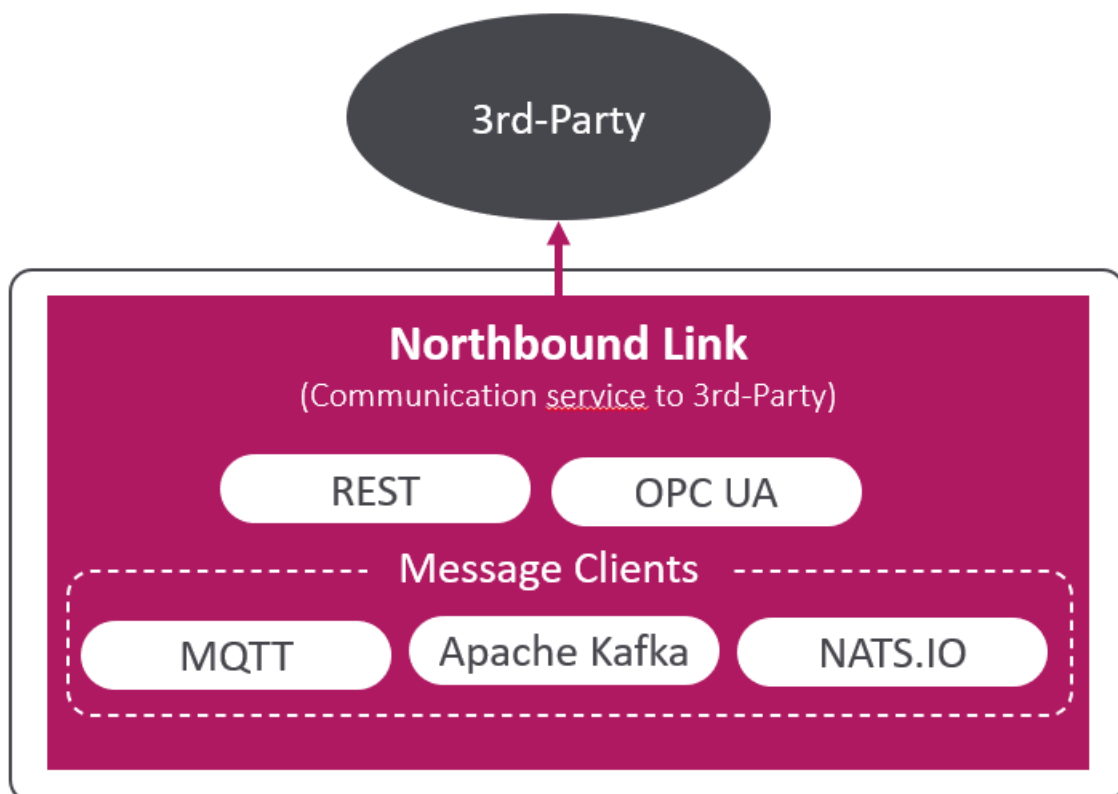


Fig. 3: Northbound Link

The Northbound Link component is used to forward asset data to superordinate systems (3rd party systems) in the form of standardized events. The following options are available for connecting superordinate systems:

- HTTPS/REST
- MQTT
- Apache Kafka
- OPC UA
- NATS.io


The message content can be configured individually for each connection and event. If MQTT, NATS.io or Apache Kafka are used, a broker is required as middleware.

The EDGE API is delivered with preconfigured standard events for communication with the MES or ERP level. If necessary, these can be further individualized.

The middleware must be provided and configured separately. It is not part of the EDGE CONNECT.

Data Lake

To obtain a digital twin of an asset or control unit, it is not only important to establish the connection to the asset, interpret the signals and pass them on to other applications, but also to store the data. With the Data Lake component, all data is stored at the signal level, the interpretation level and the event level, including configuration changes, write operations and transferred NC files. Data is made available via the Data Lake API. This allows the latest AI algorithms, visualization tools, but also audit requirements to benefit.

 The Data Lake component must be purchased separately, in addition to EDGE CONNECT.


EDGE Configuration

EDGE Configuration is the management interface for EDGE CONNECT. It can be used to manage multiple EDGE nodes. An EDGE node is the bundling of signal collection from several assets.

Depending on the amount of data, one or more EDGE nodes are used per plant. The management of the nodes is done centrally.

Machine Repository

The Machine Repository allows templates to be generated from existing asset connections or for new ones. These templates can be used to connect assets of the same type and the same usage type in a standardized manner. The template contains all configuration elements that are not machine-specific. Asset-specific/asset connection-specific configuration elements are, for example, IP address, serial number, equipment number, etc. By using an existing template, the time required to connect an asset is significantly reduced. In addition, templates lead to a standardized and unified asset configuration, which makes data more comparable when it comes to evaluation.

 The Machine Repository component must be purchased separately, in addition to EDGE CONNECT.

Configuration

The configuration of an edge node as well as a machine is done completely in the EDGE Configuration of EDGE CONNECT. The user-friendly interface will guide you through all relevant settings and shows all nodes and the statuses in the overview.

EDGE node

EDGE CONNECT lets you add nodes in just a few steps. An EDGE node corresponds to an instance of a connection variant. There can be several nodes per plant. They are logically bundled so that the machine workload is distributed efficiently.

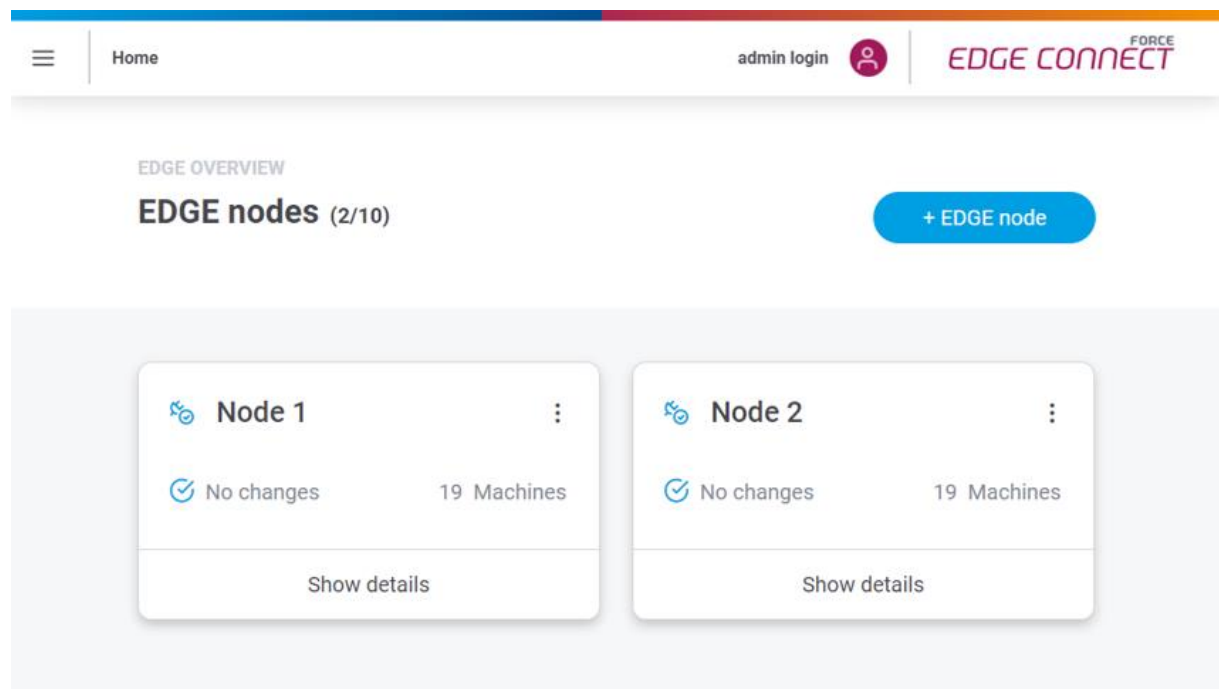


Fig. 4: EDGE CONNECT entry and overview page

The machine overview gives a description of the plant and indicates the status of each connected machine.

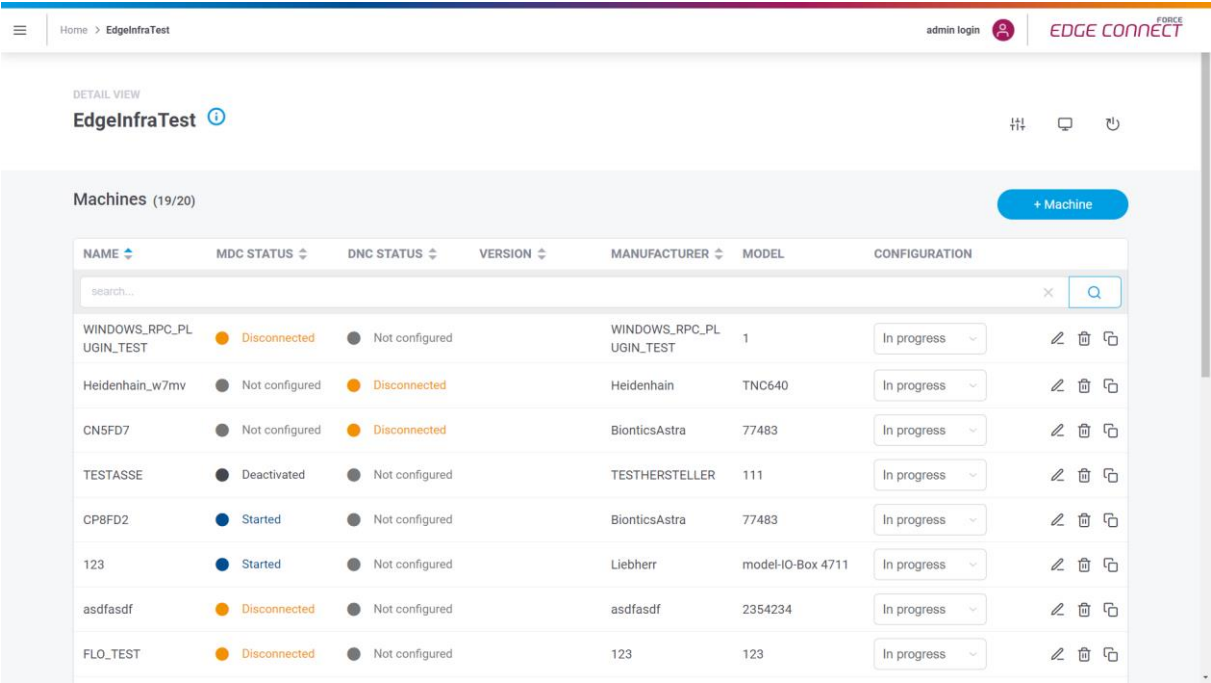


Fig. 5: Machine overview as next page after clicking on "Show details"

Machines

The dialog for adding a machine guides you through eight steps necessary for a connection. This is where MDC/DNC controls are configured and machine signals are defined, among other things.

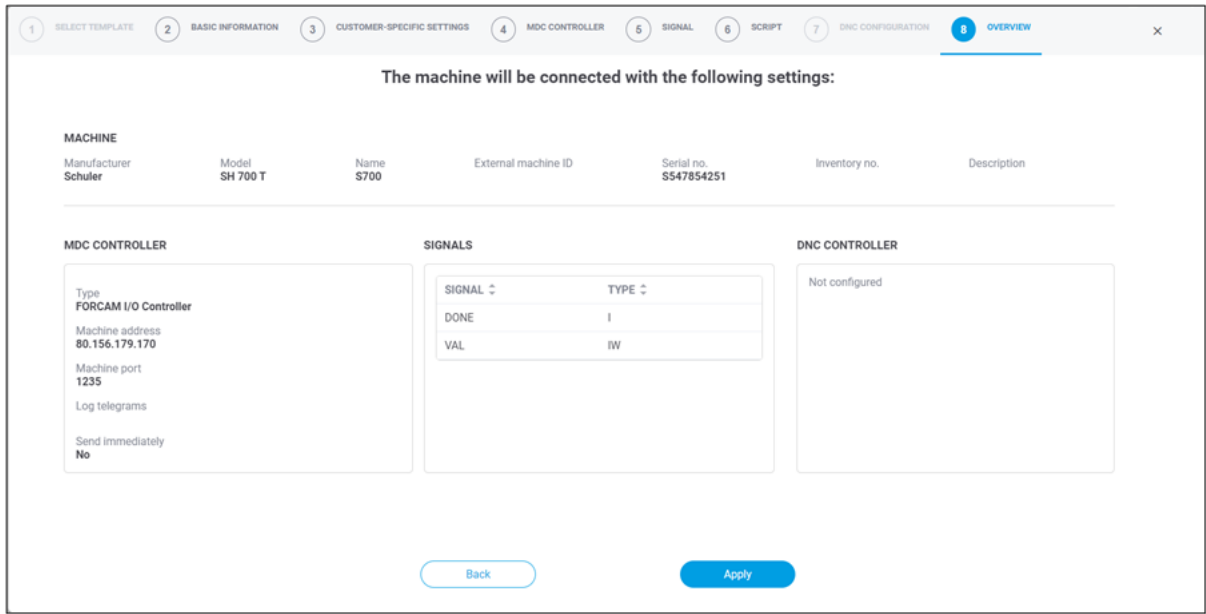


Fig. 6: Dialog for configuring a machine in EDGE CONNECT

Events

Events are used in a script to trigger outgoing events. For this, there are script functions available that generate a corresponding event depending on the type.

The configuration of the Northbound interface specifies the way the signals are sent to a superordinate system. Payload and endpoint are predefined by default, but they can be customized.

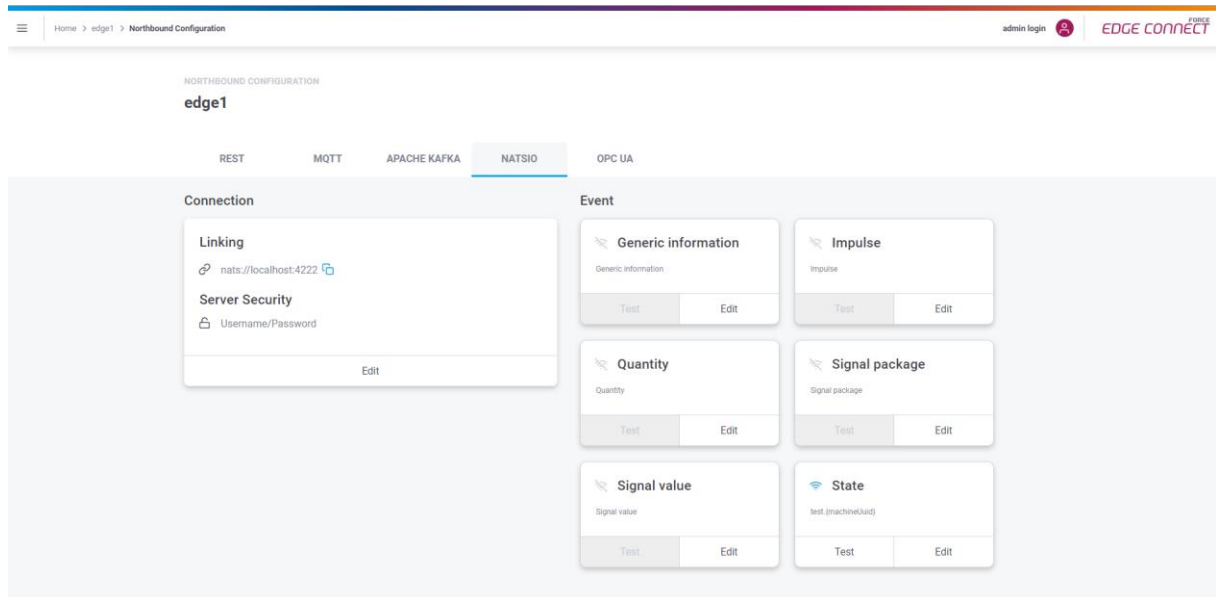


Fig. 7 Event configuration at the Northbound Link

For each type of event there is a standardized **Event**. For example, the **Quantity** event type sends the quantity produced by the machine.

Script functions allow events to use **Placeholders** (wildcards) which can be used to transfer different types of information. This can be used, for example, to get the machine ID or the time stamp formatted in UTC.

Monitoring

EDGE CONNECT provides the option to monitor individual components and extensions via the monitoring page. The page indicates whether a component is running without errors or if there are any malfunctions. Error messages and logs can be retrieved specifically for each component.

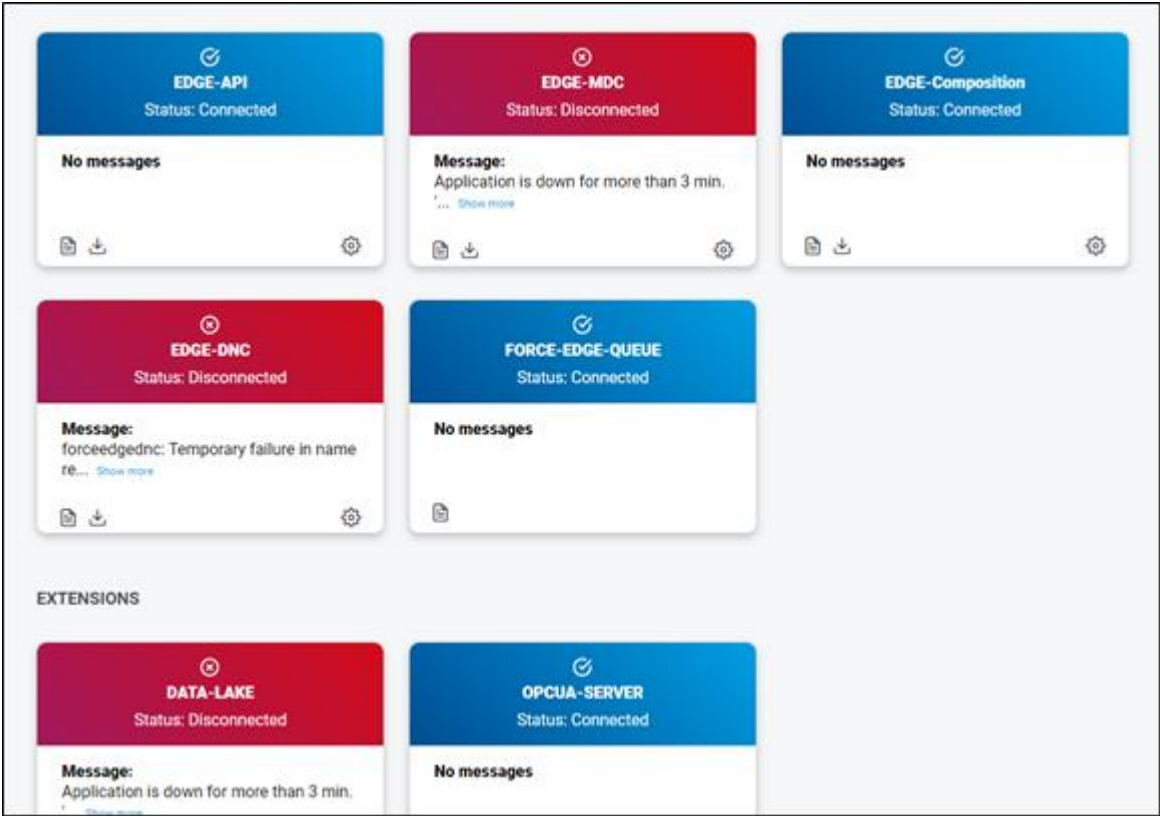


Fig. 8: Monitoring in EDGE CONNECT

Scope of functions

General

- Simple, digital connection of a heterogeneous machine park without great effort
- Almost all machines can be digitized, regardless of age or type
- Configuration of EDGE nodes and machines via modern menu navigation
- Clearly structured and user-friendly dialogs for easy machine connection
- Flexible use of EDGE CONNECT for any manufacturing company
- Substantial contribution to the Open Industry 4.0 Alliance and thus to digitization in the manufacturing industry with a focus on customer benefits


Machine signals

- Transmission of standardized events to superordinate ME or MOM systems such as SAP DM/ME or MII
- Reading machine signals to interpret when a machine is in production or at stoppage
 - Writing machine signals
 - Obtaining a wide variety of information from the machine such as current status or sensor readings such as temperatures, pressures or energy consumption
- Five technical options to supply signals and events from an EDGE-node to a third-party application:
 - HTTPS/REST
Any REST endpoint can be served. The HTTP methods POST and PUT are supported.
 - MQTT messaging
Any MQTT broker can be served, if provided by the customer or partner.
 - OPC UA
FORCAM provides an OPC UA server with “Data Access” functionality. This extension makes various machine data available via the defined OPC UA interface. The information models are prepared dynamically based on the existing configured machines in the EDGE node.
You connect to the server via the specified URL to retrieve the desired data. We assume that the client required for data retrieval already exists.
 - Apache Kafka
Any Apache Kafka broker can be served, if provided by the customer or partner.
 - NATS.io
Any NATS.io server can be served, if the infrastructure is provided by the customer. Messaging can be implemented using Core NATS or JetStream.

Plug-ins

- Innovative plug-in concept
 - Direct communication with different machine controls
 - Common machine manufacturer-specific protocols supported (e.g., HEIDENHAIN, Siemens S7, FANUC)
 - Common communication standards supported (e.g., MTConnect, OPC UA, MQTT)
 - FORCAM I/O controller as separate hardware for digitally connecting non-networked machines

- Transfer of NC programs from and to the machine

 The plug-in concept is extensible, FORCAM is continuously expanding the number of supported plug-ins.

EDGE API

- Fetching machine master data and configuring machine connections via RESTful API
- Forwarding machine data to superordinate systems in the form of standardized events using the EDGE API event service
- Connecting superordinate systems via HTTPS/REST, MQTT, Apache Kafka OPC UA or NATS.io
- Transmitting events via HTTP in JSON format (optionally: MQTT broker as middleware)
- Communication with the MES or ERP level through preconfigured standard events in the EDGE API

Monitoring

- Monitoring individual components via the monitoring option
- Real-time information about the respective component with status indication
- Retrieving error messages and downloading logs

Optional components

Data Lake

- Retrieval and storage of signal and event data, configuration changes, writing operations and transmitted NC files

Machine Repository

- Easy to create and use machine templates via the Machine Repository component
 - Definition of templates for machine connection
 - Deriving templates from existing connections to be applied to the same types of machines
 - Significant reduction of the individual effort for connecting a machine
 - Standardized connection of identical machines, which enables the comparison of machines of the same type

Annex

MDC Plug-ins

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

MDC plug-in	Read	Write	Transmission: Polling/Event-based
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 MCIS RPC	X		X/X
Siemens LOGO!	X		X/
WAGO 750	X		
Weihenstephan	X		X/
Wiesemann & Theis (WUT)	X		X/

DNC Plug-ins

DNC plug-in	Read	Write
COM	X	X
External program file transfer (Preview version)	X	X
FANUC	X	X
File Handler (File Copy)	X	X
File Handler Server	X	X
FTP Plug-in	X	X
Heidenhain	X	X

DNC plug-in	Read	Write
Mazak	X	X
Mitsubishi	X	X
MOXA-Box	X	X
RPC (Preview version)	X	X
WUT (Preview version)	X	X