



FORCE EDGE CONNECT Machine Repository System Requirements

Version 231009



Document: FORCE EDGE CONNECT Machine Repository -
System Requirements



Release date: 2023-09-27



Document version: 2



Author: FORCAM GmbH

Content

Disclaimer	3
Docker Technology	4
System Requirements (Customer Responsibility)	5
Machine Repository Minimal Hosting Requirements	5
User Rights for Installation	6
EDGE Architectural Concept	7
Network Requirements	8
General Network Requirements	8
Ports/Firewall	8
HTTPS Certificates	9
Installation of the EDGE CONNECT Installer	9
EDGE Configuration in the MR.....	10

Disclaimer


The data for the EDGE is based on standardized scenarios. This scenario assumes 50 machines per node and approx. 30 signals per machine. Other requirements may be necessary for customer-specific scenarios and IT landscapes. In this case, our consultants will discuss these with the relevant IT departments to plan a possible hardware environment.


The disc space given in this document are the minimal hosting requirements for the machine repository. It may occur that specific projects need adjusted requirements.

Some features like the "Machine Repository" mentioned below are not included in the standard EDGE package and must be purchased separately.

Docker Technology

- For productive use of FORCE EDGE CONNECT, we recommend an Ubuntu operating system with Docker runtime environment for performance reasons.
- Alternatively, we assume Windows Server 2019 (Hyper-V + Ubuntu VM + Docker)
- The provision, installation and maintenance of the Docker runtime environment is the responsibility of the customer.

 FORCAM does not assume any warranty regarding stability and performance for the Docker runtime environment!

 Windows desktop operating systems are not supported.

System Requirements (Customer Responsibility)

Machine Repository Minimal Hosting Requirements

The minimal hosting requirements for the possible Machine Repository servers are as follows.

Ubuntu

- Ubuntu = 20.04.3 LTS or a newer version of Ubuntu 20.04.
- Docker Engine \geq 20.10.22
- Docker Compose version \geq V2.14.1
- OpenSSL version: \geq OpenSSL 1.1.0l
- AdoptOpenJDK 11 64 Bit Version \geq 11.0.11

Infrastructure

- \geq 4GB RAM
- Dual-Core processor $>$ 2Ghz
- Minimum 30 GB free SSD disc space

Windows with Ubuntu VM

- Windows Server 2019 (\geq Version 1809)
- Hyper-V must be enabled in Windows
- Ubuntu 20.04.3 LTS (or a newer version of Ubuntu 20.04.) VM in Hyper-V:
 - Docker Engine = 20.10.22
 - Docker Compose Version = V2.14.1
 - OpenSSL version: \geq OpenSSL 1.1.0l
 - AdoptOpenJDK 11 64 Bit Version \geq 11.0.11

Infrastructure of the UBUNTU VM:

- \geq 4GB RAM (must be assigned to the VM)
- Dual-Core processor $>$ 2Ghz (must be assigned to the VM)
- Minimum 30 GB free SSD disc space (must be assigned to the VM)

Client Requirements

For the Browser: Google Chrome or Microsoft Edge with Chromium engine.

User Rights for Installation

It is mandatory to create a non-root user.

The rights must be assigned to the user to be able to install correctly.

User

- Must be a non-root user (name doesn't matter)
- This user must have the right to run the docker + docker compose command
- User id while installing must be the id of the current user (number):
 - o `"id"` (command)
- Add the current user into the docker group:
 - o `"sudo usermod -aG docker $USER"` (command)

EDGE Architectural Concept

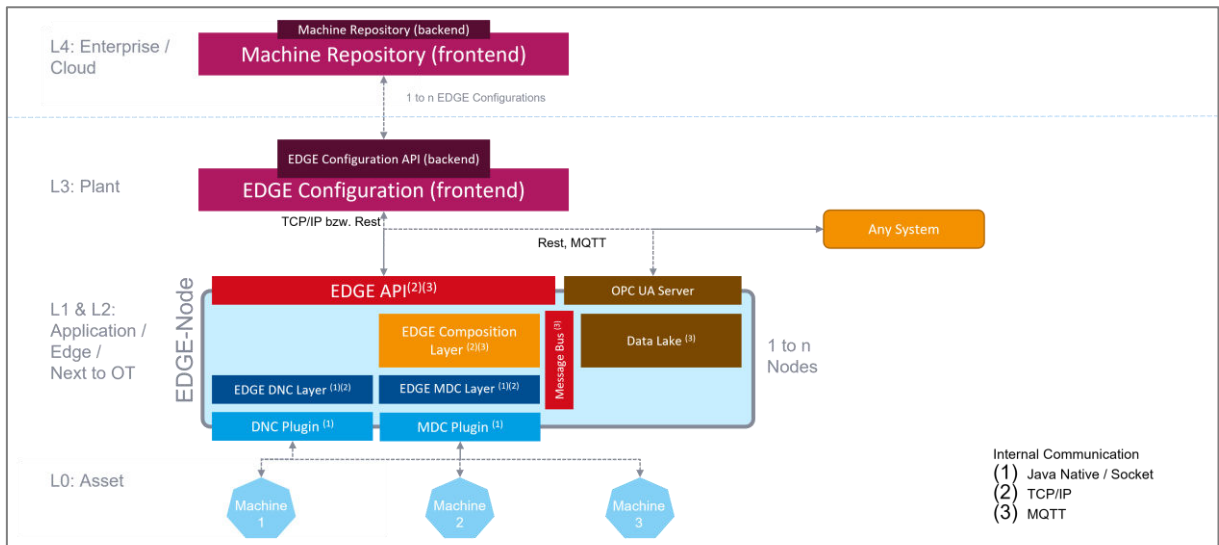


Fig. 1: Basic EDGE architectural setup for the use of additional nodes

- L4:** Machine Repository as a central component with various templates for managing the machine park
- L3:** User interface with a graphic representation of FORCE EDGE CONNECT and the user administration
- L2:** Composition layer for data processing with MDC and DNC plugins, Data Lake, and OPC UA
- L1:** Communication software component that can be accessed using plugins
- L0:** All assets to be connected

Network Requirements

General Network Requirements

- Active Internet connection
- ❗ If an online license is involved, the following URLs must be available:
Primary: <https://qlm1.net/forcam/qlmlicensesserver/qlmservice.asmx>
Fallback: <https://qlmdr.com/forcam/qlmlicensesserver/qlmservice.asmx>

Ports/Firewall

Ports that must be open:

- 60077 https UI
- 60066 https FORCE EDGE UI

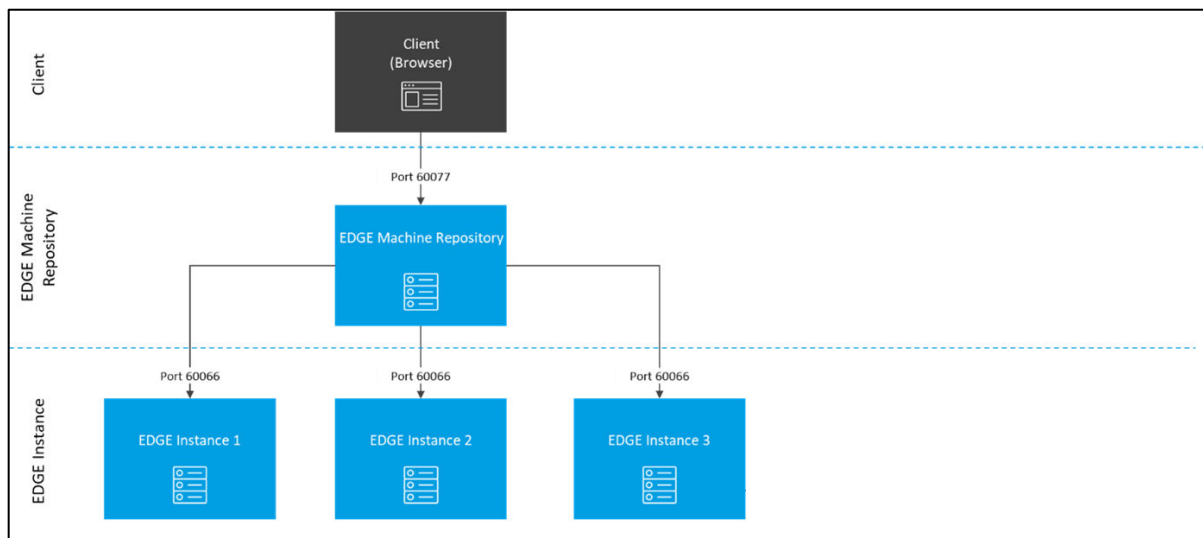


Fig. 2: Communication of EDGE CONNECT Machine Repository

HTTPS Certificates


Installation of the EDGE CONNECT Installer

- Trusted certificates or self-signed certificates are possible
Must contain host name; IP address is not supported


New certificates must be provided for each update.


The certificate must have the following PEM format:

```
-----BEGIN CERTIFICATE-----  
<<Public key>>  
-----END CERTIFICATE-----  
-----BEGIN PRIVATE KEY-----  
<<Private key>>  
-----END PRIVATE KEY-----
```

-  FORCAM is not responsible for generating or maintaining the certificates. Customers have to provide the necessary certificates.


An individual certificate must be provided for each EDGE component (EDGE Configuration, EDGE Node or Machine Repository). It is possible to use the same certificate for each EDGE component.

 We strongly recommend not to use a wildcard certificate.

 Although it is possible to use the same certificate for each EDGE component, we recommend using different certificates for each EDGE component.

No encrypted certificate files (PKCS 12 format) can be used.
Only files using the PEM format are allowed.

Whenever a web client (for example, a web browser) connects to the UI using a self-signed TLS certificate, a warning informs the user that the website could not be verified as a trusted website. Depending on the browser type, there are settings to suppress these warnings. Note, that suppressing the warnings may lead to insecure connections.

-  To get the host name from the Ubuntu system, run below command:
hostname -A

EDGE Configuration in the MR

During the addition of an EDGE Configuration please use a public certificate that has to have the following format:

```
-----BEGIN CERTIFICATE-----  
<<Public key>>  
-----END CERTIFICATE-----
```