Critical manufacturing
an ASMPT company

Critical manufacturing 11.1

System Requirements

January 2026

DOCUMENT ACCESS

Restricted

Table of Contents

Summary of Tables

Summary of Figures

1 System Requirements

Critical Manufacturing MES is a high availability Manufacturing Execution System optimized for transaction throughput with critical system requirements. For an overview of the system and its main components please refer to the [[installation-guide-systemarchitecture]] section.

The purpose of this document is to provide the user with the system requirements and different configuration options for running Critical Manufacturing MES in different environments.

Below, follows a segmented view of the necessary information to prepare beforehand for a future acquisition and installation of this software.

- Deployment
  - [[system-requirements-deploymenttargets|Available Deployment Targets]]
- Data Storage
  - [[system-requirements-persistency-layer-index|Overview]]
  - [[system-requirements-database-layer-index|SQL Database Requirements]]
  - Other External Data Storage Requirements
- Application Backend
  - [[system-requirements-application-layer-index|Overview]]
  - [[system-requirements-application-layer-applicationlayercontainers|Containers Stack Requirements]]
  - [[system-requirements-application-layer-applicationlayeroptional|Optional Components Requirements]]
- Automation Components
  - [[system-requirements-connectrequirements]]
- Clients
  - [[system-requirements-clientdevices|GUI Client Devices Requirements]]
- Additional Requirements
  - [[system-requirements-otherrequirements]]

⚠ The exact hardware requirements depend on the specific customer scenarios and environment.

2 Deployment Targets

Critical Manufacturing MES offers flexible deployment options to suit various needs and IT strategies. The below sections present the deployment options and major considerations when establishing your strategy.

2.1 Deployment Options

2.1.1 Cloud

MES may be deployed on cloud environments using:

- **Public Cloud** - Leveraging platforms like Microsoft Azure or AWS.
- **Private Cloud** - Hosted within the on your own data center or a dedicated private cloud environment.

When deploying Critical Manufacturing on a cloud infrastructure, Critical Manufacturing recommends a containerized approach, targeting a managed Kubernetes service like Azure AKS or Amazon EKS. In the Critical Manufacturing Information Center, there are detailed guides on how to configure a Kubernetes cluster.

For cloud-based installations, the following concerns should be taken into account:

Table 1: Cloud-based installation concerns

Topic	Description
Latency	In order to ensure adequate user experience and performance, it's recommended that the latency between the application clients (web browser or mobile devices) and the application servers does not exceed **200ms**.
Bandwidth	In order to ensure adequate performance of the user interface, a minimum throughput of **20 Mbps** between the application clients (web browser or mobile devices) and the application servers is recommended.
Connection availability	Cloud-based installation rely on a stable connection to the data center hosting the application backend. The system does not offer any offline capability in case there is no connectivity to the cloud.
Database and application servers co-located	In order to ensure application performance, database servers running Windows Server VMs should be co-located in the same region to ensure low latency.
On-premises components	Components related with automation are recommended to be executed on premises, due to lower latency between equipment controllers and the physical equipment to which they are connected. Connectivity is required between on-premises components and the application servers.

2.1.2 On-Premise

Installing and running the MES directly on your local servers and infrastructure is also an option. On such scenario, evaluate if your deployment is a:

- **Single-plant deployment** - For a single manufacturing facility.
- **Multi-plant deployment** - Connecting and managing MES across multiple manufacturing sites.

This may impact a lot of decisions when setting up your infrastructure.

2.1.3 Hybrid

A combination of cloud and on-premise deployments, where the MES might utilize the cloud while Connect IoT Automation Managers remain on-premise, close to the shop floor.

2.2 Considerations

2.2.1 General

When choosing your deployment target evaluate:

- **IT Infrastructure Readiness** - Ensure the chosen environment (cloud, on-premise or hybrid) has the necessary hardware, software, network connectivity, and cybersecurity measures in place.
- **Scalability Requirements** - Assess the current and future production volumes and the need for the MES to handle increasing data and user loads.
- **Security and Compliance** - Evaluate data security requirements and any industry-specific regulations that might influence the choice between cloud and on-premise options.
- **Integration with Existing Systems** - Considering how the MES will need to connect and exchange data with other enterprise systems like ERP, PLM, and automation systems.
- **Company IT Strategy** - Align the MES deployment with the overall IT strategy and preferences of the manufacturing organization.
- **Disaster Recovery and Business Continuity** - Plan for data backup, system redundancy, and failover capabilities in the chosen environment.
- **Latency and Performance** - For real-time data collection and control, especially in highly automated environments, the proximity of servers (on-premise vs. cloud) can be a crucial factor.

2.2.2 MES Specifics

Below are listed the Critical Manufacturing MES specific functions that may influence your deployment target decisions.

Deployment Process

Critical Manufacturing MES environment is managed through **Critical Manufacturing DevOps Center** (a centralized deployment platform for containerized applications). This service allows you to create, configure and automate the installation, upgrade and uninstall MES environments.

The DevOps Center provides different container orchestration platforms. For each one there are one or more deployment targets to directly deal with different vendors or deployment strategies (On-Premises, Hybrid and Cloud). Also, it provides two deployment processes:

- **Remote installation** - First you configure and deploy Critical Manufacturing Infrastructure Agent into your infrastructure. The agent will then be responsible by the communication between DevOps Center and your infrastructure. This will allow you to install, upgrade and uninstall the environments just by interacting with this service GUI or CLI.

- **Local installation** - After creating and configure the environment, you will be allowed to download a package with a recipe (scripts and configuration files) required for you to manually run the setup on your environment.

For more information, see DevOps Center deployment targets documentation.

2.2.3 Database Layer Requirements

Critical Manufacturing MES requires a SQL Server installation running on Windows Server. For high-availability purposes, an Always On High Availability cluster configuration is recommended.

SQL Server for Linux or containerized versions of SQL Server are currently **not supported** for full MES workloads, including Operational Data Store (ODS) and Data Warehouse (DWH) transactions, since they offer several limitations on components required by the system. Therefore, the database component is required to be hosted on servers running Windows Server.

Managed SQL Server services like Azure SQL Database or Amazon RDS are also **not supported** due to features that are currently unavailable in those services.

There is **limited support** for Azure SQL Managed Instances. In this case, SQL Server Analysis Services (SSAS) and Reporting Services (SSRS) must be hosted separately. Please consult Critical Manufacturing for additional details on how to configure an installation using Azure SQL Managed Instances.

For additional details on the required configuration of the database component, see the Database Component section.

2.2.4 Application Layer Requirements

Critical Manufacturing MES application layer is deployed as a containerized application on one of the following platforms:

- Kubernetes (check [[system-requirements-application-layer-applicationlayercontainers#software-requirements]] for the supported containers)
- Red Hat OpenShift
- Cloud managed container orchestration services based on Kubernetes (Azure AKS, and Amazon EKS)

The deployment process for containerized installations is managed through **Critical Manufacturing DevOps Center**.

For additional details on the required configuration for a containerized deployment, see the Application Layer - Container Stack section.

2.2.5 Optional Components Requirements

In addition to the containerized components, there are currently Application Layer optional components which, due to dependencies to Windows-specific libraries, are required to run on Windows servers. The table below describes these components and the associated modules.

Table 2: Optional components and associated modules

Component	Module	Description
Printable Documents Renderer	Advanced Layout and Printing	On containerized environments, this component supports printing documents to Windows printers. It is also possible to configure the container infrastructure to enable printing documents using a [[operation-guide-environmentvariables#cups_server
ECAD Renderer	Core	Required to render CAD file visualizations.

Component	Module	Description
IoT Automation Managers	Core	Required to support Windows-based automation drivers.

3 Persistency Layer

This section describes the contents of the persistency layer that will hold the database and other additional required components for running Critical Manufacturing MES:

- Database Component
  - SQL Server Always On
  - Storage Area Network Configuration
  - SQL Server Licensing
  - SQL Server Permissions
- External Components

4 Database Component

Critical Manufacturing MES uses SQL Server databases for persistency. This section describes the requirements for the database component, including software and hardware requirements, as well as notes about SQL Server high availability, storage and licensing.

4.1 Software Requirements

The table below describes the software requirements for database servers:

Table 3: Software requirements for database servers

Operating System	Database Engine	Additional Required Software
• Windows Server 2022	• SQL Server 2019 Standard Edition • SQL Server 2019 Enterprise Edition • SQL Server 2022 Standard Edition • SQL Server 2022 Enterprise Edition	• SQL Server Reporting Services • SQL Server Analysis Services (multidimensional model) • Microsoft Distributed Transaction Coordinator (MSDTC)

4.2 Hardware Requirements

Depending on the expected system load, the recommended hardware configuration for the database servers may vary. This section outlines reference configurations for different use cases.

Table 4: Hardware configuration for database servers

Configuration	Intended Use	Response time	Scalability	High availability
Minimum	Demonstration / Development	Medium/Slow acceptable	Not required	Not required
Training / Staging	System tests, validations and training	Medium	Not required	Not required
Production A	Low to medium volume production	Fast	Required	Very High
Production B	High volume production	Fast	Required	Very High

The exact hardware requirements will depend ultimately on the specific customer environment. Consult with Critical Manufacturing in order to get a tailored recommendation based on known requirements for your specific installation. These are the minimum requirements:

Table 5: Minimum hardware requirements for database servers

Configuration	Number of Servers	Processors	Memory	Storage
Minimum	1 *	1 x Quad Core, 2 GHz	8 GB	200+ GB
Training / Staging	1	2 x Quad Core, 2 GHz+	16 GB	Online DB: 200+ GB ODS+DWH: 1TB
Production A	2	2 x Six Core, 2 GHz+	32 GB+	Internal storage: 150+ GB Storage hard-disk: 2.5TB
Production B	2	2 x Eight Core, 2 GHz+	64 GB (Active/Active) 128 GB (Active/Passive)	Operating System: 150 GB SAS 15K rpms TempDB: 200 GB+ in SSD Storage: 10 TB+

The Minimum configuration assumes that the database server may also run other application components in addition to the database.

4.3 SQL Server Configuration

The sections below offer additional details on SQL Server configurations that are relevant to a Critical Manufacturing installation.

- SQL Server Always On
- Storage Area Network Configuration
- SQL Server Licensing
- SQL Server Permissions

5 SQL Server Always On

Always On is a feature that provides a low-cost alternative to a Storage Area Network (SAN) as each Database Server uses its own local storage as shown in the figure below. The Always On feature is supported in the SQL Server Enterprise and Standard Editions (the Standard Edition does not support Availability Groups and is limited to two nodes Failover Cluster Instances). More information about SQL Server Always On can be found in the following URLs:

- SQL Server AlwaysOn Failover Cluster Instances
- SQL Server AlwaysOn Availability Groups

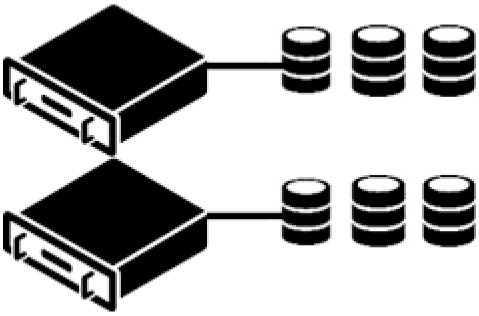


Figure 1: SQL Server Always On

5.1 SQL Server Always On Failover Cluster Instances

The Always On Failover Cluster Instances leverages Windows Server Failover Clustering (WSFC) functionality to provide local high availability through redundancy at the server-instance level - a failover cluster instance (FCI). A FCI is a single instance of SQL Server that is installed across different Windows Server Failover Clustering (WSFC) nodes and, possibly, across multiple subnets. On the network, an FCI appears to be an instance of SQL Server running on a single computer, but the FCI provides failover from one WSFC node to another if the current node becomes unavailable.

5.2 SQL Server Always On Availability Groups

The Always On Availability Groups feature is a high-availability and disaster-recovery solution that provides an enterprise-level alternative to database mirroring. Always On Availability Groups maximize the availability of a set of user databases for an enterprise. An availability group supports a failover environment for a discrete set of user databases, known as availability databases, that fail over together. An availability group supports a set of read-write primary databases and one to eight sets of corresponding secondary databases. Optionally, secondary databases can be made available for read-only access and/or some backup operations.

5.3 SQL Server Always On in Multi-Subnet Setups

Critical Manufacturing MES v{{extra.current_version}} supports SQL Server Always On across multiple subnets provided the following requirements are met:

- Use Microsoft Active Directory (AD) as the network's DNS server.
- Set `RegisterAllProvidersIP=0` on each Availability Group listener (WSFC), and adjust `HostRecordTTL` if suitable.

- The `MultiSubnetFailover` flag remains unset/false (this is the default MES configuration).

With these settings, the listener registers only the active IP, enabling reliable client connections and cross-subnet failover without `MultiSubnetFailover`.

ⓘ Support for `MultiSubnetFailover` is planned for a future MES release.

5.3.1 Microsoft Active Directory Alternatives

Using other DNS servers or different architectures may also work, assuming they follow the previous requirements.

⚠ Keep in mind that any alternative configurations are your responsibility and need to be managed and maintained independently.

Among the alternatives are:

- Using other DNS services may work as long as they implement the RFC 2136, allowing dynamic DNS updates. However, these may require custom configuration to ensure that the WSFC updates their DNS entries when performing failovers.
- Other cloud-native services - such as AWS Route 53 or Azure Private DNS - may work but are more complex to set up and typically require custom automation via provider APIs because they do not implement RFC 2136.
- Alternatively, you can leverage a Load Balancer that will only route traffic to the active replicas, assuming that it has healthchecking capabilities. The Availability Group Listener hostname must be resolved through any DNS Server to the Load Balancer's IP address. For more information, see the how to reduce failover times on AWS.

6 SQL Server Licensing

This section contains licensing information regarding SQL Server.

SQL Server Standard editions are licensed in Core-Based and Server + CAL modes, whereas Enterprise editions are licensed only in Core-Based mode.

When running Critical Manufacturing MES on SQL Server Standard edition, the following limitations apply:

- Database instance maximum hardware consists of 128 GB and 24 cores of CPU.
- Power BI Report Server is not available.
- Index rebuilding is only available in offline mode.
- Analysis Services Multidimensional has limited partitioning capabilities, severely impacting DWH cubes performance in large volume sites.

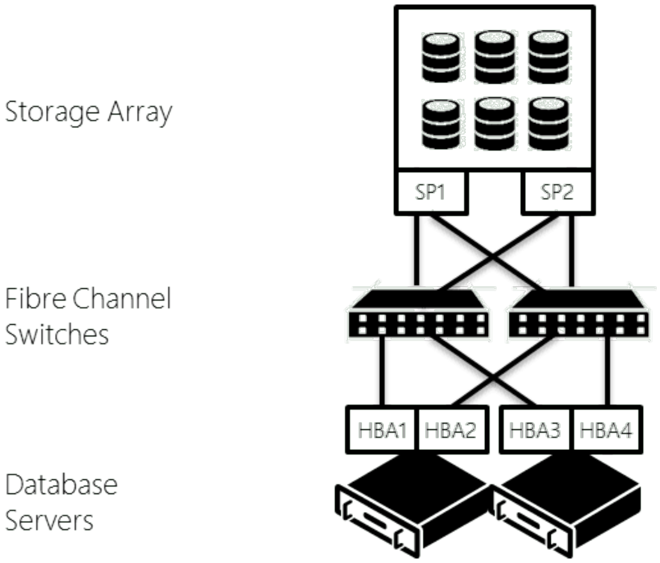
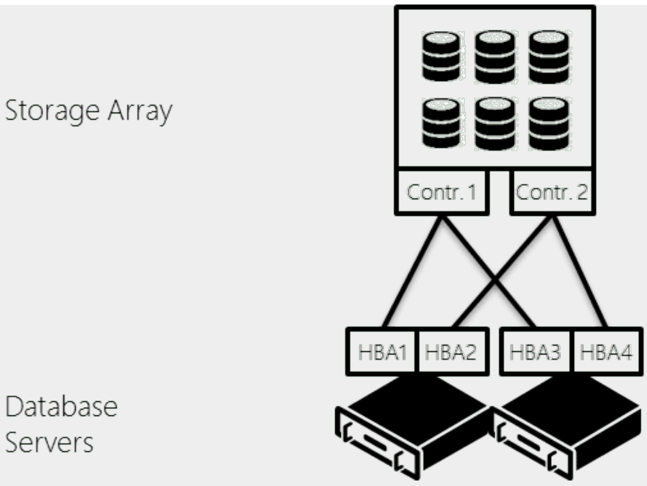
For more information about SQL Server licensing, please refer to the URLs below:

- SQL Server 2019
- SQL Server 2022

7 Storage Area Network Configuration

Critical Manufacturing recommends the usage of SQL Server Always On for productive installations of Critical Manufacturing MES. The usage of Storage Area Network (SAN) is also supported. This section describes the productive configuration of a Storage Area Network.

Table 6: Storage Area Network configuration

Type	Configuration
Storage does not support direct attach	 Storage Array SP1 SP2 Fibre Channel Switches HBA1 HBA2 HBA3 HBA4 Database Servers
Storage supports direct attach	 Storage Array Contr. 1 Contr. 2 HBA1 HBA2 HBA3 HBA4 Database Servers

8 SQL Server Login Permissions for MES Installation and Operation

This section provides essential information regarding the database user required to install and run the MES system.

To successfully install or upgrade MES to a newer version, the designated database user must be assigned the `sysadmin` role in SQL Server. This elevated permission is necessary to ensure that the installation or upgrade process completes without issues.

Once the MES system is fully operational, the `sysadmin` role can be removed from the database user, as it will no longer be required for regular system operations.

8.1 Removing `sysadmin` Role from Database User

This section outlines the necessary steps and permissions required to remove the `sysadmin` role from the database user, once the MES system is up and running.

⚠ This process involves removing the `sysadmin` role from the existing database user, **not replacing** the user itself.

By removing the `sysadmin` role, the database user will retain ownership of the MES databases and associated jobs. This ownership grants most of the necessary permissions to ensure that the system continues to operate correctly.

Therefore, before removing the `sysadmin` role from the database user, please ensure that the following conditions are met.

⚠ If the environment is configured with SQL Server Always On, the following configurations must be performed on all database servers.

8.1.1 Configure User Mapping for MES Linked Servers

To ensure proper functionality before removing the `sysadmin` role, you need to configure user mapping for the MES Linked Servers as follows:

- **Local Login**: The database user specified during MES installation
- **Impersonate**: No
- **Remote User**: The same database user specified during MES installation
- **Remote Password**: The password for the database user specified during MES installation

This user mapping must be applied to the following Linked Servers:

- `cm\{system\_name\}ODSLink`
- `cm\{system\_name\}DWHLink`
- `cm\{system\_name\}OnlineLoopback`
- `cm\{system\_name\}OnlineLink`

If the environment is configured with **SQL Server Always On**, the user mapping must also be applied to the following Linked Servers:

- `Online High Availability LinkedServer`
- `ODS High Availability LinkedServer`
- `DWH High Availability LinkedServer`

8.1.2 Configure Database User Permissions

Ensure that the database user is granted the following permissions:

- `VIEW SERVER STATE`
- `VIEW ANY DEFINITION`

8.1.3 Configure User Mapping for the `msdb` Database

To ensure proper operation of the MES system, the database user must have a user mapping to the `msdb` database. Additionally, the following permissions must be granted within the `msdb` database:

GRANT SELECT on the following tables:

- `dbo.sysjobhistory`
- `dbo.sysjobsteps`
- `dbo.sysjobs`
- `dbo.sysjobservers`

8.1.4 Trusted Assemblies

The MES databases include and utilize certain custom assemblies (using CLR) that provide various utility functions and procedures. To ensure these assemblies can be used without the `sysadmin` role, they must be added to the list of trusted assemblies for the server.

The following assemblies need to be added to the trusted assemblies list for each respective database:

- Online:
  - `OnlineDatabaseUtilsCLR`
- Operational Data Store (ODS):
  - `cmFoundationSQLServerCLRUtils`
  - `ICSharpCode.SharpZipLib`
- Data Warehouse (DWH):
  - `cmFoundationSQLServerCLRUtils`
  - `ICSharpCode.SharpZipLib`
  - `System.IO.Compression`

9 External Components

Critical Manufacturing MES has depends on external system, besides SQL Server database, that must be running prior to setup a new environment.The following subsections present these dependencies in two groups:

- **Mandatory**: systems required by the setup of a new MES environment;
- **Optional**: systems only required when a specific functionality of the MES is activated.

9.1 Mandatory

This section describes the requirements for additional required components that must be running for a successful installation and operation of Critical Manufacturing MES:

- AWS S3 (or compatible)
- ClickHouse for Data Platform
- Kafka
- RabbitMQ

The requirements will vary depending on the usage of each component.

9.2 Optional - Observability

If you intend to use the Critical Manufacturing Observability stack then the following component is required:

- ClickHouse for Observability

10 ClickHouse Requirements

ClickHouse serves as the next-generation analytics database for Critical Manufacturing MES, storing Operational Data Store (ODS), Canonical Data Model (CDM), and Data Warehouse (DWH) information in a high-performance columnar format.

Critical Manufacturing has pioneered the adoption of ClickHouse in manufacturing environments, leveraging technology typically used in telecommunications and banking sectors. Its columnar storage architecture makes it ideal for analytics workloads and high-cardinality data scenarios.

ⓘ "Learn More" For additional background on Critical Manufacturing adoption of ClickHouse, see Modernizing MES Data Analytics: A ClickHouse Journey post on the Developer Blog.

Its usage encompasses the following use cases:

- **Analytics Platform**
  - **ODS Storage** - Operational Data Store containing replicated MES transactional data.
  - **CDM Implementation** - Canonical Data Model for standardized data representation.
  - **DWH Capabilities** - Data Warehouse analytics and reporting functionality.
  - **KPI Calculations** - High-performance metrics including OEE, Rate Efficiency, and Performance Efficiency.
- **Online Database Functions**
  - **E142 Map Storage** - Used to store semiconductor equipment interface mapping data.
- **Future Applications**
  - **High-Cardinality Scenarios** - Potential future use for specialized online high-cardinality data requirements (as E142 maps).
  - **AI/ML Integration** - Exploring use cases for storing vector embeddings to support RAG and LLM.

10.1 Minimum System Requirements

Table 7: Minimum system requirements for ClickHouse

Item	Requirement
Version	24.3
Memory	16GB
CPU	4vCPUs
Disk space	1TB

10.2 Recommended System Requirements

To support the calculation of Rate Efficiency, Performance Efficiency, and OEE KPIs, it is necessary to update to ClickHouse version 25.3.

Table 8: Recommended system requirements for ClickHouse

Item	Requirement MES ≤ 11.1.4	Requirement MES ≥ 11.1.5
Version	24.8	25.3
Memory	32GB	32GB
CPU	8vCPUs	8vCPUs
Disk space	3TB	3TB

"Mandatory ClickHouse Settings" - **Cloud Upgrades**: When upgrading from MES versions earlier than 11.0.0 to 11.0.0 or later, and migrating from an in-stack ClickHouse deployment to an external ClickHouse Cloud setup, the `allow\_materialized\_view\_with\_bad\_select` setting must be configured before starting the upgrade. For more information, see ClickHouse Migration.

```
- **Correct ODS Behavior**: In ClickHouse, when the `do_not_merge_across_partitions_select_final`
    setting is set to `1`, merging across partitions may cause data duplication when querying ODS
    state tables (for example, `CoreDataModel_T_Material`). ClickHouse 25.3 sets this value to `0` by
    default; however, some operators may have manually overridden it to `1`. To ensure correct
    behavior, this setting must always remain set to `0`. Starting from MES version 11.2.3, the system
    automatically enforces this configuration.
```

10.3 Access Control Lists (ACLs)

The MES system user must have the following permissions to function properly:

- **SELECT** - Read data from tables
- **INSERT** - Write data into tables
- **ALTER** - Modify table schema
- **DROP** - Delete databases (during setup) or tables
- **SHOW** - View metadata
- **CREATE** - Create databases (during setup) or tables

"Database Setup and Migration" During initial setup and version upgrades, the MES installation process requires elevated permissions to create or modify database schemas. Ensure these permissions are granted before installation or upgrade procedures.

11 ClickHouse (Observability) Requirements

ClickHouse Observability serves as the telemetry storage backbone for the Critical Manufacturing MES Observability platform. This specialized ClickHouse instance stores and provides high-performance access to all operational metrics, logs, and traces generated by MES components.

The Observability platform uses ClickHouse's columnar storage architecture to efficiently process large volumes of telemetry data while maintaining query responsiveness. This enables interactive public dashboards to visualize real-time performance metrics across the MES ecosystem.

11.1 Minimum system requirements

Table 9: Minimum system requirements for ClickHouse (Observability)

Item	Requirement
Version	24.9.2.42
Memory	16GB
CPU	4vCPUs
Disk space	100GB (per environment, for 7 days TTL)

11.2 Recommended system requirements

Table 10: Minimum system requirements for ClickHouse (Observability)

Item	Requirement
Version	24.9.2.42
Memory	32GB
CPU	8vCPUs
Disk space	100GB (per environment, for 7 days TTL)

12 Kafka Requirements

Kafka is an integral part of the MES event management pipeline, serving as the central messaging backbone for data replication, analytics, and IoT integration. Namely for:

- **Replication Events**
  - The HouseKeeper component identifies SQL Server Online Database changes and publishes replication events to Kafka.
  - HouseKeeper instances consume these events and write data to ClickHouse ODS (with horizontal scaling support).
  - Events are retained in Kafka for reprocessing when new CDM events are created.
- **CDM Events**

  - Canonical Data Model (CDM) Events are a core component of the Critical Manufacturing MES Data Strategy and Data Platform.
  - HouseKeeper produces CDM events based on replication events and publishes them to Kafka.
  - External systems may also emit CDM events to MES.
  - CDM events serve as the data foundation that powers reports and dashboards built with Grafana, Stimulsoft Reports, and CubeJS.
  - CDM events trigger Data Platform workflow execution when configured.

- **IoT Events**

  - MES can receive events from equipment and external sources as part of the Data Platform.
  - Events undergo authentication, authorization, and schema validation.
  - Valid IoT events are published and also stored in Kafka.

12.1 System Requirements Considerations

In regards to Kafka, the system requirements might change depending on several key factors such as:

- message volume (count and size)
- throughput
- number of partitions
- replication factor
- data retention policies

Proper benchmarking and monitoring of Kafka is essential for fine-tuning resource allocation based on your specific workload.

12.2 Minimum system requirement

Table 11: Minimum system requirements for Kafka

Item	Requirement
Version	>=3.8.1 to <4.0.0
Brokers	3
Memory	8GB per broker
CPU	2vCPUs per broker
Disk space	1TB per broker

12.3 Recommended system requirements

Table 12: Recommended system requirements for Kafka

Item	Requirement
Version	>=3.8.1 to <4.0.0
Brokers	3
Memory	16GB per broker
CPU	4vCPUs per broker
Disk space	3TB per broker

12.4 Access Control Lists (ACLs)

12.4.1 Topic Naming Convention

To simplify ACL setup, Kafka topics created by MES follow a standardized naming convention using the system name as a prefix.

Example: If the system name is `MESIntegrationEnvironment`, all Kafka topics will use the lowercase prefix:

- `mesintegrationenvironment.*`
- `\_mesintegrationenvironment.*`

12.4.2 Required Permissions

The MES system user must have the following permissions:

- **Topic Permissions**
  - Alter
  - AlterConfigs
  - Create
  - Delete
  - Describe
  - DescribeConfigs
  - Read
  - Write
- **Consumer Group Permissions**
  - Read
  - Delete
  - Describe
- **Cluster Permissions**
  - Create
  - Describe
  - DescribeConfigs

13 RabbitMQ Requirements

RabbitMQ serves as the message broker for Critical Manufacturing MES workflows execution, since it allows acknowledgement of messages per unit, instead of offset (like Kafka does).

When events require workflow execution, RabbitMQ queues individual messages for each workflow instance. This ensures reliable, controlled processing across multiple MES contexts:

- **Factory Automation** - Equipment and production workflows
- **Low-Code Enterprise Integration** - Business process automation
- **Data Platform** - Data processing and analytics workflows

The message-per-workflow approach allows fine-grained control over execution flow and reliable error handling at the individual workflow level.

13.1 Minimum system requirements

Table 13: Minimum system requirements for RabbitMQ

Item	Requirement
Version	>=3.13.1 to <4.0.0
Brokers	1
Memory	1GB per broker
CPU	2vCPUs per broker

13.2 Recommended system requirements

Table 14: Recommended system requirements for RabbitMQ

Item	Requirement
Version	>=3.13.1 to <4.0.0
Brokers	2
Memory	2GB per broker
CPU	4vCPUs per broker

14 AWS S3 (or compatible) Requirements

S3 storage is currently used to store Kafka Messages payloads when they exceed Kafka maximum payload sizes. But, its planned to become the primary storage solution for all binary/file storage in the MES ecosystem.

While designed for AWS S3, this implementation supports any S3-compatible storage provider (MinIO, Ceph, Google Cloud Storage, or other) that implements the required operations listed in section Used S3 APIs.

14.1 Minimum system requirements

Table 15: Minimum system requirements for AWS S3 (or compatible)

Item	Requirement
Version	–
Memory	4GB
CPU	2vCPUs

14.2 Recommended system requirements

Table 16: Recommended system requirements for AWS S3 (or compatible)

Item	Requirement
Version	–
Memory	16GB
CPU	4vCPUs

14.3 Required S3 APIs

The following S3 APIs are used by Critical Manufacturing MES components for storage operations:

Table 17: S3 APIs used by CM MES

API Operation	Purpose	Required Permission
DoesS3BucketExist	Validates bucket availability before performing operations	`s3:HeadBucket`
GetObject	Downloads and accesses stored data from S3	`s3:GetObject`
PutBucket	Creates new S3 buckets for storage initialization	`s3:CreateBucket`
Upload	Stores data and files in S3 storage	`s3:PutObject`

"API Compatibility" Ensure your S3-compatible storage provider supports all listed operations for full CM MES functionality. Furthermore, ensure that the IAM role or service account used by CM MES has all the listed (or equivalent) permissions assigned.

15 Application Layer

This section describes the contents of the application layer that will hold the business logic as well as other additional optional components for running Critical Manufacturing MES:

- Container Stack
- Optional Components

16 Application Layer - Container Stack

The Critical Manufacturing MES application tier is deployed in a containerized environment, providing solid gains in terms of configuration and usability by taking advantage of a mature containerization architecture such as the one supplied by the Kubernetes or Docker engines.

The application tier takes advantage of a container orchestrator in order to ensure almost boundless horizontal scalability. Additional worker nodes can be added to the cluster and additional instances of specific components can be launched to accommodate extra load on a particular installation.

16.1 Software Requirements

Containerized deployments rely on a container orchestrators. Critical Manufacturing MES supports the following container Orchestrators:

- Vanilla Kubernetes v1.28 or later (On-Premises)
- Red Hat OpenShift 4.15 or later (Cloud/On-Premises)
- Azure Kubernetes Services (Cloud)
- Amazon Elastic Kubernetes Service (Cloud)

When using these container orchestrators, keep in mind the following restrictions:

- **Operating System**
  - Any Linux distribution compatible with the selected container orchestrator.
  - Recommended: Ubuntu Server 20.04 LTS or Red Hat Enterprise Linux 9.
  - **Only x64 architecture is supported.** (Container images for x86 or ARM architectures are not available).
- **Container Engine**
  - CRI-O (Kubernetes only)
- **Additional Required Software**
  - Powershell 7.1 (On-Premises installations only)

16.2 Workload characterization

Due to the flexibility offered by container orchestrators, Critical Manufacturing does not recommend specific hardware configurations, but provides an adequate description of the expected workload generated by the MES system. With this information, system administrators can provision and adequately size a cluster to run the MES application tier, considering high-availability and scalability requirements.

This section offers a description of the MES workload in different example configurations.

Similarly to what happens in the more traditional approach, the exact hardware requirements depend ultimately on the specific customer environment. Please consult with Critical Manufacturing for more specific recommendations adapted to specific deployments.

16.2.1 Hardware requirements for computational resources

The following table describes the approximate requirements for some sample configurations, depending on their purpose. All examples assume no workloads related with equipment integration.

- **Development**: Development sandbox with no significant system load.
- **Training / Staging**: System with a similar configuration to a productive deployment, with higher resource demands.
- **Production (MES only)**: Productive MES deployment with low to medium volume, without significant usage of Data Platform or Machine Learning capabilities. This configuration includes adequate overhead to support a new MES deployment during an upgrade process.
- **Production (MES with Data Platform and Machine Learning)**: Productive MES deployment with medium to high volume, considering usage of Data Platform and Machine Learning features. This configuration includes adequate overhead to support a new MES deployment during an upgrade process.

Table 18: Hardware requirements for computational resources

Workload	vCPU	Clock speed	Memory
Development	10	2+ GHz	18 GB
Training / Staging	20	2+ GHz	32 GB
Production (MES Only)	30	2+ GHz	64 GB
Production (MES + Data Platform + Machine Learning)	50	2+ GHz	128 GB

The workloads defined above may be combined in the same cluster by simply adding the characteristics of the intended environments to host on the same cluster.

For example, if you want to host two Development Systems (one Staging and one MES-only production system) in the same cluster, the resource requirements will add up, as shown below:

Table 19: Hardware requirements example for cluster

Workload	vCPU	Memory
Development 1	10	18 GB
Development 2	10	18 GB
Staging	20	32 GB
Production (MES only)	30	64 GB
Total	70	132 GB

If several environments are deployed to the same cluster, Critical Manufacturing recommends resource quotas to be defined for each environment in order to prevent resource starvation on other environments or applications hosted on the same cluster. Usually, this is only possible on Kubernetes clusters.

16.2.2 Persistent Storage

The application layer requires access to persistent storage volumes to hold object attachments, documents, installation packages, and other application files. The storage requirements depend heavily on the expected usage of the system.

Persistent volumes can be provisioned in different ways, depending on the deployment target platform. At the time of release of this version, Critical Manufacturing MES supports the following volumes types:

Table 20: Support for persistent volume types

Volume type	Usage
Local	Refers to local path on the node file system.
SMB/CIFS	Refers to a shared folder accessible through the SMB protocol.
NFS	Refers to a shared folder accessible through the NFS protocol.
Azure File	Refers to an Azure File Share available in an Azure Storage Account (for AKS and OpenShift deployments only).
Storage Class	Refers to a Kubernetes API for dynamic PV provisioning based on defined storage profiles (provisioners).
Persistent Volume	Refers to a cluster-level, provisioned storage resource with an independent lifecycle, accessed via PVCs; can be static or dynamically provisioned.

Support for additional volume types may be added in the future.

The table below lists all the required volumes along with the component that requires it. Please note that, especially for **Kubernetes** deployments, there are different requirements for access modes. Some volumes require Read-Write-Many (RWX) access mode.

Note that due to the specific requirements of some technologies, high-performance block storage is required (identified in the table above). The volumes can be statically provisioned using existing Persistent Volumes (previously created by the customer) or dynamically provisioned using Storage Classes in Kubernetes deployments. For high-performance block storage, we recommend a storage solution with at least 10000 IOPS (Input/Output Operations Per Second)

Table 21: Volumes required for persistent storage.

Volume	Component	Access Mode	Storage Type	Minimum Size
connect-iot-repo	connectiot-manager, envmanager, host	RWX	Any	2 GB
grafana-share	grafana	RWX	Any	1 GB
⚠ installation-data	aggregation-engine, cube, data-manager, discoveryservices, envmanager, epf-alarm-mng-at, epf-alarm-mng-erh, epf-alarm-mng-mes-eh, grafana, help, host, housekeeper, messagebus, rasa-actions, securityportal, traefik-fwdauth, ui	RWX	Any	5 GB

Volume	Component	Access Mode	Storage Type	Minimum Size
mes-host-documents	host	RWX	Any	10 GB
ml-agent-folder	mlplatform-agent	RWX	Any	1 GB
ml-agent-folder	mlplatform-training	RWX	Any	1 GB
ml-export-folder	data-manager	RWX	Any	1 GB
ml-export-folder	mlplatform-training	RWX	Any	1 GB
ml-training-folder	mlplatform-training	RWX	Any	1 GB
redis-data	redis	RWO	Block Storage :information_source:	1 GB
cube	cube	RWO	Any	5 GB

⚠ The `grafana-share` volume needs to POSIX compliant. It is recommended to avoid using Azure Files as they can cause issues related with this requirement.

⚠ The `installation-data` volume is used by the system to store installation artifacts consumed by other components. In this volume, a directory named `<environmentname>/backups` will be created in order to extract the database backups for initial database deployment. This directory must be accessible by the SQL Server engine and the path must be provided during system installation, in order to ensure the initial database backup can be restored. For this reason, dynamic provisioning of this volume is not recommended since it may not generate a deterministic folder path that can be provided to an external SQL Server.

Optional Volume configuration

Table 22: Volumes of optional components or optional volumes.

Volume	Component	Access Mode	Storage Type	Minimum Size
dagster	dagster	RWO	Any	1 GB
kafka-data	kafka	RWO	Block Storage :information_source:	100 GB (per broker)
clickhouse-data	clickHouse	RWO	Block Storage :information_source:	100 GB
clickhouse-log	clickHouse	RWO	Block Storage :information_source:	20 GB
rabbit-data	rabbit	RWO	Block Storage :information_source:	1 GB
rabbit-log	rabbit	RWO	Block Storage :information_source:	1 GB
storage-data	storage	RWO	Block Storage :information_source:	25 GB
⚠ mes_logs_share	host, discoveryservices, envmanager, help, messagebus, UI	RWX	Any	20 GB

ⓘ The amount of storage required by the Critical Manufacturing components greatly depend on how the system is modelled and used. The size indications present in this document are a recommendation for low/medium volume sites, based on our experience. During the configuration of the system these values should be adjusted according to the defined parameterization.

⚠ While the `mes\_logs\_share` volume does not require Block Storage, it is crucial to consider that the overall system performance will be significantly impacted by the write speed to this volume. For example, a high-latency network share must be avoided in production environments to maintain optimal performance. When installing MES in containerized environments on a Linux-based environment in Ext4 (or a Unix file system family), a `lost+found` folder will be created (typically through fsck) for the volumes. Kafka containers are deployed together with the application and will not start due to inconsistent naming format. This folder must be either removed before starting the relevant Kafka containers or a different file system must be used that doesn't create this folder.

16.2.3 Scalable Components

The table below lists all the components along with the images that can be scalable.

Table 23: Components can be scalable.

Component	Image
core-host	criticalmanufacturing/core-host
core-ui	criticalmanufacturing/core-ui
data-manager	dataplatform/datamanager
edgesquidproxy	criticalmanufacturing/edgesquidproxy
epf-alarm-mng-at	dataplatform/epf-alarm-management-action-trigger
epf-alarm-mng-erh	dataplatform/epf-alarm-management-event-rule-handler
epf-alarm-mng-mes-eh	dataplatform/epf-alarm-management-mes-event-handler
grafana	criticalmanufacturing/grafana
help	criticalmanufacturing/help
host	criticalmanufacturing/host
housekeeper	dataplatform/housekeeper
messagebus	criticalmanufacturing/messagebus
mlplatform-agent	dataplatform/mlplatformagent
mlplatform-training	dataplatform/mlplatformagent
rasa	criticalmanufacturing/rasa
rasa-actions	criticalmanufacturing/rasa-actions
reference	criticalmanufacturing/reference
securityportal	criticalmanufacturing/securityportal
traefik	traefik
traefik-fwdauth	criticalmanufacturing/traefik-fwdauth
ui	criticalmanufacturing/ui

17 Application Layer - Optional Components

This section outlines the software and hardware requirements for application layer components of Critical Manufacturing MES that require traditional installation methods, distinct from the container-based main stack.

17.1 Software Requirements

The target application server must meet the following software specifications:

- **Operating System (OS):**
  - Windows Server 2016 64-bit
  - Windows Server 2019
  - Windows Server 2022
- **Microsoft .NET Runtimes:**
  - Version 8.0
- **Microsoft PowerShell:**
  - Version 5.1 or later (available in the `Dependencies` folder of the MES installation ISO or downloadable here)
- **Visual C++ Redistributable Package:**
  - Visual Studio 2013 (required solely by the ECAD Service)

17.2 Hardware Requirements

The precise hardware demands will vary based on the specific customer environment. The table below highlights key factors influencing response time, scalability, and high availability. It also suggests possible improvements, while noting that enhancing each factor may incur additional costs.

Table 24: Driving factors for hardware requirements

KPI	Primary Driving Factors	Improvement Options
Response Time & Scalability	• Load (number of users and concurrent transactions) • Model and Logic Complexity • Data Volume • Hardware	• Application Optimization • Model Fine-tuning • Database Optimization • Reduction of Data Retention Time • Enhanced Hardware (CPU, Memory, I/O and Network)

KPI	Primary Driving Factors	Improvement Options
High-Availability	• Software and Hardware Failures & Maintenance • Human Errors • Redundant Hardware Components (memory, processing, storage, communications and power supply) • Operations Management (support processes, monitoring procedures, backup policies, administration skills and contingency plans)	Implementation of Redundant or More Reliable Hardware (memory, processing, storage, communications, and power supply) • Strengthening Operations Management (support processes, monitoring procedures, backup policies, administration skills, and contingency plans)

The subsequent parts of this section reference hardware configurations tailored for different environment setups.

Table 25: Environment configuration options

Configuration	Intended Use	Response time	Scalability	High availability
Minimum	Demonstration / Development	Medium/Slow acceptable	Not required	Not required
Training / Staging	System tests, validations and training	Medium	Not required	Not required
Production A	Low to medium volume production	Fast	Required	Very High
Production B	High volume production	Fast	Required	Very High

The exact hardware configuration for any Critical Manufacturing MES deployment project must be carefully sized according to the specific project requirements.

The following table specifies the hardware requirements for each environment configuration outlined above:

Table 26: Hardware requirements for different environment configurations

Configuration	Number of Servers	Processors	Memory	Storage
Minimum	1 *	1 x Quad Core, 2 GHz	8 GB	200+ GB
Training / Staging	1	2 x Quad Core, 2 GHz+	16 GB	100+ GB
Production A	2	2 x Quad Core, 2 GHz+	16 GB+	150+ GB
Production B	3	2 x Quad Core, 2 GHz+	32 GB+	150+ GB

The Minimum configuration assumes that the database server may host other application components in addition to the database.

17.3 Network Considerations

Firewall systems are crucial for preventing unauthorized access to computer resources. If a firewall is active but not properly configured, connection attempts to Critical Manufacturing MES may be blocked.

Ensure that the ports specified during the Network Configuration step of the Critical Manufacturing MES installation are opened in the firewall, allowing the application server to receive incoming requests.

To enable database communication between application and database servers, the necessary SQL Server ports must also be configured in the firewall.

For more information, see:

- Configure the Windows Firewall to Allow SQL Server Access
- Configure a Windows Firewall for Database Engine Access

18 Connect IoT Requirements

The Connect IoT **Automation Manager** component has a low hardware footprint and is designed to run in a variety of platforms.

18.1 Software Requirements

The supported operating systems (must be supported by Node.js), are listed in the table below:

Table 27: Supported operating systems for Connect IoT

System	Architecture	Version
GNU/Linux	x64	kernel >= 3.10, glibc >= 2.17
GNU/Linux	arm64	kernel >= 4.5, glibc >= 2.17
Windows	x64, x86 (WoW64)	>= Windows 7/2008 R2/2012 R2
macOS	x64	>= 10.11

Requirements based but not limited to the *Tier 1* values of the Node.js platform list compilation/execution support, available in https://github.com/nodejs/node/blob/v16.x/BUILDING.md#platform-list.

Independent from the Operating System, the following software components must also be available:

Table 28: Software components required for Connect IoT

Requirement	Description
NodeJS	NodeJS must be installed and available in each computer that runs the Connect IoT Automation Manager (the Connect IoT runtime). Critical Manufacturing recommends version **20.x LTS** (available from https://nodejs.org/dist/latest-v20.x/) for improved functionality.
Local Package Repository	- There must be one local package repository available per site. We support **NPM** based repositories or our custom **directory** based directory for installation/server free solution - The repository must be accessible by both Critical Manufacturing MES and Connect IoT runtime computers.

Furthermore, there some driver specific requirements as listed in the following table that are applicable to the computer that will host the Connect IoT runtime engine that uses that driver:

Table 29: Driver requirements for Connect IoT

Driver	Requirement	Can be containerized
Bluetooth (BLE)	- Must fulfill the noble pre-requisites - refer to the link https://github.com/sandeepmistry/noble - Must have a compatible Bluetooth adapter - https://github.com/noble/node-bluetooth-hci-socket#Prerequisites - In Microsoft Windows, it is necessary to change the Bluetooth driver to use WinUSB instead of the Microsoft Bluetooth Stack. To accomplish this, use the *zadig* tool supplied in the Setup ISO.	Yes (requires hardware devices to be forwarded to container)
File (CSV + Raw)	- The directory(ies) to be used by the driver must be fully accessible, mapped and authenticated within the OS.	Yes
IPC-CFX	- .Net Core 8.x SDK - refer to https://github.com/dotnet/core/blob/main/release-notes/8.0/supported-os.md - Install Visual C ++ Redistributable 2015-2022 in every machine - refer to https://learn.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist?view=msvc-170	Yes
Keyboard Wedge	- Can only work in Linux OS (it is restricted by Windows because it acts as a keyboard logger).	Yes (requires hardware devices to be forwarded to container)
OIB	- Must have Microsoft .Net Framework 4.6 installed.	No

Driver	Requirement	Can be containerized
OPC DA	- Can only run on a Windows OS (OPC DA protocol requirement) - Must have .Net Framework 4.8 installed. - Must have *AdvosolOpc-CoreComponents* (supplied in the Setup ISO) installed.	No
OPC UA	- Must have OpenSSL installed (for on-demand certificate generation). For more information, see https://www.openssl.org/docs/.	Yes
SECS/GEM	- Minimal support .Net Core 2.x SDK - refer to https://github.com/dotnet/core/blob/master/release-notes/2.0/2.0-supported-os.md - Recommended support .Net Core 8.x SDK - refer to https://github.com/dotnet/core/blob/master/release-notes/8.0/supported-os.md - Install Visual C ++ Redistributable 2015-2022 in every machine - refer to https://learn.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist?view=msvc-170	Yes

18.2 Hardware requirements

Connect IoT **Automation Managers** are designed to be executed on edge devices, with limited hardware capabilities. However, they can also be deployed in containerized environments together with the MES Application Layer. The actual requirements in terms of processing power and memory will depend on the number of integrations and the workflow implemented for each integration.

Contact Critical Manufacturing for additional information to help you estimate the workload associated to your integrations.

19 Client Devices

The Critical Manufacturing MES UI is an HTML5 Single Page Application accessible with a compatible web browser. It does not require any installation on client devices.

This section outlines the supported browsers and specific requirements for client devices.

19.1 Supported Browsers

Table 30: Supported browsers

Operating System	Supported browsers
Android 10 or later	• Google Chrome
Linux ARM	• Chromium
Microsoft Windows 10 / 11 (32 or 64-bit editions)	• Google Chrome • Microsoft Edge
macOS	• Google Chrome

19.1.1 Supported Chrome Drivers

To ensure browser compatibility and UI stability, each MES version undergoes automated testing using a specific Web Driver (ChromeDriver).

For more information, see the section Supported Browsers of the Release Notes {{ extra.current_version }} in the Critical Manufacturing Information Center.

⚠ Important information to prevent possible compatibility issues:

```
* It's highly recommended that Critical Manufacturing customers use the enterprise version of Google
    Chrome, which can be obtained from: <https://cloud.google.com/chrome-enterprise/browser/>. The
    Chrome Browser Deployment Guide, offering information on organizational deployment, is available
    at: <https://support.google.com/chrome/a/answer/3115278>.
* Auto-updates for the browser **must be disabled** to ensure optimal compatibility with the tested MES
    version. This is often managed through group policies or browser settings
    (<https://support.google.com/chrome/a/answer/187202>)). If newer browser versions are used,
    support will be provided on a best-effort basis.
* Refer to
    <https://chromium.googlesource.com/chromium/src/+/master/docs/chromium_browser_vs_google_chrome.md>
    for the differences between Google Chrome and Chromium on Linux.
```

19.2 Full GUI Functionality Configuration

In order to enable all the functionalities of the GUI, it is necessary to configure the browser used to access the HTML GUI site as described in the table below:

Table 31: Full GUI Functionality Configuration

Setting	Description
Allow Pop-ups	Pop-ups must be allowed for the GUI site.
Whitelist ad-blockers	The GUI site must be whitelisted in any ad-blocker software

19.3 Minimum Hardware Requirements

⚠ "Try before buying new devices" UI performance should be validated before moving forward with the purchase of additional devices. Different manufacturers have different flavors of the operating system and additional system applications running that may significantly affect a device's performance. Depending on your specific requirements or workload, more powerful devices may be required (for example, if FabLive 3D is used with complex scenes).

The following sections present the minimum hardware requirements for clients accessing the Critical Manufacturing MES GUI.

19.3.1 Desktop Devices

Desktop Clients must meet the following minimum requirements:

- **Processor**: 1 x Quad Core, 2.4 GHz
- **Memory**: 6 GB
- **Free Disk Space**: 10 GB
- **Minimum resolution**: 1366 x 768

19.3.2 Mobile devices

Mobile devices must meet the following minimum requirements:

- **Processor**: 1xOcta Core, 2.4 GHz
- **Memory**: 6 GB

We recommend the following resolution formats for optimal viewing and operation of the Critical Manufacturing MES on mobile devices:

Table 32: Mobile device resolutions for client devices

Horizontal Resolution	Vertical Resolution	DPI	Scaling	Android Density Qualifier
360	640	160	1	mdpi
540	960	240	1.5	hdpi
720	1280	320	2	xhdpi
1080	1920	480	3	xxhdpi
1440	2560	640	4	xxxhdpi

20 Other Requirements

20.1 Authentication

You need to decide how your users shall authenticate in the system. Critical Manufacturing MES supports:

- **Active Directory**
- **Single Sign-On** (Open ID Connect)
- **Local Users** (account credentials stored on MES database).

Each enabled strategy will require distinct requirements. Check Critical DevOps Center and Operations guides for additional details about their requirements and setup.

20.2 SAP Integration

If you intend to generate SAP interface code or proxies based on the Critical Manufacturing MES ERP module, then you must acquire an ERPConnect license from Theobald Software.

20.3 HTTPS

To enable HTTPS, you will need to provide SSL certificate information during DevOps Center new environment configuration.

Although is not mandatory, it is highly recommended to use HTTPS. Some features that require access to device peripherals (Augmented Reality, Clipboard) may not work when running on an insecure connection.

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 and 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 written 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.