ZEROSWARM

ZERO-enabling Smart networked control framework for Agile cyber physical production systems of systems

D4.4 Federated data infra & toolkit for data-driven model.v1

Project funded by Horizon Europe, Grant Agreement #101057083

Topic	HORIZON-CL4-2021-TWIN-TRANSITION-01-08
Project Title	ZERO-enabling Smart networked control framework for Agile cyber physical production systems of systems
Project Number	101057083
Project Acronym	Zero-SWARM
Contractual Delivery Date	M12
Actual Delivery Date	M12
Contributing WP	WP4
Project Start Date	01/06/2022
Project Duration	30 Months
Dissemination Level	Public
Editor	NXW
Contributors	AIM, CERTH, AALTO, S21SEC, FhG, ABB

Author List

Leading Author (Editor)			
Surname	**Initials**	**Beneficiary Name**	**Contact email**
Andolfi	M A	NW	m.andolfi@nextworks.it
Co-authors (in alphabetic order)			
Surname	**Initials**	**Beneficiary Name**	**Contact email**
Barja	L B	AIM	lara.barja@aimen.es
Kolesnikov	MK	AALTO	mikhail.kolesnikov@aalto.fi
Lazaridis	GL	CERTH	glazaridis@iti.gr
Lopez	OL	S21Sec	olopez@s21sec.com
Mendes	BM	UBI	bmendes@ubiwhere.com
Mpatziakas	S M	CERTH	ampatziakas@iti.gr
Safari	P S	FhG	pooyan.safari@hhi.fraunhofer.de
Stanica	MPS	ABB	marius-petru.stanica@de.abb.com

Reviewers List

List of reviewers (in alphabetic order)			
Surname	Initials	Beneficiary Name	Contact email
Drosou	AD	CERTH	drosou@iti.gr
Khodashenas	P S K	HWE	pouria.khodashenas@huawei.com

Document History

Document History			
Version	**Date**	**Author**	**Remarks**
V1	15/03/2023	Matteo Andolfi (NXW)	ToC
V1.1	04/04/2023	Pooyan Safari (FhG)	Contribution to Section 4
V1.2	14/04/2023	Lara Barja Besada (AIM) Beatriz Mendes (UBI)	Contribution to Section 2
V1.3	02/05/2023	Oscar Lopez (S21Sec)	Contribution to Section 2
V1.4	16/05/2023	Stelios Mpatziakas (CERTH) George Lazaridis (CERTH) Mikhail Kolesnikov (AALTO)	Contribution to Section 4 Contribution to Section 3
V1.5	20/05/2023	Matteo Andolfi (NXW)	Contribution to Section 4 and minor editorial fixes
V1.6	26/05/2023	Matteo Andolfi (NXW) Pooyan Safari (FhG)	Fix review comments
V1.7	26/05/2023	Marius Stanica (ABB)	Added chapter 2.1.4 and 3.1.1. about OPC UA info modelling
V1.8	29/05/2023	Pouria Khodashenas (HWE)	Review of the whole document
V2.05	08/06/2023	Marius Stanica (ABB)	Updated 2.1.4. and 2.1.6 chapters and added some comments in the text of chapter 1 and some other comments of missing captions and list of figures.
V2.1	09/06/2023	Marius Stanica (ABB)	Updated OPC UA related references
V2.2	12/06/2023	Pouria Khodashenas (HWE)	Review of the whole document
V3	13/06/2023	Anastasios Drosou (CERTH)	Final submission

DISCLAIMER OF WARRANTIES

This document has been prepared by Zero-SWARM project partners as an account of work carried out within the framework of the contract no 101057083.
Neither Project Coordinator, nor any signatory party of Zero-SWARM Project Consortium Agreement, nor any person acting on behalf of any of them:

- makes any warranty or representation whatsoever, express or implied,
    - with respect to the use of any information, apparatus, method, process, or similar item disclosed in this document, including merchantability and fitness for a particular purpose, or
    - that such use does not infringe on or interfere with privately owned rights, including any party's intellectual property, or
- that this document is suitable to any particular user's circumstance; or
- assumes responsibility for any damages or other liability whatsoever (including any consequential damages, even if Project Coordinator or any representative of a signatory party of the Zero-SWARM Project Consortium Agreement, has been advised of the possibility of such damages) resulting from your selection or use of this document or any information, apparatus, method, process, or similar item disclosed in this document.

Zero-SWARM has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101057083. The content of this deliverable does not reflect the official opinion of the European Union. Responsibility for the information and views expressed in the deliverable lies entirely with the author(s).

Executive Summary

Modern businesses rely heavily on data, which plays a critical role in providing valuable insights and facilitating real-time control over essential processes and operations. The sheer volume of data available is vast, with remotely located IoT devices and sensors collecting vast amounts of information from harsh environments worldwide. However, this abundance of data is changing how computing is handled in business. Traditional centralized data centres and internet infrastructure are insufficient to support the endless flow of real-time information. Network disruptions, bandwidth restrictions, and latency issues can hinder data collection efforts. To overcome these difficulties, businesses are embracing edge computing architecture and federated data. Federated data refers to the concept of distributed datasets that are connected across multiple sources or systems. This data can be from various types of sources, such as databases, data warehouses, cloud-based storage, or even third-party data providers. The idea behind federated data is to create a centralized view of all the data available in an organization, regardless of where it is stored. This can facilitate better data collaboration and analysis, enabling better decision-making and insights. Additionally, federated data solutions allow users to seamlessly access and query data across multiple systems and sources, without having to manually switch between them.

The federated data concept is interrelated with the Edge computing for working together to enhance performance and reliability of distributed systems, like the ones involved in the Zero-SWARM project. Edge computing and federated data work together in a distributed system by enabling data to be processed and analysed at the edge of the network, where it is generated, while still being able to share insights with other participants using federated data. This approach provides a more scalable, reliable, and secure way to manage data in a distributed system, while also providing greater flexibility and efficiency.

In this document we discuss about the state of the art of technologies that are involved in the project and that are important for inferring a data infrastructure that deals with different types of data and protocols, like OPC Unified Architecture (OPC-UA), Asset Administration Shell, several aspects of Edge computing, like the Federate learning approach that will be used to keep the data locally to the shop floor with lower privacy-risks. Another aspect is the Cybersecurity aspect that should be addressed to store and use the data gathered from the shop floor.

We also discuss about the technology scouting that was made by all partners involved in the project, regarding the Data infrastructure. In addition, the DevOps environment that we propose to use in Zero-SWARM and the introduction of a ML operations (MLOps) environment for machine learning and training purposes are introduced and explained in this document.

Table of Contents

List of Figures

List of Tables

List of Acronyms

Acronym	Description
5G-ACIA	5G Alliance for Connected Industries and Automation
AI	Artificial Intelligence
CPS	Cyber Physical System
IIC	Industrial Internet Consortium
IIRA	Industrial Internet Reference Architecture
ICT	Information and Communications Technology
ISO	International Organization for Standardization
IEC	International Electrotechnical Commission
IIoT	Industrial Internet of Things
IDSA	International Data Space Association
IoT	Internet of Things
IA	Industrial Automation
LNI4.0	Labs Network Industrie 4.0
MNO	Mobile network operator
NPN	Non-public (private) network
OT	Operational technology
PNI-NPNs	Integrated non-private networks
RAMI 4.0	Reference Architectural Model Industrie 4.0
SDO	Standardisation Organisation
SGAM	Smart Grid Architecture Model
SOA	Service Oriented Architecture
VDMA	Mechanical Engineering Industry Association
OPC-UA	OPC Unified Architecture

1 Introduction

Federated data infrastructure refers to a decentralized approach to data management, where data is distributed across multiple sources or organizations while maintaining control and privacy over the data. In a federated data infrastructure, data remains local to its original source, and computations or analyses are performed locally or in a decentralized manner, without requiring data to be transferred or centralized.

The main idea behind federated data infrastructure is to enable collaboration and analysis on sensitive or proprietary data without the need to share or disclose the actual data. It offers several benefits, such as enhanced privacy, security, and data governance, while also allowing organizations to leverage the collective intelligence of multiple data sources.

There are several technical challenges that need to be addressed in implementing federated data infrastructure:

- Data Heterogeneity: Data sources in a federated infrastructure may have different formats, schemas, or structures. Interoperability challenges arise when integrating and processing diverse data types.
- Data Security and Privacy: Federated data infrastructure must ensure the privacy and security of data across multiple sources. Sensitive information should be protected, and access control mechanisms must be in place to ensure that only authorized parties can access specific data.
- Data Governance: Coordinating and governing data across multiple sources can be complex. Establishing data ownership, defining data sharing agreements, and maintaining data quality and consistency are critical challenges.
- Distributed Computing: Federated data infrastructure often involves performing computations and analysis across distributed data sources. This requires efficient algorithms and frameworks to handle distributed computing, network latency, and data synchronization.
- Scalability and Performance: Federated data infrastructure should be capable of handling large volumes of data and supporting real-time or near-real-time analysis across multiple sources. Scalable architectures and optimization techniques are necessary to ensure efficient processing and analysis.
- Data Bias and Representation: In federated environments, data sources may have different biases, resulting in skewed or incomplete insights. Addressing these biases and ensuring representative and fair analysis across multiple sources is a significant challenge.
- Metadata Management: Federated data infrastructure requires effective metadata management to enable data discovery, understanding, and integration across multiple sources. Consistent metadata standards and tools are necessary to facilitate data exploration and analysis.

Addressing these technical challenges requires a combination of advanced algorithms, secure data exchange protocols, standardized metadata models, and collaborative frameworks to enable effective collaboration while preserving data privacy and security. Ongoing research and development efforts

are focused on overcoming these challenges to fully realize the potential of federated data infrastructure.

In Industry 4.0 (I4.0), federated data infrastructure plays a crucial role in enabling the efficient and secure exchange of data across various entities and systems. It supports the vision of interconnected industrial systems, automation, and data-driven decision-making. In Figure 1 a schema illustrating the federated data infrastructure in I4.0:

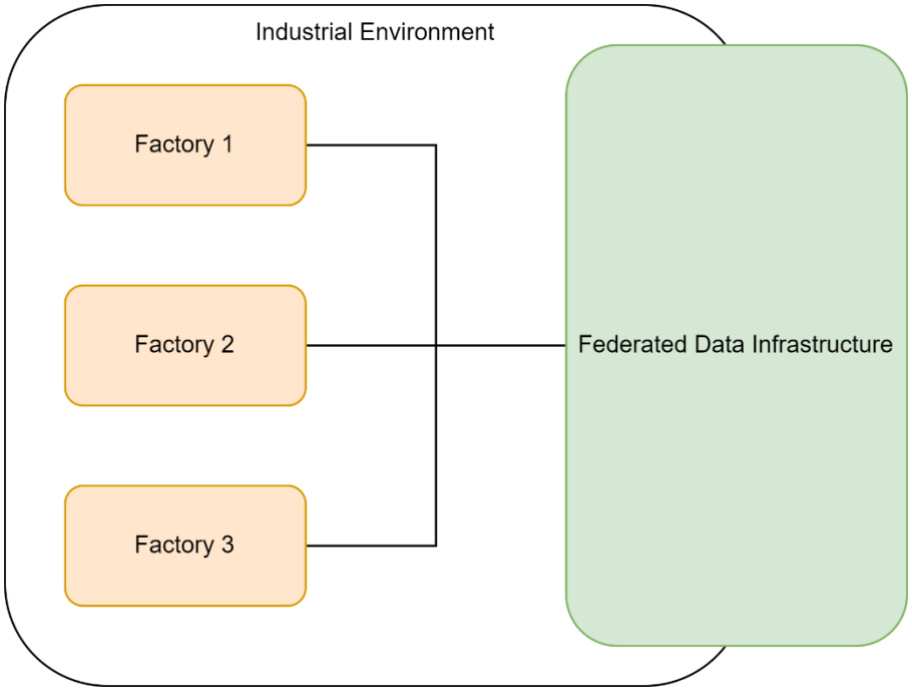


Figure 1: Federated Data Infrastructure in Industry 4.0

In this schema, the federated data infrastructure encompasses multiple factories or industrial environments (Factory 1, Factory 2, and Factory 3) within the same organization or across different organizations.

Each factory represents a production facility or system that generates large volumes of data from various sources, such as sensors, machines, production lines, and quality control systems. This data can include information about machine performance, production metrics, product quality, maintenance logs, and more.

The federated data infrastructure acts as a centralized or distributed framework that enables data sharing, integration, and analysis across these factories. It provides the following capabilities:

- Data Collection: The infrastructure collects data from diverse sources within each factory, including IoT sensors, control systems, and machine interfaces. Data is captured in real-time or near real-time.
- Data Integration: It integrates data from different factories, combining datasets from multiple sources to provide a unified view of the production processes and performance metrics. This allows for cross-factory analysis and decision-making.
- Data Security and Privacy: The federated infrastructure ensures the security and privacy of

sensitive industrial data. Access controls, encryption, and secure communication protocols are implemented to protect data during storage and transmission.

- Data Analytics and Insights: The infrastructure supports advanced analytics techniques such as machine learning, predictive modelling, and statistical analysis. It enables the discovery of patterns, anomalies, and optimization opportunities across factories, leading to improved efficiency, quality, and productivity.
- Real-time Monitoring and Control: The federated infrastructure enables real-time monitoring and control of industrial processes. It facilitates remote monitoring, alerting, and intervention to ensure smooth operations and proactive maintenance.
- Decision Support: The infrastructure provides data-driven insights and decision support tools to optimize production processes, resource allocation, maintenance scheduling, and supply chain management.

In this deliverable we discuss about various technologies that are developed and used in the Zero-SWARM architecture to address the challenges related to the federated data infrastructure in I4.0 context. In particular, Zero-SWARM introduces and promotes the concept of Active Information Continuum (AIC) as a booster in the digital and green transformation of manufacturing sector. The AIC in Industry 4.0 refers to the progressive stages or levels of comprehension and insight gained from data and information within the context of advanced manufacturing and industrial processes. It represents the evolution of knowledge and understanding as data is collected, analysed, and transformed into actionable insights. It typically consists of the following stages:

- Data Collection: At the initial stage, data is collected from various sources within the industrial environment, including sensors, machines, production lines, and other data-generating devices. This raw data serves as the foundation for further analysis.
- Data Integration: In this stage, data from different sources and systems are integrated and consolidated into a unified view. Integration allows for a comprehensive understanding of the overall manufacturing process, combining data from multiple areas such as production, quality control, inventory, and maintenance.
- Descriptive Analytics: Descriptive analytics involves organizing and summarizing data to gain insights into what has happened in the past. It includes statistical analysis, data visualization, and reporting, enabling the identification of patterns, trends, and anomalies.
- Diagnostic Analytics: Diagnostic analytics aims to understand why specific events or outcomes occurred. It involves deeper analysis and investigation into the factors contributing to certain patterns or incidents. Diagnostic analytics helps identify root causes, correlations, and relationships between different variables.
- Predictive Analytics: Predictive analytics uses historical data and advanced algorithms to forecast future events or outcomes. By analyzing patterns and trends, predictive analytics enables proactive decision-making, anticipates maintenance needs, optimizes production processes, and forecasts demand, among other applications.
- Prescriptive Analytics: Prescriptive analytics goes beyond prediction by providing recommendations and actionable insights to optimize decision-making. It suggests the best

course of action based on predictive models, simulations, and optimization algorithms. Prescriptive analytics helps organizations make informed choices to achieve desired outcomes and performance improvements.

Figure 2 presents schema of Zero-SWARM AIC enabled by tools and solutions for the Federated Data Infrastructure and the Machine Learning Operations (MLOps) environment developed in the project. In this schema, the continuum starts with Data Collection, where raw data is collected from various sources within the industrial environment. This raw data serves as the foundation for subsequent stages. Here the project looks into different solutions including OPC-UA. The main benefit of OPC UA in this context is the capability of OPC UA-based ecosystem to create a secure, harmonised and standardised mechanism for information modelling and data exchange. The OPC Foundation has a large number of industrial automation representatives and there are connected workgroups (e.g., VDMA, ISA-95, Fieldcomm, etc) which create companion specifications, following the rules of certification defined by the OPC Foundation. The OPC Foundation has thus became a sort of 'an UNO of the industrial automation application layer'.

Next, Data Integration consolidates and integrates data from different sources into a unified view, enabling a comprehensive understanding of the manufacturing processes. In order to achieve successful data integration, Zero-SWARM uses the capabilities that I4.0 implements. The base of the I4.0 is to enable the information exchange between users with generic technology-neutral standard, independent of manufacturers. This is made through the use of the Asset Administration Shell (AAS), which is the representation of the Digital

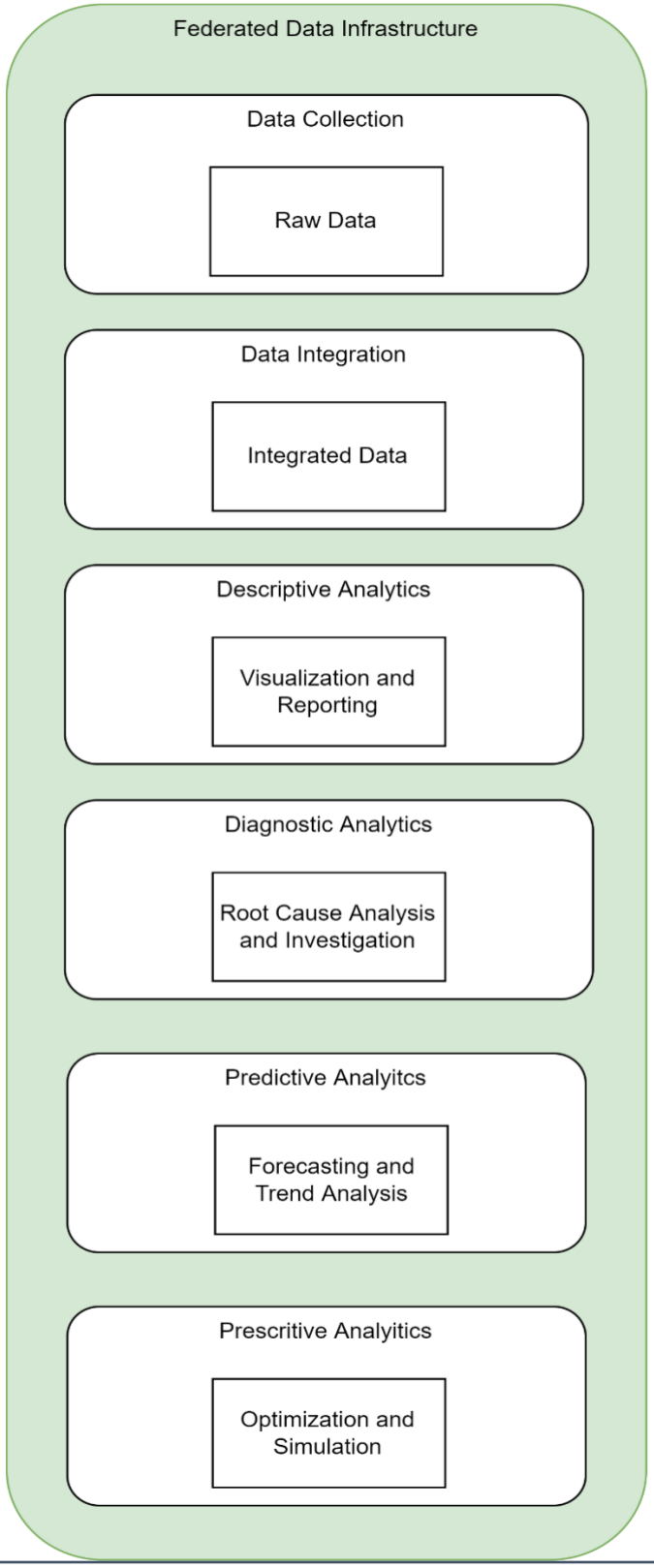


Figure 2: Zero-SWARM AIC enabled federated data infrastructure.

Twin in I4.0 and contains all the information that characterises an asset, simplifying the interaction of information between different manufacturers with industry-neutral standards and enabling a unified view of data in a decentralised world.

Descriptive Analytics involves organizing and summarizing the data, utilizing techniques like visualization, and reporting to gain insights into historical events and patterns. Diagnostic Analytics goes deeper into the data to understand the causes behind specific outcomes or incidents, identifying root causes and correlations. Predictive Analytics leverages historical data and algorithms to forecast future events or outcomes, enabling proactive decision-making and anticipating maintenance needs. Prescriptive Analytics provides actionable insights and recommendations based on predictive models, simulations, and optimization algorithms, guiding organizations to make informed choices for achieving desired outcomes. Here the project proposes a MLOps environment. The benefit of our solution is that one or multiple entities can collaboratively train a machine learning model without displacing the data or transferring it to a central location. This enables different data owners to participate in the training without sharing the raw data. It is also possible to safeguard the model from certain types of attack by adding multiple layers of privacy protection such as Secure Aggregation or Differential Privacy. On the other hand, avoiding huge volume of data transfer acts in the favour of communication efficiency.

The AIC represents a progression from raw data to actionable insights, enabling organizations to gain deeper understanding, make data-driven decisions, optimize processes, and improve overall performance within the industry 4.0 framework. This has been enabled over the foundation of edge computing resources and 5G connectivity. In the following we will present the Zero-SWARM solution to realize efficient and effective use of edge computing. Its aim is to facilitate the convergence between the distinct formats, data types, and protocols of industry 4.0 and the IT/OT realm. In Industry 4.0, the integration of IT and OT enables real-time access to data, predictive maintenance, remote monitoring and control, and overall optimization of industrial processes. This convergence is critical for achieving the full potential of the technologies such as the internet of things (IoT), machine learning, and artificial intelligence, encompassing diverse kinds of data. In other words, the AIC in Zero-SWARM serves as a joining step towards homogenizing the data between the two different worlds.

Cybersecurity threats associated in the previous mentioned environments like AIC and also IIoT, and Industry 4.0 are mentioned in in Chapter 2.3 which provides an updated overview and explains the concepts how different Cybersecurity standards cover and make a treatment about the issues related to Data Security and Privacy. The Chapter also provided some guidelines and frameworks to enhance the security in CPSoS in Zero-Swarm. Chapter 2.3.1, focus on cyber threats associated with Cyber-Physical Systems of Systems (CPSoS). Additionally, Chapter 2.3.4 discusses the cybersecurity threat landscape related to access control, with a focus on Operational Technology (OT) domain that extends to Information Technology (IT). The chapter concludes by emphasizing the increasing cybersecurity threat landscape in the industry due to the digitization of the supply chain, which exposes systems to new threats and potential equipment damages. The European Union Agency for Cybersecurity (ENISA) conducts regular assessments of the cybersecurity threat landscape in Europe.

To complement the methodology and guidelines for cybersecurity by design introduced in deliverable D2.3, in Chapter 2.3.12 further work in cybersecurity for zero-SWARM is introduced, which will cover specific aspects that will be developed in the project, including vulnerability assessments,

cybersecurity monitoring, and event detection and incident response. These measures aim to enhance cybersecurity maturity in a solution like zero-SWARM.

1.1 Purpose of the document

This deliverable provides the initial studies and the first draft solution for a data infrastructure and a Machine Learning Operations (MLOps) environment to be used within the project, which the consortium has discussed over the past months and will begin to develop in the near future.

1.2 Relationship with other deliverables

This deliverable is tightly coupled with the next deliverables of WP4 and with the deliverables of the other Work Package, especially the architectural deliverables of WP2 and the AI-focused deliverables of WP5. Some outcomes of WP4 activities will be tested and evaluated within the project trials under WP6. The results of the evolutions and related Key Performance Indicators (KPIs) of those items will be further detailed in the deliverables of WP6.

2 Data infrastructure, enabling tools, gap analysis and Zero-SWARM solutions

2.1 Data Heterogeneity and Data Collection

One of the major challenges in Industry 4.0 is data collection and analysis due to the heterogeneity of data sources. Heterogeneity refers to the diversity of data sources, formats, and types, which are used in different stages of manufacturing processes. This heterogeneity results in the need to bring together data from multiple sources, which can pose significant challenges in data collection, integration, and analysis. For instance, data could come from different manufacturing machines, sensors, and systems, and they could be in various formats such as text, images, and video. Moreover, collecting and handling such diverse and voluminous data require advanced data processing and analytical capabilities, which are still emerging in some industries. The complexity of data collection and analytics in I4.0 requires a robust information management infrastructure that can handle the massive stream of data generated from different sources and provide meaningful insights for decision-making.

In recent years, technological advancements such as OPC-UA, AAS, etc. have revolutionized the Industrial Internet of Things (IIoT) landscape, facilitating seamless communication between devices, improving efficiency, and enhancing safety. The adoption of these technologies has led to the emergence of a new era of I4.0, where machines can communicate and work together in real-time, enabling unprecedented levels of automation and productivity. This section provides comprehensive understanding of the latest trends and advancements of these technologies.

In the next subsections we described some technologies already addressed in other deliverables [1], the objective is to make the deliverables as mush self-contained as possible to be used in the context of Zero-SWARM.

Open Platform Communications (OPC) is a standard for the secure and reliable exchange of data, which facilitates interoperability in process control and manufacturing automation. It implements a common system interface for the different devices in industrial environments, which is independent of manufacturers and vendors.

Initially, the OPC standard (OPC Classic) was restricted to the Windows operating system and with the introduction of service-oriented architectures, OPC UA was developed to comply with the specifications of these architectures and to provide a feature-rich technology scalable and extensible open-platform.

OPC UA is a platform-independent standard with a common infrastructure model to facilitate information exchange that can be request and response messages or network in function of the exchange mechanism selected. It supports robust, secure communication that assures the identity of OPC UA Applications and resists attacks.

OPC UA provides interoperability between higher level functions, enabling vertical and horizontal communication between different assets in a security form, with access control, fault tolerance, encryption, and redundancy. Therefore, OPC provides robustness of published data with mechanisms to detect and recover from communication failures.

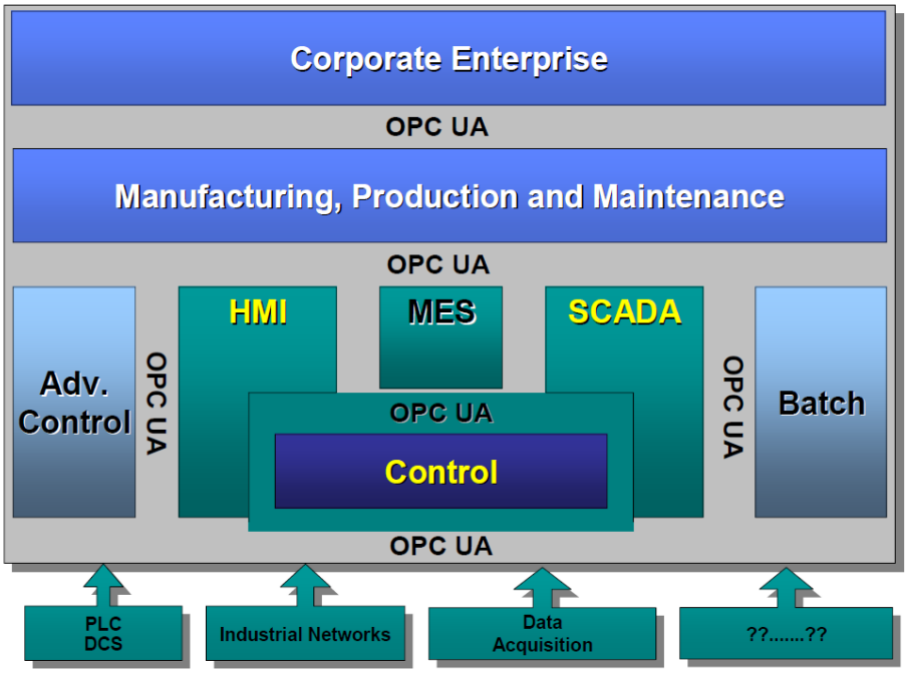


Figure 3: OPC UA target applications [2]

2.1.1 Information model

OPC UA provides a framework to be used to represent complex information (objects) in an AddressSpace, accessible through services. The AddressSpace is the set of objects (data type, object definition, object instantiation, etc.) and related information that the server makes available to clients, which is represented as a set of nodes connected by references. This structure permits any kind of relation between objects, different from a tree structure as in Classic OPC.

A node in OPC UA is composed of attributes, which describe the node, and are identified in unique and unambiguous form through its NodeID, which is formed by three elements: a namespace index associated to a URI, the identifier type (numeric, string, GUID or custom) and the identifier itself.

2.1.2 OPC UA services

The interface between clients and servers is defined as a set of services. These services give to the client the capabilities to send request to the servers, receive responses from them and subscribe to

server's notifications. All services are defined by the OPC-UA standard and messages for request and response are fixed data structures with a fixed binary encoding.

2.1.3 Exchange information models

OPC UA has to mechanisms to exchange the data, which can be used separated or combined:

- **Client-Server model:** OPC UA defines sets of services that servers may provide and the servers specify to clients their supported services. Servers provide access to current and historical data, as well as alarms and events to notify important changes to the clients. This peer-to-peer approach provides a secure and confirmed exchange of information, but with limitations regarding the number of connections.

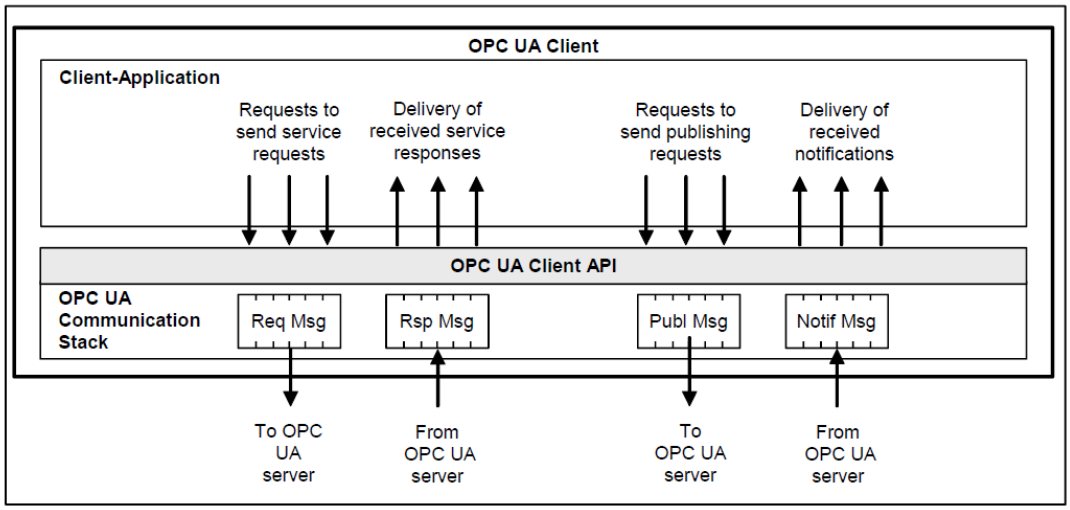


Figure 4 OPC UA client architecture (OPC Foundation, 2022)

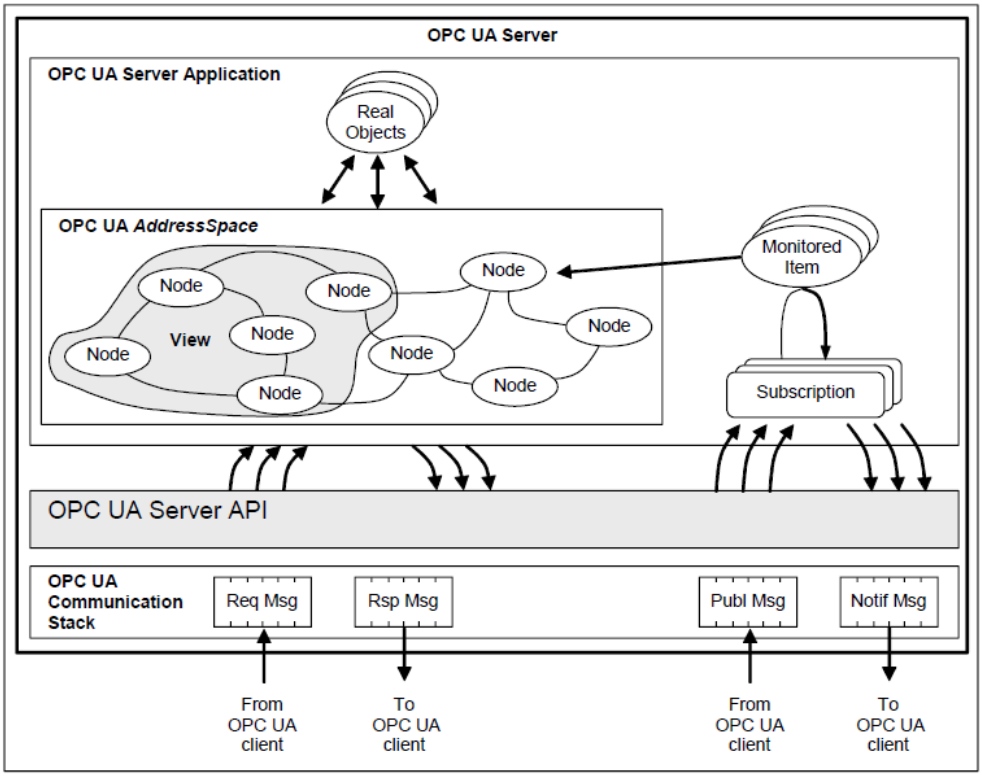


Figure 5 OPC UA server architecture [2]

- **Publisher-Subscriber (PUB/SUB) model:** an OPC UA server makes configurable subsets of information available to any number of subscribers, where the publisher (server) sends messages to a Message Oriented Middleware, without knowledge of what subscriber there may be (if there is anyone). In the same way, the subscriber expresses interest in specific data and process messages that contain this data without any knowledge of the publisher. Therefore, this model is used to communicate messages between different participants without these components having to know each other's identity. This kind of broadcasting mechanism provides an unconfirmed "fire and forget"-style exchange of information.
  There is a second approach inside the PUB/SUB model, one which does not use Message Oriented Middleware. This system solely relies on the infrastructure provided by the network to deliver messages between publisher and subscriber. In this mode a message sent will be forwarded to all members of a group, which in turn is represented by an IP address. It also decouples the communication between entities but only in space and synchronization. This direct method of sending messages has smaller latencies and is therefore considered to have better performance in comparison to the previous concept which uses a relay broker.

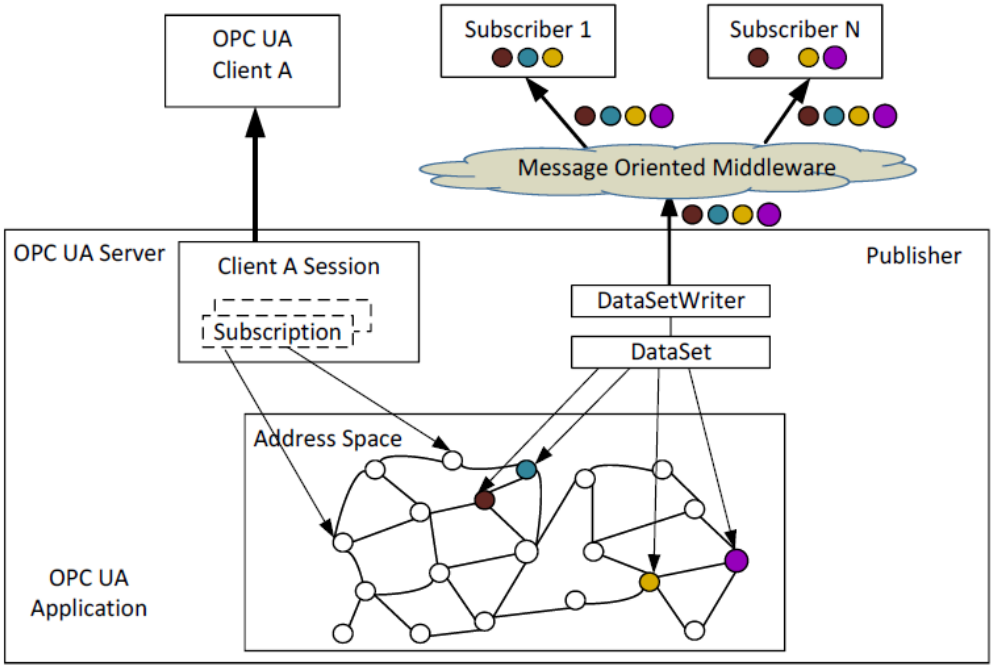


Figure 6 PubSub model integrated with Client-Server model [2]

2.1.4 OPC UA information modelling and its notation model

OPC UA provides a framework that can be used to represent complex information as Objects in an AddressSpace which can be accessed with standard services. These Objects consist of Nodes connected by References. Different classes of Nodes convey different semantics. For example, a Variable Node represents a value that can be read or written. The Variable Node has an associated DataType that can define the actual value, such as a string, float, structure etc. It can also describe the Variable value as a variant. A Method Node represents a function that can be called. Every Node has a number of Attributes including a unique identifier called a NodeId and non-localized name called as BrowseName.

OPC UA also supports the concept of sub-typing. This allows a modeller to take an existing type and extend it. There are rules regarding sub-typing defined in OPC UA OPC 10000-3 and 10000-5, but in general they allow the extension of a given type or the restriction of a DataType. For example, the modeller may decide that the existing ObjectType in some cases needs an additional Variable. The modeller can create a subtype of the ObjectType and add the Variable. A Client that is expecting the parent type can treat the new type as if it was of the parent type.

References in OPC UA allow Nodes to be connected in ways that describe their relationships. All References have a ReferenceType that specifies the semantics of the relationship. References can be hierarchical or non-hierarchical. Hierarchical references are used to create the structure of Objects and Variables. Non-hierarchical are used to create arbitrary associations. Applications can define their own ReferenceType by creating subtypes of an existing ReferenceType. Subtypes inherit the semantics of the parent but may add additional restrictions.

The notation is summarized in Figure 7. UML representations can also be used; however, the OPC UA notation is less ambiguous because there is a direct mapping from the elements in the figures to Nodes in the AddressSpace of an OPC UA Server.

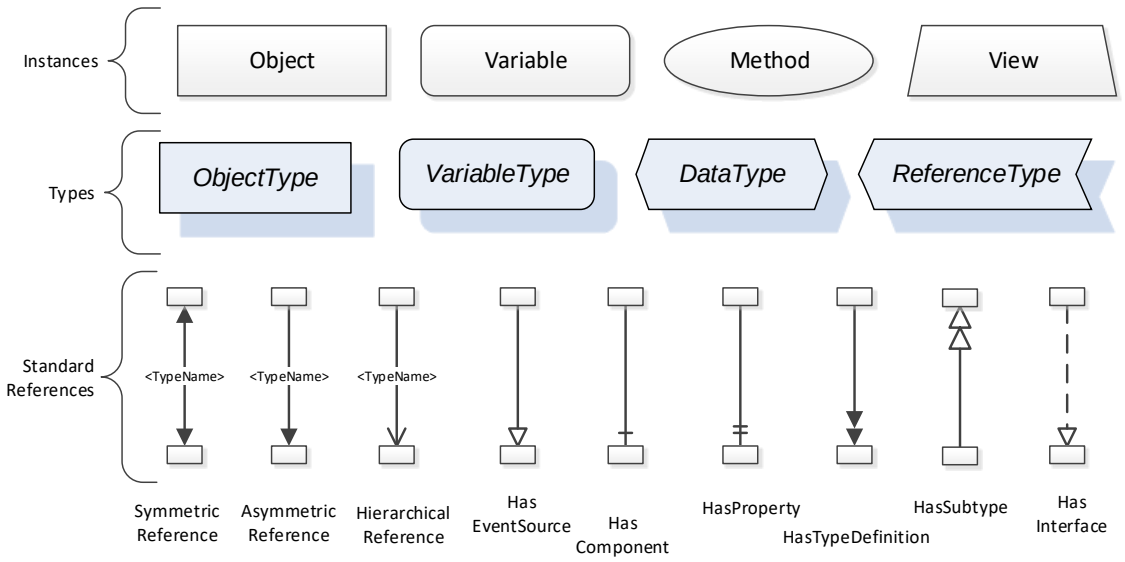


Figure 7 The OPC UA Information Model notation

A complete description of the different types of Nodes and References can be found in OPC 10000-3 and the base structure is described in OPC 10000-5 [3].

OPC UA specification defines a very wide range of functionality in its basic information model. It is not required that all Clients or Servers support all functionality in the OPC UA specifications.

OPC UA allows information from many different sources to be combined into a single coherent AddressSpace. Namespaces are used to make this possible by eliminating naming and id conflicts between information from different sources. Each namespace in OPC UA has a globally unique string called a NamespaceUri which identifies a naming authority and a locally unique integer called a NamespaceIndex, which is an index into the Server's table of NamespaceUris. The NamespaceIndex is unique only within the context of a Session between an OPC UA Client and an OPC UA Server- the NamespaceIndex can change between Sessions and still identify the same item even though the

NamespaceUri's location in the table has changed. The Services defined for OPC UA use the NamespaceIndex to specify the Namespace for qualified values.
The NamespaceUri for the Zero Swarm project is initially proposed to be as follows: 'http://zeroswarm.org/Nodesets'. At the same location, the possibility must exist to also really host the OPC UA nodesets created for the purposes of this project.
Designing decentralized data infrastructure plays critical role for scalable and flexible control systems that include multi-purpose components. In most of the human-centric production systems, components such as AGV, Cobots, assembly lines appear frequently.

2.1.5 Unifying data from different sources

Obtained data can only serve purpose while standardized for the intended use. An example of data structure that could be used to describe devices on the shopfloor might be VDMA OPC UA companion specification.
Below are the tables that include meta-data for the data flows that could be fetched from various sources (equipment output, surveys, and manually filled data) and benefit virtual commissioning of the human-centric production systems.

Table 1: Human Factors related meta-data fetched from ERP

Data Name	Description of Purpose	Source of Input*	Range	Accuracy (Resolution)	Units of measure	Update Frequency
Static						
Qualification	Acquired qualifications	ERP	Array of qualifications	N/A	ARRAY STRING	1hr
Age	Age of the worker	HR	0-99	1	years	Daily
Body Mass Index	Filled BMI of the worker	HR	10-50	0.1	m/kg	Monthly
Real-Time						
Availability	Current status of availability based on entrance to shop-floor	ERP	TRUE; FALSE	N/A	BOOL	1sec
Total Time Worked	Calculated hours worked based on Availability	ERP	0-24	0.01	hr	1sec
Time Worked on Specific Operation	Current hours worked on specific operation	ERP	0-24	0.01	hr	1sec
Performance Rate	Calculated performance rate	ERP	0-999	0.01	units/hour	1min
Defect Rate	Calculated defect rate	ERP	0-100	0.00001	% units/1000	1min

					pcs	
Indoor Coordinates	Current indoor absolute coordinates calculated from CV detection	CV	Coordinates	10cm	Coordinates	500ms
Operation	Current processing operation performed by operator	ERP	Array of operations	N/A	STRING	1sec
Break Time	Current time of the break	ERP	0-24	0.01	hr	1sec
Time Since Last Break	Calculated time from last break	ERP	0-24	0.01	hr	1sec

* HR – data from Human Resources, ERP – data from Enterprise Resource Planning system, CV – Computer Vision

Table 2: Ergonomics related meta-data fetched from HMI

Data Name	Description of Purpose	Source of Input*	Range	Accuracy (Resolution)	Units of measure	Update Frequency
Real-Time						
Heart Rate	Current heart rate from chest HRM	HRM	0-200	N/A	bpm	1sec
Heart Rate Variability Condition	Calulcated heart rate variance condition based on levels 70<X<90	HMI	Stressed; Optimal; Relaxed	N/A	STRING	30sec
Stress Flag	Calculated stress flag based on: HRV Condition, Stress or Physical Fatigue Survey	HMI	TRUE; FALSE	N/A	BOOL	1sec
Time Since Last Stress Flag	Calculated time from last stress flag	HMI	0-24	0.01	hr	1sec
Steps Count	Current steps counted from smart watch	SW	0-99999	1	steps	10sec
Heart Rate	Current heart rate from chest HRM	HRM	0-200	N/A	bpm	1sec
Heart Rate Variability Condition	Calulcated heart rate variance condition based on levels 70<X<90	HMI	Stressed; Optimal; Relaxed	N/A	STRING	30sec
Stress Flag	Calculated stress flag based on: HRV Condition, Stress or Physical Fatigue Survey	HMI	TRUE; FALSE	N/A	BOOL	1sec
Time Since Last Stress Flag	Calculated time from last stress flag	HMI	0-24	0.01	hr	1sec

Survey						
Sleep quality	At the beginning of the working day operator fills it's own assessment of sleep quality	WEB Survey	0-10	1	INT	Once, each day
Physical Fatigue	Operator fills it's own assessment of physical fatigue level	WEB Survey	0-10	1	INT	15min
Stress	Operator fills it's own assessment of stress level	WEB Survey	0-10	1	INT	15min
*HMI – Ergonomically aware human-Machine Interaction system (includes heart rate, smart wrist band, and decision-making module), SW – Smart wrist band						

Table 3 AGV related meta-data fetched from AGVs

Data Name	Description of Purpose	Source of Input	Range	Accuracy (Resolution)	Units of measure	Update Frequency
Real-Time						
Availability	Status of availability of the equipment	AGV	TRUE; FALSE	N/A	BOOL	1sec
Time Worked on Specific Operation	Current hours worked on specific operation	AGV	0-24	0.01	hr	1sec
Indoor Coordinates	Current indoor absolute coordinates calculated from lidars	AGV	Coordinates	10cm	Coordinates	500ms
Mission	Currently executed mission	AGV	Array of operations	N/A	STRING	1sec

2.1.6 OPC-UA information modelling specific to Zero SWARM

The OPC UA information model is not a singular information model, valid for all the project or for a given trial.

As shown in 2.1.4, there are several aspects to be considered when creating OPC UA information models:

- Data and domain models, following which a system, or a system-of-systems represents its entities: actors, devices, software functionalities (reference to Raffaele's domain modelling documents needed)

- Data exchange models between the entities involved in the system or in the system-of-systems.

- Event models needed to support the required functionality of the system or of the system-of-

systems.

The architectural concept of the system-of-systems under discussion in this document contains the following entities (see also Figure 8):

- Cloud level infrastructure
- Cloud-located software infrastructure allowing machine learning over data gathered over one or more edge devices or directly from the shop floor.
- Cloud-located software infrastructure allowing data gathering from edge device(s) or directly from the shop floor – e.g.: OPC UA client(s)
- Operator(s) located remotely (acting over the system-of-systems) over the cloud or locally.
- One or mode edge devices containing software infrastructure to do data exchanges with the cloud-located software entities, as described above and also to perform data exchanges with the devices at the shop floor; additionally, machine-learning software infrastructure may be placed at the edge, so that to optimize the data volumes exchanged with the cloud (which following cloud – the edge devices may be used for information aggregation coming from the shop floor systems, thus needing its own information model
- One or more AGVs at the shop floor, each of which may represent a system per se, needing its own information modelling.

The architecture of the system may be summarized as follows:

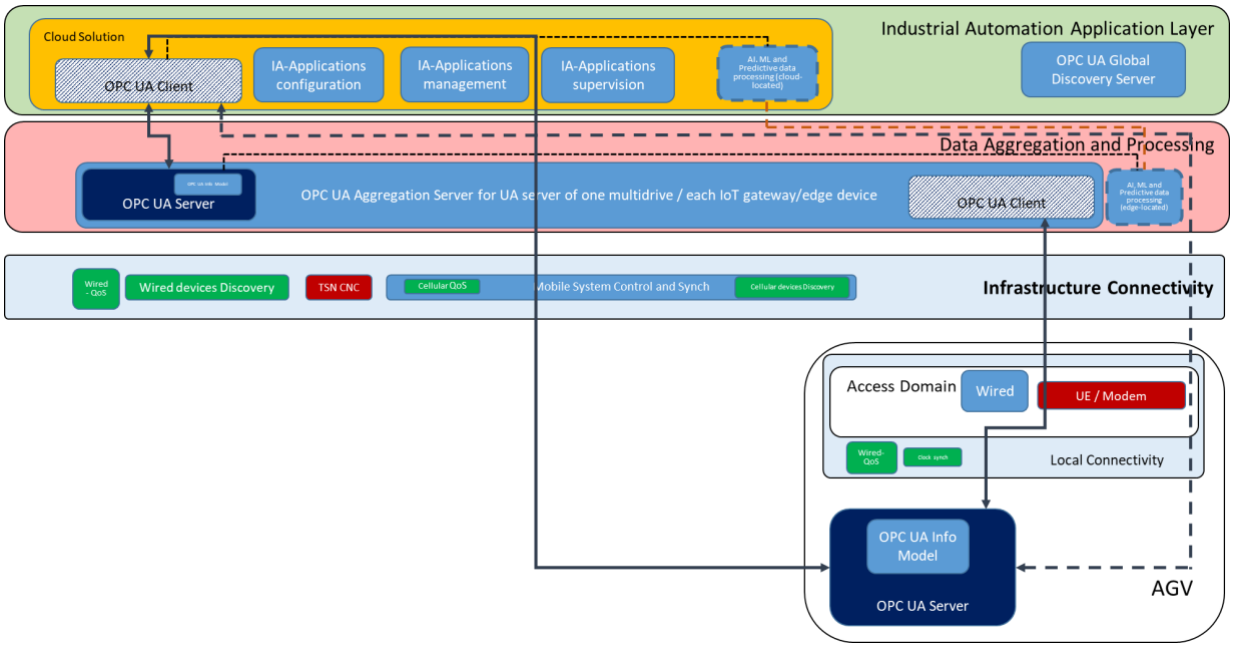


Figure 8: Simple architectural overview

Figure 8 can be related to Figure 12 by the fact the first concentrates on an architectural view or overview, whilst the latter concentrates on the machine-learning software infrastructure at edge and cloud level.

Following up on the points described above, concerning the architectural overview, an information modelling methodology may be created with regards to the OPC UA servers needed: one located at the level of each AGV device and the other one located at the level of the edge device, the latter comprising an aggregation aspect.

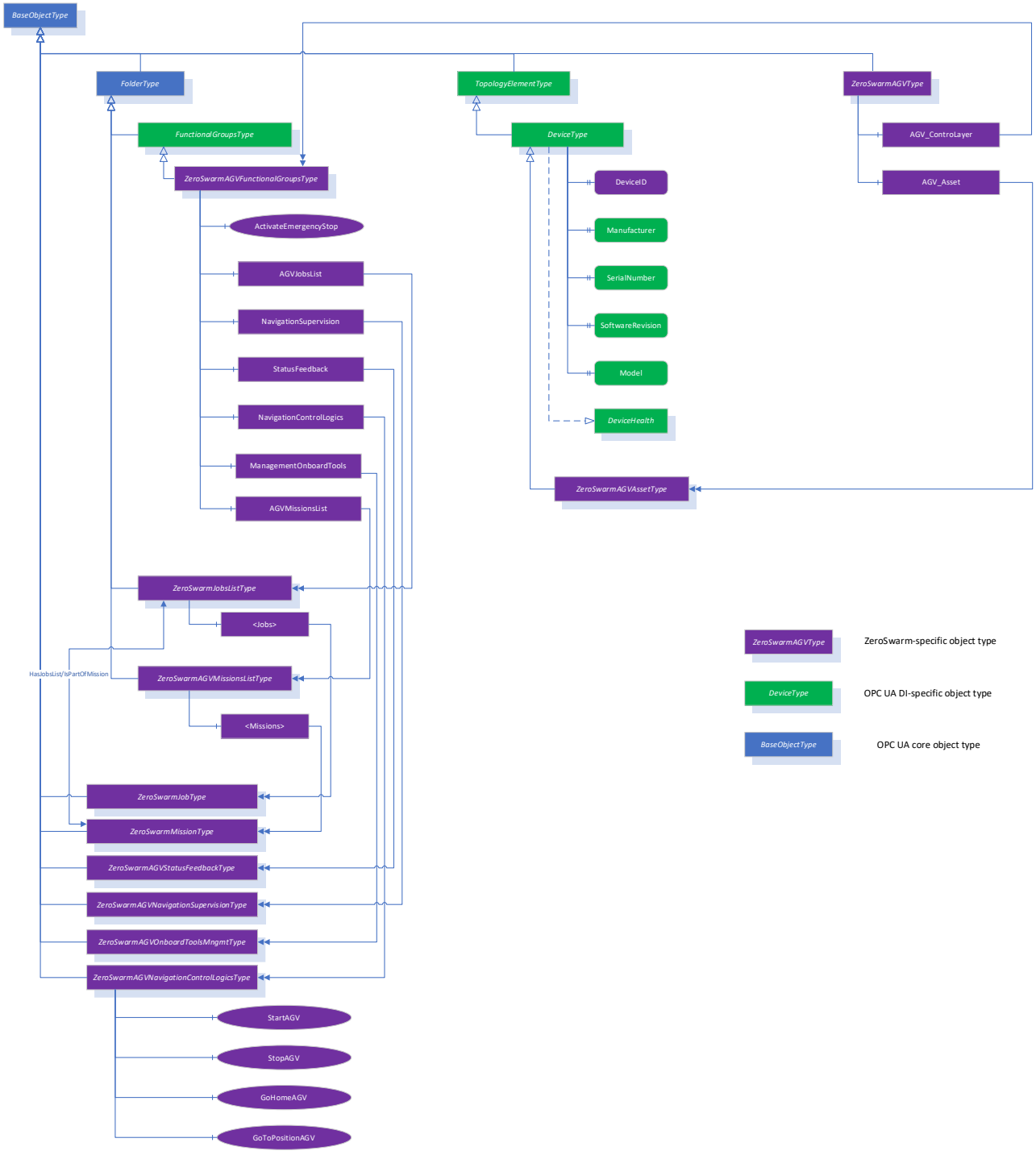


Figure 9: Zero-SWARM OPC UA Server information model example for an AGV

Furthermore, OPC UA requires the existence of a counterpart for information gathering once such information was represented in an OPC UA server. This counterpart is an OPC UA client and may bear various forms, from the simplest one, just reading data from one or more OPC UA servers and presenting it further to other software entities to realize a given business logic processing of it. This is

the case of this project where the business logic is represented by various machine learning software infrastructures, as presented in other chapters of this deliverables and of several other deliverables of this project.
In order to create the information models of the OPC UA servers mentioned above, the OPC UA core object types may be used and a baseline specification for devices called OPC UA DI (Device Integration) may be used. A color-coding for emphasizing the belonging of object types to the main OPC UA namespace or to the OPC UA DI namespace was realized.
The proposed conceptual information model for the OPC UA server running on an AGV is based on the following perspectives:

- Each AGV has two aspects: an asset aspect (with its own details, mostly covered by OPC UA DI DeviceType object type – therefore its extension with an object type defined for the purpose of this project) and a functional aspect (with its own details, as required by the domain modelling – therefore the extension of another OPC UA DI object type called FunctionalGroupType)

- An important number of newly defined object types, concerning AGV's missions, jobs and other functional aspects are derived from the OPC UA core object type BaseObjectType; an important point to mention is that currently, the OPC Foundation has released a new specification called OPC UA FX (Field eXchange) where the asset and the functional aspect of an industrial automation component are even more in detailed defined – the current modelling example though follows only its philosophy

- A new OPC UA reference type was created in order to better describe the relationship between missions and jobs of an AGV

- A software artifact must be created, next to the chose OPC UA server stack, at the level of the AGV's firmware in order to map the OPC UA information model to the capabilities of the AGV's firmware.

A detailed description (including detailed job and mission variables content, as well as navigation control and supervision of AGVs) of the object types for the OPC UA server running on an AGV may make the object of further future deliverables of this project.
Another specific OPC UA server information model can be imagined as running at the edge devices, but containing an aggregation aspect of the information potentially gathered over an OPC UA client from the various AGSs which form the system-of-systems.
Several further aspects of information modelling for an OPC UA server in the edge devices should be considered:

- The OPC UA server should be accessible for manipulation of the AGV fleet from the cloud level, potentially over an OPC UA client

- A software artifact, at the level of the firmware of the edge device, must be created which correlates the OPC UA server and the OPC UA client located both at the edge device level, so that data gathered and sent from the AGV fleet, over the OPC UA client is mapped to the information model of the OPC UA server running on the edge device (additionally, it is possible to also include data coming from other sources, e.g. the cellular network slices management)

- Several new object types should be created so that to support the CpSoS aspect of system-of-systems need to manage a fleet of AGVs, as it is depicted in Figure 10

- Furthermore, OPC UA server information model for the edge device may also combine data coming e.g. from cellular network slices management (not yet included with the example).

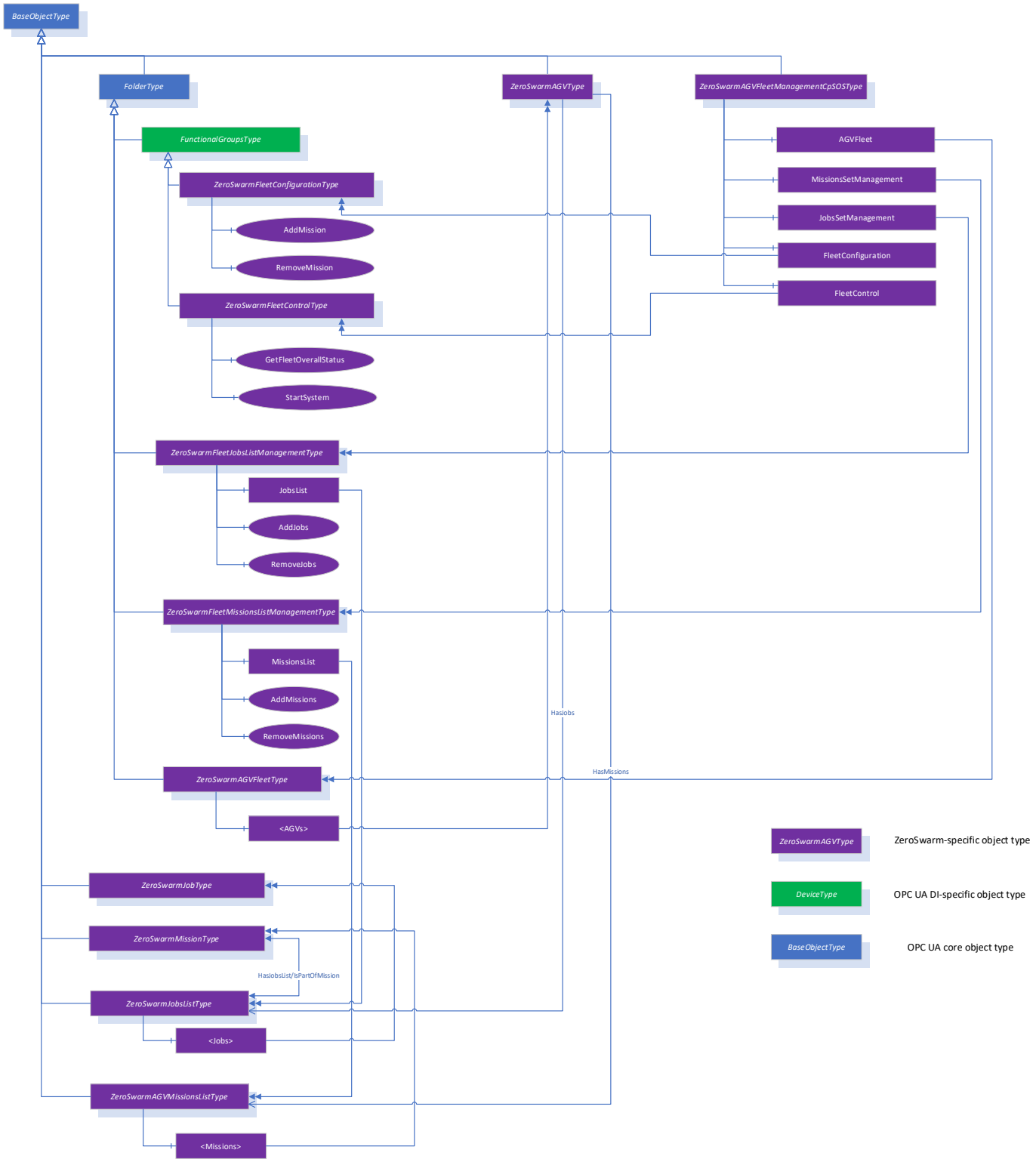


Figure 10: Zero-SWARM OPC UA Server information model example for an edge device

Instantiation is another very important element of creating an OPC UA server and its information model, as described in 2.1.4. This is required, so that an OPC UA client can connect to one OPC UA server and retrieve data as described in Figure 8 and Figure 12.

When instantiating object types, the hierarchies defined in those OPC UA object types is maintained, while not all the components of an object type need to be instantiated, given their mandatory or optional modelling rule. Thus, one can obtain a flexibly-sized information model.

In what follows, this deliverable will exemplify the instantiation specific to a potential OPC UA server running at the board of an AGV, using the UA Modeller tool from Unified Automation:

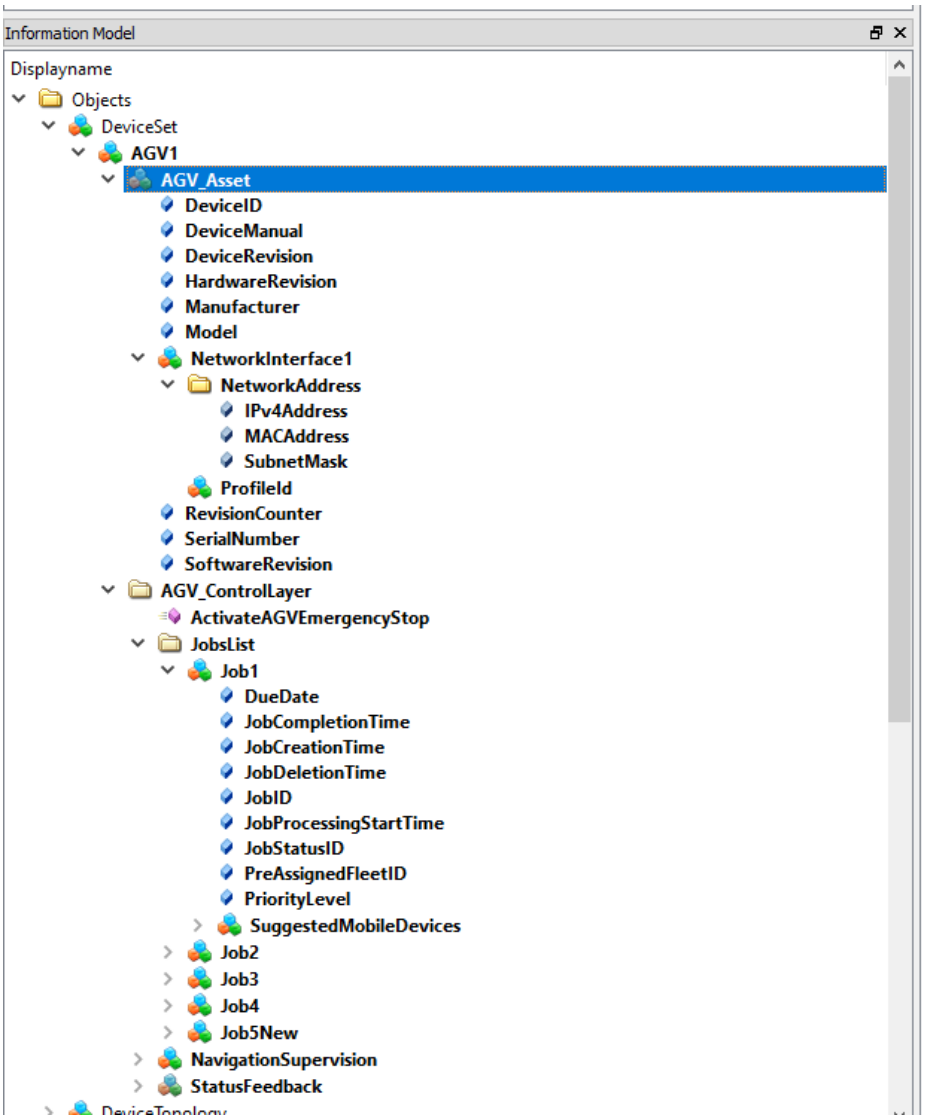


Figure 11: Instantiation of OPC UA object types for one OPC UA server running on one AGV

Further on, there is, as mentioned above, the possibility that the user of the UA Modeller modelling tool chooses which of the OPC UA object types instances' components are really used in a given information model, like the one in Figure 11.

The particular instantiation example in Figure 11 shows the two aspects one can consider for an AGV: the asset model (AGV_Asset) and the functional model (AGV_ControlLayer). Currently, not all the functions are modelled and detailed descriptions will be present in various other deliverables.

It is to mentioned that there are already some existing OPC UA specifications which cover parts of the needed information modelling needed for the various OPC UA servers information models in the Zero Swarm project, such as ISA-95-5 Job Control [4] which is an OPC UA companion specification under the name OPC 10031-4 which contains a detailed Job, Job Order and Job Response information model. Additionally, the model of an AGV is somewhat covered in the AutomationML OPC 30040 [34] and in the Tobacco Machinery OPC 30060 [35] companion specifications. Nevertheless, this modelling is not fully harmonised and it needs further evaluation in future deliverables of this project if its harmonisation must be or not in the scope of the Zero Swarm project. Such a harmonisation may need a large standardisation effort which may only partly be covered by this project budgets and plans. Therefore, a simpler information model was presented in the current subchapter.

2.1.7 Zero-SWARM solution to secure and efficient data distribution for historical and real time measurements and events

The Collecting Platform high-level architecture is depicted in Figure 12. It will integrate a number of open-source software components, adopted as baseline, which have been customized, integrated together, and extended with additional components that will be developed from scratch, like the Config Manager and the low-level data drivers.

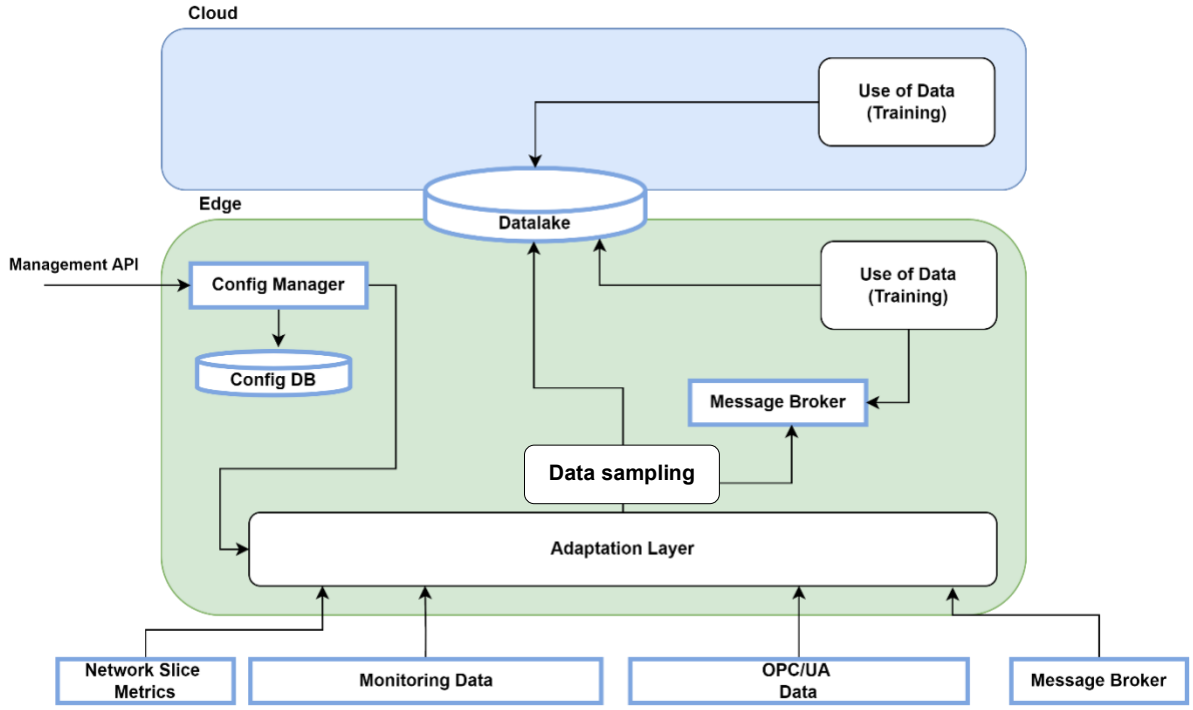


Figure 12: Collecting Platform Architecture

The concept behind the Collecting Platform is to retrieve different types of data in the shop floor and route them to specific targets. The goal of this platform is to facilitate the integration between the industry 4.0 domain with their own formats, data types and protocols, and the IT/OT domain that has other types of information.

In the High-Level Architecture of Figure 12, the main actor is the Adaptation Layer, although OPC/UA has the mechanism to route the data directly to message brokers like Kafka or MQTT, or directly to some data lakes, the Adaptation Layer can provide some initial aggregation/transformation for storing and using more interest data structure. The high-level architecture is based on the Edge computing

paradigm because the needed of quick response and fast use of data is mandatory in an industry 4.0 environment. The data are maintained near the shop floor, but they are also aggregated to be moved to the cloud environment for a heavier training or for a possible sharing mechanism with other industry 4.0 realities.

A group of Telegraf Docker containers make up the Adaptation Layer, with each container corresponding to a specific data source.

Telegraf is a server agent that utilizes plugins to collect metrics from various inputs and transfer them to a variety of outputs. Customized Telegraf plugins have been developed to gather information from different data sources, and these plugins process and convert the collected data into a shared format by using processor plugins before sending it to the Message Broker bus (as a Data Stream) and a Data Lake through two output plugins.

The Collecting Platform collects data from several data sources with different levels of granularity.

Reduced aggregation facilitates increased information at the expense of greater storage requirements. Accordingly, it is recommended that different levels of aggregation be utilized to suit different requirements and timeframes. To facilitate real-time data usage, highly detailed, or raw, data can be employed with a limited retention period to optimize storage capacity.

Storage capacity is the total amount of data that a system can store. The available storage capacity of a system can determines the level of data granularity that can be sustained. Insufficient storage capacity may necessitate the storage of generalized data, as highly granular data would rapidly consume the available space. Conversely, even a high storage capacity may not enable the long-term storage of real-time data from numerous shop floors due to the overwhelming number of raw data involved.

As is depicted on Figure 13 and Figure 14, the greater the granularity of the stored information, the greater the amount of space required to store it, and thus the shorter the time interval that can be stored in the data infrastructure. On the other hand, the greater the data aggregation, the less space is needed to store it, but the less detailed the information in the saved data.

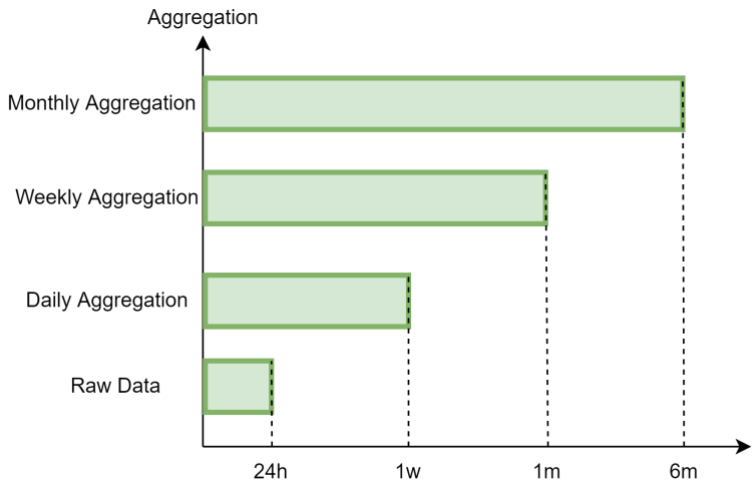


Figure 13: Aggregation-Time storage

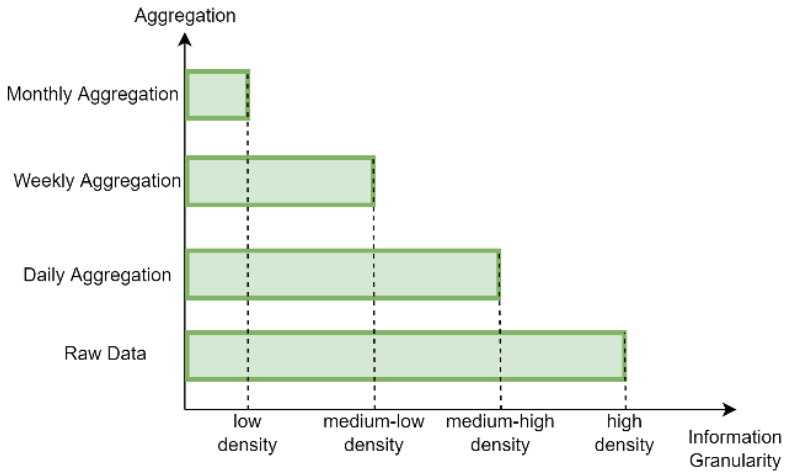


Figure 14: Aggregation-Granularity storage

Moreover, the Collection platform will be remotely managed by a Config Manager, in order to let the operators or the end users choose what data needs to be collected, and what data needs to be shared and used. With this Config Manager, the Collection platform can be placed and be used in different types of shop floor without concerning the low-level design of each shop floor.

Collecting platform will pose a critical component for the overall performance of the system and a specifically for the training procedure. Overall performance of the training depends heavily on the volume as well as the quality of the data. Collection platform will employ techniques to ensure that the appropriate amount of data is stored and that they are of high-quality.

Large amounts of data are not only computationally expensive to process but also need a large volume of resources for storage. On the other hand, extensive research has proved that not all data are equally useful and beneficiary during training [5]. Low-quality data include bias, outliers, duplicate samples, or samples that do not offer any new information. Existence of such data hinder performance of the training by increasing the time required to converge and at the same time increase storage. Especially in the case of deep neural networks research has demonstrated that once the initial training phase is complete, most of the data do not provide new information to the model and thus could easily be ignored. Research has been focused on discovering “information-rich” subsets of the data to accelerate training and decrease the resources required [5],[6],[7],[8]. Significant performance can be achieved by training only on a small fraction of the dataset using data sampling techniques. Data sampling techniques aim to reduce the resources required while maintaining the representativeness and performance of the model. This can result reduced computational and storage costs and thus improving system efficiency without sacrificing performance.

Commonly used data sampling techniques include random sampling, stratified sampling, importance sampling [8], active learning, mini-batch sampling, and hard example mining. The choice of the technique will be depended on the specific characteristics of the dataset, the learning task, and the available resources.

2.1.7.1 Data sampling for federated learning

Federated learning though, also suffers from heterogeneity problems. Data in real-world federated settings are imbalanced and not independent and identically distributed (non-IID). By incorporating data sampling techniques in federated learning, the training process can be made more efficient and

effective. It will allow the system to prioritize the most informative data samples, leading to faster convergence, improved model performance, and reduced communication and computational costs.
Most of the data sampling techniques in centralized learning require inspecting the whole dataset which is not feasible in the distributed and privacy preserving setting of federated learning. Recent studies in federated learning focus on addressing the challenges of heterogeneity [9],[10],[11] and limited resources [12] by employing data sampling techniques in a privacy preserving manner.
Given the different characteristics of training algorithms between centralized and federated learning different approaches will be employed. Data sampling strategies will evaluate the quality of the raw data online allowing only high-quality data to be stored. This will result in reducing the volume of data that needs to be stored and reducing the computational costs. In addition, it will improve convergence and performance of the trained models. Lastly, such strategies can be applied in an iterative manner to re-evaluate the data stored after training and discard samples that are no longer required.

2.2 Data Integration

The actual change of paradigm from mass production to mass customization demands to companies of higher resilience, sustainable production and more flexibility. Industry 4.0 enables companies to achieve these goals due to the analysis of data of products with Artificial Intelligence (AI). However, the lack of high-quality data or poor diversity could lead to inaccurate AI models. This can be solved with cross-company collaboration, sharing the data in a decentralized infrastructure, which lead to open, dynamic and collaborative systems of data structures.
Nowadays, sharing data in a multi-party structure leads to a lack of interoperability, since there is no data reference standard integration to be implemented in the I4.0. This gap can be solved using a standardised data exchange in form of Asset Administration Shell (AAS) to encapsulate all information of the asset.
The use of the AAS model over communication protocol standards, such as OPC UA or MQTT, will allow multiple assets from different companies to interact and be interoperable with different information model that are shared in federated data infrastructure, enabling collaboration between third companies. Therefore, using AAS will give a harmonized structure of data and metrics, achieving the data heterogeneity needed to implement a unified view of information in the decentralized approach to data management between multiple parties.
The use of the AAS to model data in decentralized federated structures is something that is not widely implemented. At ZeroSWARM, this new paradigm will be used to demonstrate cross-company data access and collaboration.

2.2.1 Asset Administration Shell

The Asset Administration Shell (AAS) has been defined by RAMI4.0 which provides a 3-dimensional structure view of an asset through its entire life. The AAS is the model with the highest degree of maturity for mapping asset information across the entire lifecycle of it, turning the assets into digitally manageable assets.

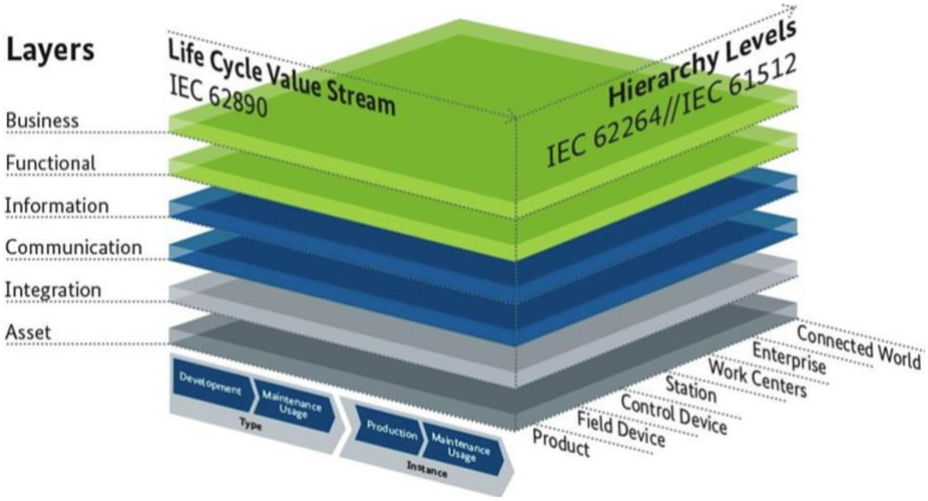


Figure 15: three-dimensional RAMI's structure

The Asset Administration Shell is standardized in IEC 63278-1 where it is defined as a standardized digital representation of an asset [13], which is a physical, digital or intangible entity that has value to an individual or an organization. Using this representation, an asset is identified as an entity in a specific state of its life providing all the technical functionality and communication ability contained by it.

2.2.2 AAS metamodel

The AAS is implemented by a generic technology-neutral manufacturer-independent standardized interface to manage the asset information which is called metamodel [14]. The metamodel describes the overall structure of Asset Administration Shells and it submodels, which are the representation of the aspects of the asset.

The AAS organizes the asset information as a tree of submodels to manage its complex information, where each submodel is the representation of an aspect of an asset. A submodel is used to structure the digital representation and technical functionality in a set of submodel elements, which can be properties, references, relationships, operations, etc [14].

The submodels can be standardized and become submodels templates and it is essential for interoperability. The Industrial Digital Twin Association (IDTA) (Industrial Digital Twin Association, s.f.) [15] works in the submodel standardization and provides a list of potential common submodels like *Identification*, *TechnicalData*, *ConfigurationData* or *OperationalData*. But other submodels that describes specifics functionalities of the asset cannot be standardized and their implementation will depend on the granularity of the model, abstraction level and use case.

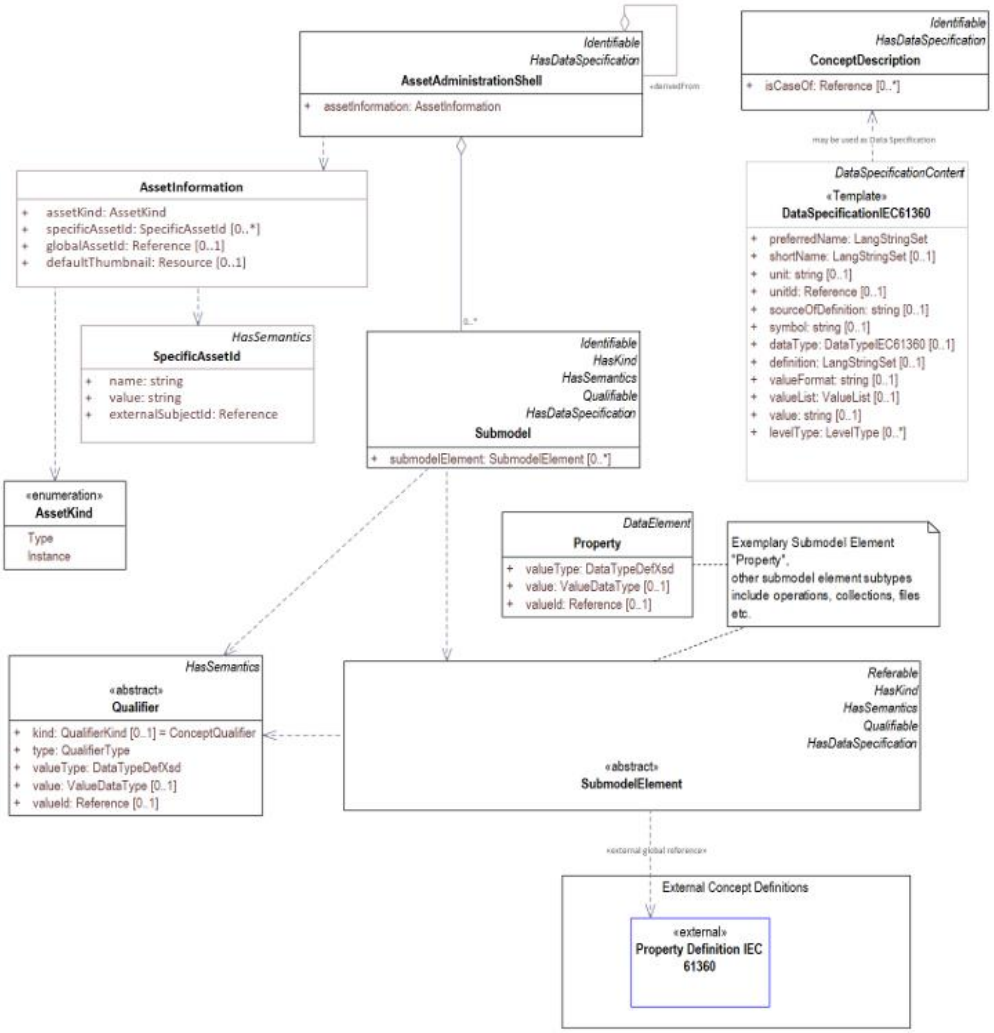


Figure 16: Overview Metamodel of the Asset Administration Shell [14]

2.2.3 AAS metamodel representation in OPC UA information model

OPC UA is the suitable for the operating phase of Asset Administration Shells and especially applicable in case of machine-to-machine communication. The works of the mapping to the OPC Unified Architecture are carried out in a joint working group between OPC Foundation, ZVEI and VDMA to map the AAS metamodel into the OPC UA information model [16]. This map must be used to implement the Industrie 4.0 conformant digital twins based on OPC UA as implementation technology, in order to represent AAS and their submodels in the address space of OPC UA servers.

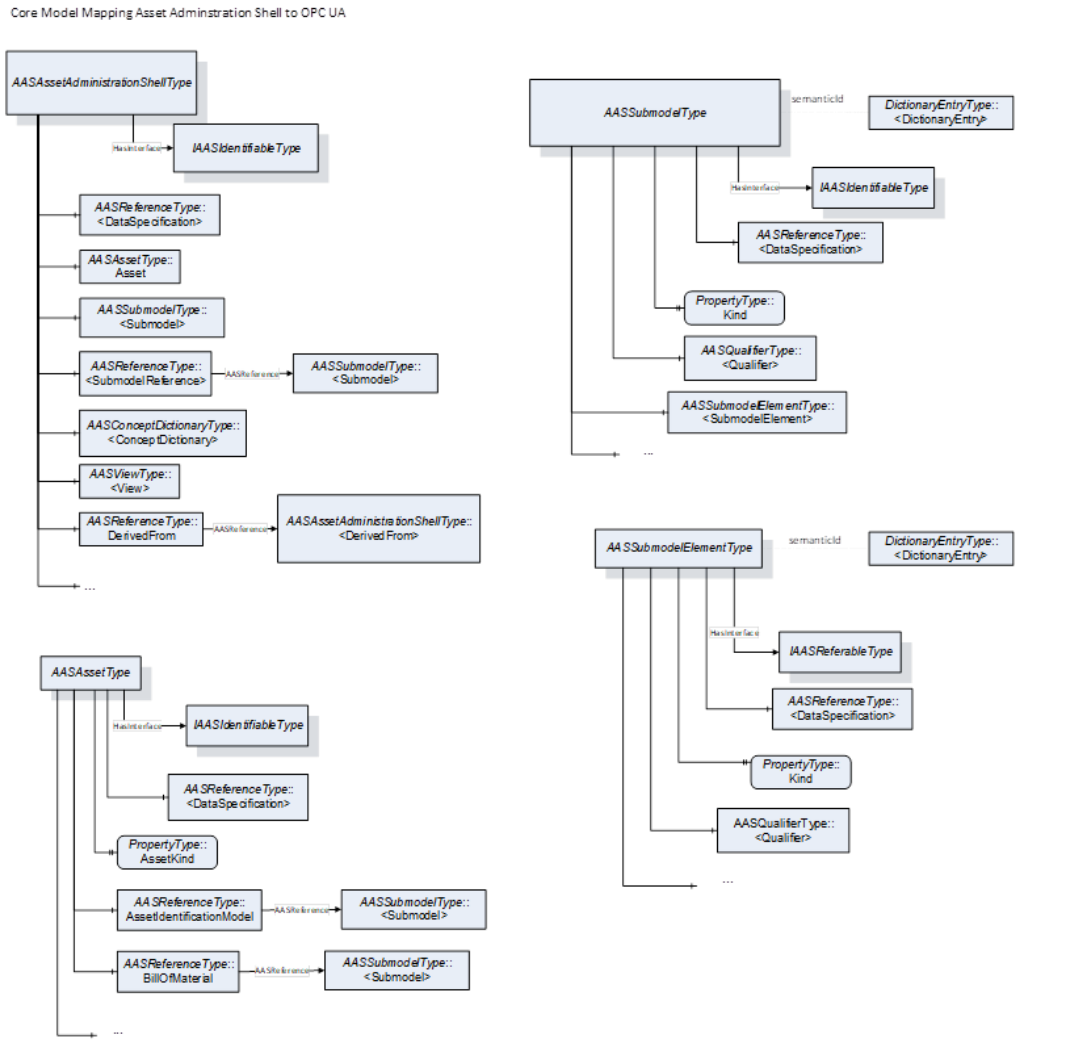


Figure 17: Overview of AAS in the OPC UA information model [16]

In this front, within the framework of WP4 the project will take some initial steps which will be reported in the following deliverables of the project.

2.3 Data Security and Privacy

Deliverable "D2.3 Cyber security implementation templates and methodological approach" [17], researched and reviewed on different cybersecurity related standards that describe the importance of a standardized approach when implementing security mechanisms to protect from cyber threats exposed to operational technologies OT environments, Industrial Automated Control Systems (IACS), Industrial Internet of Things systems (IIoT) and in general cyber physical system of systems (CPSoS).
The cybersecurity standards mentioned in the D2.3 are:

- IEC/62443
- NIST 800-53
- DIN SPEC 27070 (based on IEC 62443-4-2)

These references mentioned can be updated also with concrete OT domain with

- NIST 800-82r3. Guide to Operational Technology (OT) Security

Also mention as a general and broadly accepted reference for implementing cybersecurity management system is IEC/ISO 27001 which provides a framework for establishing, implementing, maintaining, and continuously improving an Information Security Management System (ISMS).
It is also worth mentioning the work done by the European Cybersecurity Organization (ECSO) [18] on organizing cybersecurity standardization landscape. This report provided an extensive state of the art on the standardization landscape for the cybersecurity domain also including certification schemes in different sectors and verticals related linked with the zero-SWARM scope and interest such as

- Industry 4.0 and ICS
- IoT device vendors
- Critical infrastructures
- Smart cities and smart buildings
- Secure software development

Deliverable D2.3 provides an overview of how different architectures have included a set of cybersecurity mechanisms or security controls transversal or vertical to the different layers of the architecture showed in different architectures approaches such as Reference Architectural Model Industry RAMI 4.0, Industrial Internet Reference Architecture (IIRA) v1.9, OpenFog reference Architecture (OpenFog RA), IoT IEEE P2413 or the Arrowhead Framework.
Finally, deliverable "D2.3 Cyber security implementation templates and methodological approach" also consider a possible engineering approach to consider the five layered architecture described in IEC 62443 to provide a comprehensive implementation broadly used in the industrial domain as a best practice approach.

2.3.1 Cyberthreats associated with CPSoS

One of the objectives for the zero-SWARM project is the data sharing among different domains in a Cyber-Physical System of Systems (CPSoS) approach. A Cyber-Physical System of Systems (CPSoS) refers to a network of interconnected Cyber-Physical Systems (CPS) that work together to achieve a common goal. CPS are systems that combine physical and computational elements, such as sensors, actuators, and controllers, to monitor and control physical processes. A CPSoS integrates multiple CPS, often across different domains and organizations, to achieve a larger-scale objective. A CPSoS typically relies on real-time data exchange and coordination among the different CPS, which makes it vulnerable to cyber threats.
Some of the cyber threats that can affect CPSoS include:

- Unauthorized access - Hackers can exploit vulnerabilities in the CPS or the networks that connect them to gain unauthorized access to the CPSoS. This can lead to data theft, system disruption, or physical harm.
- Malware - Malicious software can be used to infect the CPS and spread across the CPSoS, disrupting system operations, and potentially causing physical harm.
- Denial of service attacks - Cybercriminals can launch denial of service attacks on the CPSoS, overwhelming the system with traffic and causing it to shut down.
- Insider threats - Malicious or negligent insiders can cause harm to the CPSoS by intentionally or unintentionally causing system disruptions or stealing sensitive data.

2.3.2 Data sharing and data federation in CPSoS

As mentioned before a CPSoS integrates multiple CPS, often across different domains and organizations. CPSoS will exchange data among CPS interconnect and will require a configuration or set of parameters that define how a device or system connects to a network or data source. A model for data connection and data sharing may include details such as network protocols, authentication methods, encryption settings, and other parameters necessary for establishing a secure and reliable connection.

Data federation in CPSoS can provide a secure exchange of data among several CPS although federating data from multiple sources raises security and privacy concerns. Protecting sensitive data during transmission, ensuring secure authentication and authorization, and addressing privacy requirements become critical challenges. Data federation frameworks need robust security mechanisms, encryption protocols, and access control mechanisms to safeguard the data.

2.3.3 Security mechanisms for secure exchange of data in industrial domains

The implementation of security mechanisms for authentication and authorization is critical in industrial scenarios where sensitive data is being exchanged. The primary goals of these mechanisms are to ensure that only authorized people or systems have access to the data and to protect the integrity and confidentiality of that data.

Authentication mechanisms are designed to verify the identity of the individual or system requesting access to the data. The goal of authentication is to ensure that only legitimate users or systems can access the data. This is typically done using a combination of usernames, passwords, security tokens, biometrics, or other authentication factors.

Authorization mechanisms are designed to control access to specific resources based on the user or system's identity and permissions. The goal of authorization is to ensure that only authorized users or systems can access specific resources. This is typically done by assigning different levels of access rights to users or systems based on their role or level of authorization.

In industrial scenarios, the implementation of security mechanisms for authentication and authorization helps to prevent unauthorized access, data theft, and data breaches. It can also help to improve the overall security posture of an organization, which is critical in protecting against cyberattacks, industrial espionage, and other security threats. Additionally, compliance with regulatory requirements is often a goal for implementing these security mechanisms in industrial scenarios.

2.3.4 Cybersecurity threat landscape associated to access control.

Authentication and authorization systems are present in IT, OT, and cloud environments but in OT there is great gap in the use of these kind of systems because of the following reasons:

- Many of the systems in the industry have been working from more than 20 years.
- The systems do not have implemented robust password verification procedures.
- They only have 2 authorization profiles, user, and admin, and in many cases, there is only one: admin profile.
- The systems do not also control how many times a user has tried to access with wrong password, so they cannot block the user or the source.
- It is a lack of login timeout, in many cases it is delegated to the source, so the user is logged in until it log-off or the connection is closed by the source.

- The systems do not have, even the new ones, the possibility to use RADIUS or 802.1X (for example) to control the access from external servers.

Additionally, there is an increasing cybersecurity threat landscape associated to the industry as the digitization of the supply chain (the growth of industrial IoT, Device-to-cloud communication, and remote access services for ICS networks) exposes the companies' systems to new threats and massive equipment damages [19]. The European Union Agency for Cybersecurity (ENISA) conducts regular assessments of the cybersecurity threat landscape in Europe.

These cybersecurity issues can be mitigated (zero-risk cybersecurity does not exist) with the observance and the alignment with standardization approaches on cybersecurity best practices and with the implementation of security controls stablished in different cybersecurity standards as mentioned in D2.3 [17] deliverable (Cybersecurity implementation templates and methodological approach.

2.3.5 Standardization and Security controls for AAA

AAA stands for Authentication, Authorization, and Accounting. AAA is a framework that provides a comprehensive approach to managing and controlling access to computer systems, networks, and resources. It encompasses three distinct but interconnected processes that play crucial roles in ensuring secure and authorized access: Authentication, Authorization and Accounting.

As mentioned in D2.3 cybersecurity mechanisms and security controls are present in the different architectures: Architectural Model Industry RAMI 4.0, Industrial Internet Reference Architecture (IIRA) v1.9, OpenFog reference Architecture (OpenFog RA), IoT IEEE P2413, Arrowhead Framework and 5G architectures.

The aforementioned architectures require the implementation of AAA and the security standards describe them as authentication, access and auditing controls which are present in the different regulations described below.

2.3.6 AAA controls in IEC/ISO 27001

The IEC/ISO 27001 standard provides a comprehensive framework for implementing an Information Security Management System (ISMS) that includes a range of controls to protect against cyber threats. Several of these controls can be mapped to security mechanisms for authentication and authorization: Annex A.9 is dedicated to different controls associated with Access Control.

- Access control (Annex A.9.1) - This control requires the implementation of access control mechanisms to ensure that only authorized users are granted access to information and system resources. This includes implementing authentication and authorization mechanisms to verify user identities and control their level of access to resources.
- System and application access control (Annex A.9.2) - This control requires the implementation of access controls for systems and applications to ensure that only authorized users can access and modify data.

2.3.7 AAA controls in NIST-800-53

NIST-800-53 [20] refers to the Special Publication 800-53 published by the National Institute of Standards and Technology (NIST) in the United States. It is titled "Security and Privacy Controls for Federal Information Systems and Organizations" and provides a comprehensive catalogue of security controls for information systems.

The publication provides a framework for security and privacy controls for federal information systems in the United States. Several controls relate to authentication and authorization, including access control, identification and authentication, and system and information integrity.
NIST SP 800-53 is widely used as a reference by organizations, both within and outside the federal government, to establish robust security practices and align with industry standards. It is regularly updated to incorporate emerging technologies, new threats, and evolving best practices in the field of information security.
Several cyber security controls in NIST SP 800-53 can be mapped to the security mechanism for authentication and authorization.
NIST-800-53 define different control families for AAA: Access Control (AC), Audit and Accountability (AU), Identification and Authentication (IA), and Incident Response (IR).
Related with AC in NIST-800-53 the following controls are detailed:

- AC-2: Account Management: This control focuses on the management of user accounts and includes requirements for strong authentication mechanisms, password complexity, and periodic password changes. It also includes controls for account provisioning, deprovisioning, and the implementation of multi-factor authentication (MFA) where necessary.
- AC-3: Access Enforcement: This control addresses the enforcement of access controls based on established policies. It includes requirements for user authentication before granting access, the use of access control lists (ACLs), and the implementation of role-based access control (RBAC) to ensure that users have appropriate authorization.
- AC-6: Least Privilege: This control emphasizes the principle of least privilege by restricting access rights to the minimum necessary privileges required for users to perform their authorized tasks. It includes requirements for user permissions and privileges to be defined and managed based on job functions and responsibilities.
- AC-16: Security Attributes: This control focuses on using security attributes to determine access rights. It includes requirements for the use of attributes such as user roles, organizational affiliations, and security clearances to make access control decisions and enforce authorization policies.
- IA-2: Identification and Authentication: This control addresses the identification and authentication of users and systems. It includes requirements for strong user authentication, the use of cryptographic mechanisms, and the protection of authentication information. It also covers the implementation of centralized authentication services and the management of authentication credentials.
- IA-4: Identifier Management: This control focuses on the management of unique user identifiers and the association of those identifiers with authenticated individuals or subjects. It includes requirements for ensuring the uniqueness and integrity of identifiers, as well as controls for managing the assignment, release, and reuse of identifiers.

2.3.8 AAA controls in NIST 800-82r3

NIST 800-82r3 provides guidelines for securing industrial control systems (ICS). Several cybersecurity controls in this standard can be mapped to the security mechanisms for authentication and authorization, including:

- Access control (section 5.1) - This control requires organizations to implement access control mechanisms to ensure that only authorized users can access critical ICS assets. This includes implementing authentication and authorization mechanisms to verify user identities and control their level of access to resources.
- Identification and authentication (section 5.2) - This control requires organizations to implement mechanisms for identifying and authenticating users, including password policies, two-factor authentication, and biometric authentication.
- Authorization and accountability (section 5.3) - This control requires organizations to establish procedures for managing user access to ICS assets and to maintain an audit trail of all security-relevant events, including authentication and authorization events.
- System and information integrity (section 5.4) - This control requires the implementation of access controls for systems and applications to ensure that only authorized users can access and modify data.
- Network security (section 5.5) - This control requires organizations to implement network security controls to protect against unauthorized access to ICS assets, including access control mechanisms, intrusion detection systems, and network segmentation.

2.3.9 Zero-SWARM approach for IEC 62443

This section does not describe access control, but rather the evaluation guides that we have proposed in D2.3 to evaluate, based on the IEC 62443-3-3 [21] standard, the security level of an industrial system based on security requirements aligned with seven categories. Foundational Requirements.

IEC 62443-3-3 provides detailed technical control system requirements (SRs) associated with the seven foundational requirements (FRs), including defining the requirements for control system capability security levels. These FR requirements are intended to be used, along with the defined zones SL and to conduit the system under study, evaluating the appropriate security capabilities at the control system level (Foundational Requirements FR, System Requirements SR or Enhance Requirements).

IEC 62443-3-3 covers security requirements and is aligned with the concept of seven foundational requirements (FR) as defined in IEC 62443-1-1. The technical security requirements are grouped according to the FRs Identification and authentication control (FR1), Use control (FR2), System integrity (FR3), Data confidentiality (FR4), Restricted data flow (FR5), Timely response to events (FR6), and Resource availability (FR7). For each of the foundational requirements, there exist several concrete technical security requirements (SR) and requirement enhancements (RE) and these are assigned to the 5 security levels according to the level of threat mitigation provided. In the context of communication security, these security levels are specifically interesting for the “conduits” connecting different zones.

To perform the access control system in zero-SWARM, the following Foundational requirements FR, from IEC 62443, can be mapped: FR1, 2 and 3 [Identification and Authentication Control (IAC), Use Control (UC) and System Integrity (SI)].

Effectively, 62443 lays out a roadmap to engineer cyber security defences which is in the scope of zero-SWARM, and to iterate between risk assessments and system design until an acceptable level of protection is deployed.

2.3.10 Recommendations and implementation guidelines for AAA frameworks

Access control management in the form of API Keys, Identity Provisioning tools or other AAA schemes need to be put in place to control the access to specific datasets used in zero-SWARM as described in previously in this section in D4.4 and detailed in D2.2 and D2.3.

2.3.11 Identity provisioning solutions

Identity provisioning refers to the process of creating, managing, and deactivating user accounts and their associated access privileges within an organization's digital systems and applications. An identity provisioning solution, also known as an identity and access management (IAM) provisioning solution, automates and streamlines these processes, ensuring that users have the right level of access to the resources they need while maintaining security and compliance.

Some IAM solutions can be summarized below:

- FIWARE Keyrock. Keyrock is the Identity and Access Management (IAM) component of the FIWARE platform, which is an open-source framework for building smart applications and services. Keyrock provides authentication, authorization, and user management capabilities for applications developed using FIWARE technologies.
- IAM Keycloack. Keycloak is an open-source Identity and Access Management (IAM) solution that provides a comprehensive set of features for managing user identities, authentication, and authorization within applications and services. It is developed by Red Hat and offers a scalable and flexible platform for implementing IAM capabilities.

2.3.12 API gateway solutions

API Gateway is a concept in software architecture that acts as an intermediary between client applications and backend services, providing a unified interface for accessing multiple APIs. It serves as a single-entry point for all API requests and offers various functionalities such as authentication, authorization, request routing, rate limiting, caching, logging, and monitoring. API Gateways help simplify the development process, enhance security, improve scalability, and enable easier versioning and management of APIs.

Some API gateways implementations can be summarized below:

- Kong API gateway. It is an open-source platform built on top of Nginx, designed to manage, and secure APIs at scale. Kong provides a flexible and extensible framework for API management, allowing organizations to control the flow of API traffic, apply authentication and authorization policies, transform, and route requests, and collect analytics and monitoring data. It supports various protocols and standards like REST, WebSocket, gRPC, OAuth, and JWT, making it suitable for diverse API ecosystems.
- Amazon API Gateway: A fully managed service provided by Amazon Web Services (AWS) that allows developers to create, publish, monitor, and secure APIs at any scale. It integrates well with other AWS services and offers features like caching, throttling, request transformation, and AWS Lambda integration.
- Apigee API Platform: A comprehensive API management platform offered by Google Cloud. It provides tools for building, analyzing, and securing APIs, along with developer portal features, traffic management, and analytics capabilities.

- Azure API Management: A service provided by Microsoft Azure for creating, publishing, and managing APIs. It offers features like developer portal, authentication, rate limiting, caching, and monitoring, and integrates well with other Azure services.
- Tyk API Gateway: An open-source API Gateway that offers features like rate limiting, access control, analytics, and developer portal. It can be deployed on-premises or in the cloud and supports various protocols and authentication mechanisms.

The security mechanisms chosen to guarantee the secure exchange of data between the CPSoS, implementing the AAA framework, must map to a security level (SL) associated with the foundational requirements (FR) described in IEC-62443-3-3.

2.3.13 Summary and further work in cybersecurity for zero-SWARM

There is an important concern on implementing cybersecurity and privacy in IoT, IIoT and all Industry 4.0 based solutions, and there is an extensive threat landscape as mentioned before. Industrial components were not designed thinking in cybersecurity and now they are exposed to incoming risks from Internet. Another reason is because in industry is not usual to update or upgrade the systems with the newest firmware or OS patches, so it is a need to protect the systems against the risks that can be exploited in the network.

D2.3 describes a methodology and guidelines to implement cybersecurity by design in as a novel approach as treated in zero-SWARM.

CPS and CPSoS are complex solutions that have several and different components working altogether. This means that the hole system is as week as the weaker component that is part of it, and as there is no way of knowing what is inside of each CPS or CPSoS we need to provide a solution to protect the Systems against different kind of attacks, and we need to develop a solution compatible with all those CPS and CPSoS. Usually, the components provide basic security solutions that are suitable for basic operations but no for complex systems where the security is important. The solution that is going to be developed will be compatible with each CPSoS, will fit with cybersecurity standards to ensure that the data exchange and communication between CPSoS are secure and reduce the risk known cyberthreats explained before.

In this way, the implementation of the methodology and guidelines described in D2.3 complemented with cybersecurity solutions as software modules for vulnerability assessments, cybersecurity monitoring and event detection and incident response will include additional to gain cybersecurity maturity in the zero-SWARM solution:

1. Penetration testing module: It will detect vulnerabilities and will determine the attack surface and it will inform the next module about this data.
2. Anomaly Detection module: It will detect known attacks and anomalous behaviour, analysing the network traffic and informing about ongoing attacks and/or anomalies to the next module.
3. Mitigation Engine Module: It will decide the best strategy to counter/handle the attacks and/or anomalies, reported from the previous module, based on a predefined list of actions that depends on the component (hardware/software) attacked. It will inform about the decided strategy and predefined list of actions based on the component attacked to the next module.
4. Hypothesis testing module: It will compare different mitigation strategies based on static KPIs and statistical difference, and it will implement the response, based in the data received from detection module and in historical data, to mitigate the attack or the anomaly detected.

These modules are being developed in WP5 and it will be reported in the deliverables D5.4 (Section 7) and D5.5 (Section 5 and 6).

2.4 Edge computing as an enabler of AIC

Edge computing is a form of distributed computing that brings data processing, storage, and applications closer to the source of data generation. Instead of sending data to a centralized location for processing, edge computing processes data closer to where it is generated, often at the network edge, which is in close proximity to the shop floor devices such as sensors, cameras, and IoT devices.

It is possible to define and to see the edge computing as a foundation layer of the Zero SWARM AIC architecture. Edge computing enables the rapid processing of data generated by devices in real-time, which allows for faster decision-making and reduced latency. This technology is critical to supporting emerging technologies like 5G and IoT, which require fast and reliable network communications. Edge computing also has significant security advantages. By processing data closer to the source, edge computing can reduce the risk of cyber-attacks. This is because data is processed locally, reducing the risk of a breach during data transfer to a centralized location.

Edge computing involves relocating part of the storage and computing resources away from the main data centre to the origin of the data [22][33]. This replaces the process of transmitting raw data to the central data centre for analysis and processing, and instead carries out those tasks where the data is initially generated. This could be a retail store or factory floor. The outcome of this computing action is then forwarded to the main data centre for examination and further human input, such as real-time business insights, equipment upkeep predictions or other practical advice.

The Edge connection plays a significant part in ensuring that coverage reaches all necessary locations speedily. While full 5G does not guarantee full coverage on its own, Edge Computing can facilitate effective communication between 5G and any connected application and any connected device. Using the cloud for such processing will quickly become costly, and the consumer experience will be affected. Therefore, Edge Computing is crucial for 5G to handle its processing responsibilities effectively. By having the entire system processed through local Edge networks, data can be evaluated and rationalized before being forwarded to a centralized cloud, improving the processing procedure. This further encourages application creators to use the new 5G network, allowing it to grow in parallel with Edge Computing.

Edge computing and 5G provide numerous advantages for end-users. One of the 5G's principal goal is to offer significantly improved service quality and reduce latency. By collaborating with Edge Computing, 5G can transmit data quickly within devices and applications such as self-driving and navigating vehicles/AVGs. Edge Computing processes data within its local network and then passes on all appropriate information to the 5G network, allowing self-driving and navigating devices to receive the information in just few milliseconds. Without Edge Computing, and by relying on the cloud, it can cause an unacceptable delay when working with smart devices, robots and AGVs.

The increasing amount and time-sensitivity of data produced by organizations today has led to emerging network problems that edge computing has become a relevant solution for. The rise of autonomous vehicles, for example, creates a huge demand for real-time data exchange between vehicles and traffic control signals, which requires a fast and responsive network. Edge computing

addresses three limitations of the network: bandwidth, latency, and congestion or reliability. More specifically, bandwidth refers to the amount of data a network can carry over time and all networks have finite limits. Increasing bandwidth can be expensive and still does not solve other problems. Latency is the time it takes for data to move across a network and delays can be deadly in, for example, autonomous vehicles. Congestion can cause high levels of delay and even outages for Internet of Things (IoT) devices. Edge computing can address these issues by deploying servers and storage where data is generated, creating a smaller and more efficient local area network (LAN) with exclusive access to ample bandwidth. Local storage protects raw data, and local servers can perform edge analytics or pre-process data to make real-time decisions before sending results or essential data to the cloud or central data centre.

2.4.1 Edge computing and Industry 4.0

In the scope of the I4.0 revolution, traditional centralized cloud computing architectures face limitations in meeting the stringent requirements of real-time data processing, low latency and high bandwidth. Edge computing emerges as a powerful paradigm to overcome these challenges by bringing computational capabilities closer to the data sources and applications at the network edge. Based on literature, the significance of incorporating IIoT and edge computing has been outlined, trying to clarify the importance of the future of edge computing in IIoT. One of the key benefits of edge computing in Industry 4.0 is its ability to enable real-time data processing. By placing edge devices in close proximity to data sources, such as IoT sensors and industrial equipment, latency can be significantly reduced. Real-time data analysis at the edge empowers timely decision-making, enabling immediate actions and responses to changing conditions on the factory floor. This capability enhances operational efficiency, quality control and overall productivity. Moreover, Industry 4.0 applications often require low-latency interactions for mission-critical operations. Edge computing provides the necessary infrastructure for such applications by reducing round-trip times between data sources and processing nodes. For example, in collaborative robotics or autonomous systems, edge computing enables fast response times, ensuring safe and efficient operation. In addition, augmented reality (AR) and virtual reality (VR) applications benefit from edge computing, delivering seamless and immersive experiences by minimizing latency [23].

Not limited to the aforementioned benefits from using edge computing in the era of Industry 4.0, it also helps optimize bandwidth utilization by reducing the need to transmit large volumes of raw data to the cloud for processing. Instead, edge devices can perform data filtering, aggregation, and pre-processing locally. By sending only relevant and valuable insights to the cloud, bandwidth usage is significantly reduced, resulting in cost savings and improved network efficiency. This is particularly crucial in scenarios where network connectivity is limited or unreliable, such as remote industrial sites or mobile applications. Another important aspect, which should be taken into account is the insurance of data privacy and security in an industrial environment. Edge computing addresses these concerns by keeping sensitive data within the local environment, minimizing the risk of unauthorized access or data breaches. As data is processed and analysed at the edge, the need to transmit sensitive information to the cloud is reduced. This localized approach enhances data privacy, compliance with regulations and safeguards critical intellectual property [33].

2.4.2 Edge learning

Edge computing also enables the distribution of intelligence across the network infrastructure, empowering autonomous decision-making capabilities at the edge. By leveraging edge devices equipped with AI algorithms and machine learning capabilities, real-time data analysis can be performed locally, enabling autonomous systems to respond to local conditions without relying on constant cloud connectivity. This distributed intelligence enhances the autonomy, responsiveness and adaptability of I4.0 systems, allowing them to operate even in disconnected or intermittent network environments. To this end, edge learning will play a crucial role in enabling intelligent and autonomous systems, empowering real-time decision-making and driving efficiency and productivity in the industrial sector [24].

2.4.3 Federated learning

Federated learning is a distributed machine learning approach that allows training models across multiple edge devices or nodes while keeping the data local and preserving privacy. Federated learning has emerged as a potent tool for facilitating distributed advanced analysis in the realm of industrial IoT. This model training technique empowers data owners to conduct local model training while safeguarding data privacy and curtailing communication expenses. Significant efforts have been dedicated to developing sophisticated federated learning algorithms aimed at enhancing learning performance, encompassing aspects such as privacy preservation and learning efficiency [25]. Federated learning is particularly suitable for edge computing applications, effectively harnessing the computational capabilities of edge servers alongside the data gathered from geographically distributed edge devices. The allure of federated learning has captured the attention of numerous users and there are several notable advantages worth highlighting.

More specifically, the first advantage negotiates the Reduced Training Time. In other words, by utilizing multiple devices to compute gradients in parallel, federated learning significantly speeds up the training process, leading to faster model convergence and reduced training time. Another advantage of the use of Federated learning is the Decreased Inference Time. As each device maintains its own local copy of the model, predictions can be made swiftly without relying on slow queries to the cloud. This results in reduced inference time, enabling real-time or near-real-time decision-making. Furthermore, Enhanced Privacy Preservation can be considered as benefit. Federated learning addresses the privacy concerns associated with uploading sensitive information to the cloud. By keeping data local and only sharing model updates, privacy risks are minimized, making it particularly suitable for applications where data privacy is critical, such as healthcare devices. Last but not least, the use of federated learning Facilitates Collaborative Learning, which means that it enables a form of crowdsourcing where data collection and labeling can be distributed among participating devices. This collaborative approach simplifies the data acquisition process, reduces the burden of collecting a massive centralized dataset, and saves time and effort in data preparation.

2.4.4 Management of storage, computing, and network resources

The following section provides a brief presentation of the functionalities provided by two open-source platforms, OpenStack, and Open-Source MANO (OSM), both of which are very commonly used [26],[27] in the telecommunication domain for VNF management. These platforms handle VNF management with different capabilities, but can be paired to provide complementary functionalities.

OpenStack is an open-source cloud computing platform [28] that provides a wide range of functionalities for building and managing private and public clouds. It advertises a modular and extensible architecture that can enable flexibility and scalability, making it suitable for a wide range of cloud deployment scenarios. It offers a set of software tools that allow managing large pools of compute, storage, and networking resources. The key functionalities of OpenStack, provided by modular components be summarized as the following: it provides efficient handling of VM and instance provisioning, allowing users to retrieve, launch and manage VMs stored in a secure registry, define networks, and allocate and monitor available resources. Additionally, it provides features like virtual networking, load balancing, and firewall services for creating complex network topologies. It offers services to handle identity management and centralized authentication and integration with systems like Lightweight Directory Access Protocol (LDAP) and Active Directory. Orchestration is handled by the so-called Heat component, which automates infrastructure resource deployment using templates to describe the desired cloud infrastructure, including instances, networks, storage, and their interdependencies.

Finally, OpenStack provides multiple storage solutions that simplify the provisioning, scaling, distribution and management of relational and non-relational databases within the OpenStack environment.

OSM (Open-Source MANO) is an ETSI-hosted open-source project [29] that provides a comprehensive management and orchestration framework for Network Functions Virtualization (NFV) environments, aligned with the ETSI VNF specifications [30]. It aims to simplify the design, deployment, and lifecycle management of virtual network services. OSM encompasses several key functionalities for network service management and orchestration. It enables network service designers to define and model complex services using templates, describing the components, relationships, and behaviour of virtualized network functions (VNFs) and their interconnections. It automates the deployment and management of VNFs, orchestrating service deployment policies, resource allocation, and scaling rules while making use of vendor-neutral information models. It provides a standardized process for on boarding VNFs into the management framework, ensuring compatibility and interoperability. OSM handles resource management in a dynamic manner, provisioning and scaling compute, storage, and network resources based on service demands. Fault and performance management capabilities in OSM include real-time monitoring, logging, and alerting to detect and respond to issues. The lifecycle management of network services, including instantiation, scaling, healing, updating, and termination, is supported. OSM offers service assurance mechanisms through testing, benchmarking, and verification processes. It supports multi-VIM environments, working across different virtualization platforms. Additionally, OSM provides integration capabilities with external systems and interfaces, enabling operators to integrate with existing network management systems and align NFV operations. Overall, OSM provides a robust and flexible management and orchestration framework for NFV environments.

OSM and OpenStack are both open-source software platforms that can efficiently handle VNF management, but they have distinct purposes and architectures. OSM is primarily focused on managing and orchestrating network functions and services in virtualized environments. OpenStack, on the other hand, is a computing platform that enables the creation and management of public and private clouds. It offers infrastructure as a service (IaaS) capability to provide a comprehensive cloud infrastructure platform. OpenStack is a general-purpose cloud platform used across various industries

while OSM is specific to managing virtualized network functions (VNFs) and software-defined networks (SDNs). OpenStack can be integrated with OSM to leverage its compute, storage, and networking capabilities for NFV and SDN orchestration.

2.5 Data Analytics

Security and privacy are a crucial part of I4.0 and IIoT: They involve the interconnection of numerous devices, networks and systems, which if left unchecked creates a vast attack surface for potential cyber threats. Without robust security measures, these interconnected systems become vulnerable to various threats such as unauthorized access, data breaches, and cyber-attacks. By ensuring security, organizations can protect their valuable data, prevent disruptions to operations, and safeguard against intellectual property theft, thereby maintaining the trust of their stakeholders and avoiding significant financial and reputational damages. Moreover, the secure generation and transmission of vast amounts of sensitive information is a crucial requirement of I4.0. This includes proprietary data, customer details, operational insights, and trade secrets. Protecting the privacy of this information is vital to maintain compliance with data protection regulations, respect customer privacy rights, and avoid potential legal liabilities. Machine / Deep Learning is, one of the enabling technologies of IIOT network security and privacy. The following section presents a secure, automated mechanism for the lifecycle of ML/DL components of Zero Swarm.

2.5.1 Automation mechanism for management of ML pipelines and deployment/update of ML/DL components

In D2.3 “Cyber-security implementation templates and methodological approach” a secure DevOps methodology that will be applied in the project was presented. The following section initially presents a high-level overview of the capabilities offered by GitLab CI/CD (Continuous Integration/Continuous Deployment) pipelines specifically for building, testing, and deploying AI models throughout their development lifecycle, along with a high level ML operations (MLOps) architecture. These pipelines enable the developers to automate the entire process, from training and testing to deployment and monitoring. After the overview, we present a distributed learning framework, using CI/CD pipelines for MLOps that will be utilized in the project.

While there are multiple platforms that offer CI/CD functionalities (e.g., Jenkins, CircleCI, AWS CodeBuild, Azure DevOps), GitLab provides a comprehensive set of features for managing AI models while also being open source [31]. It offers robust version control capabilities, allowing effective code, data, and configuration management. The platform supports automated building and packaging of models, ensuring consistency and reproducibility. CI/CD pipelines enable various testing and validation stages, including unit tests, integration tests, performance tests, and code quality checks. Pipelines can be configured for automatic model training and evaluation tasks, including hyper parameter tuning stages to facilitate finding the best-performing model configurations, utilizing GitLab's computing resources or external infrastructure. GitLab pipelines also streamline model deployment to production servers, cloud platforms, or containerized environments, with support for deployment strategies like blue-green or canary releases. Monitoring and feedback mechanisms can be integrated into the pipelines to continuously evaluate model performance and trigger alerts if needed. Overall, GitLab CI/CD pipelines provide a comprehensive framework for managing the lifecycle of AI models, from development to deployment and maintenance. They enable automation, reproducibility, and

collaboration, helping the developers to streamline and enhance the efficiency of AI development workflows.

Figure 18 presents a generalizable, high-level architecture for a MLOps framework utilizing a CICD pipeline to automate the management, deployment and update of ML components, taking into account both distributed and non-distributed learning models.

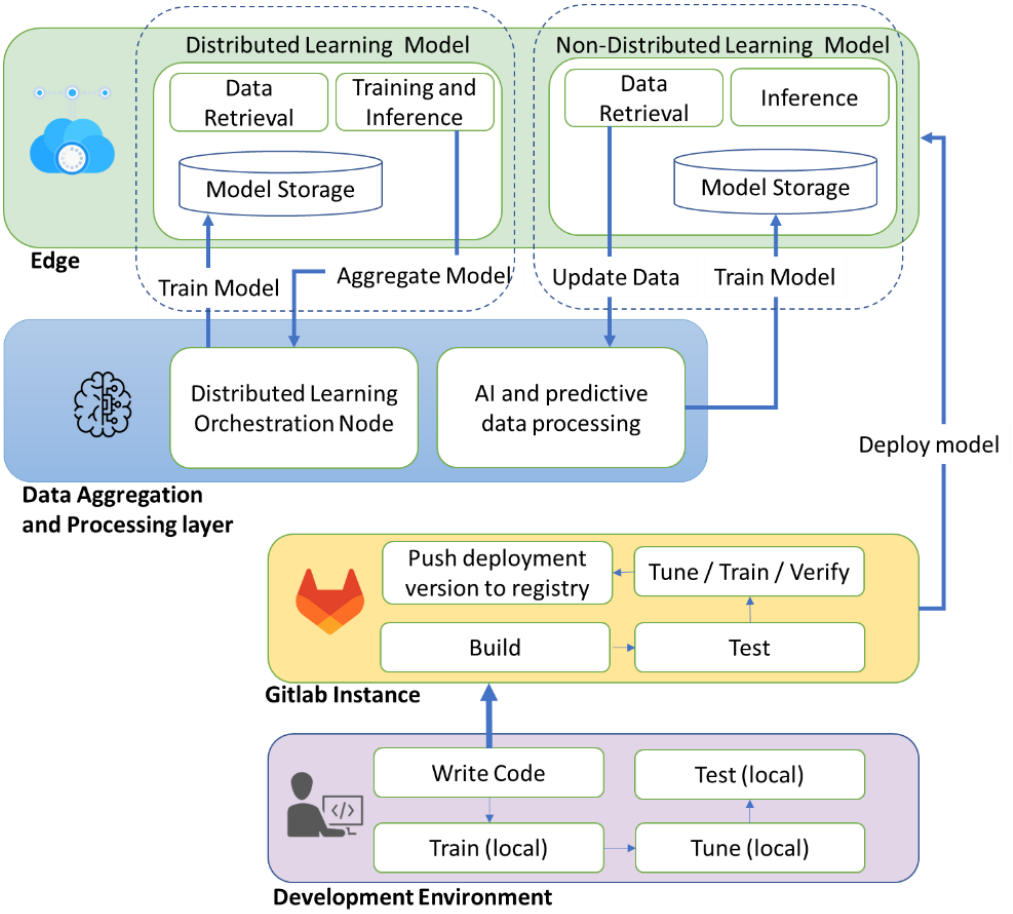


Figure 18: High-level architecture for the MLOps framework

There are different flavours of distributed learning. The one which we discuss in this deliverable considers a central orchestration serve, hereafter referred to as Training Coordinator Node (TCN), which organizes the training, but never receives or gets access to the raw data of other parties involved in the training process. However, the TCN receives the contributions of all other client nodes, hereafter referred to as Edge Contributor Nodes (ECNs). The framework can exploit the data of a single owner which is distributed over a set of ECNs to train a global model or in a multi-ownership paradigm.

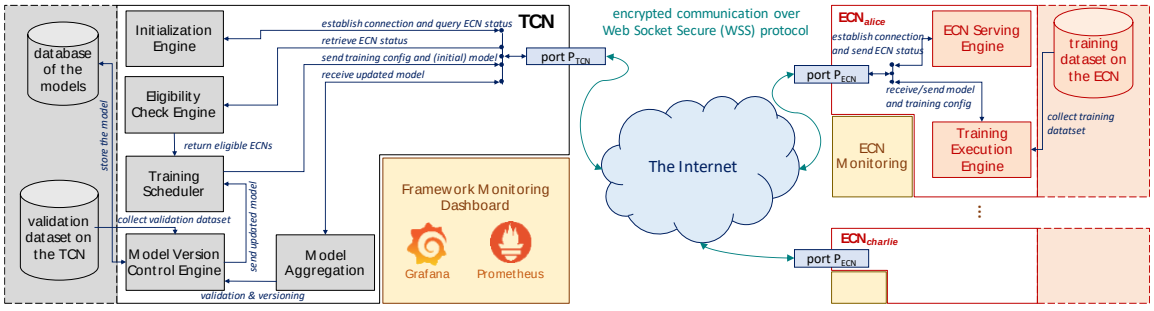


Figure 19: Interaction between different building blocks of the distributed learning framework

The block diagram representation of our proposed architecture is shown in Figure 18. It is composed of two main components which together contribute in the training of a global ML model using data hosted on a set of geo-distributed edge nodes. On one side there is a TCN and on the other side there are several ECNs. The TCN acts as a moderator that manages the overall training procedure. It can be hosted on any edge node assuming that the node owns the required capabilities. ECNs are any of the distributed edge nodes, which host data and are capable of performing training computation.

The TCN comprises different interconnected segments such as Initialization Engine, Eligibility Check Engine, Training Scheduler, Model Version Control Engine, and Model Aggregation. The process on the TCN side starts by initiating a connection to the ECNs and submit a query to be informed about the status of all ECNs. This process is conducted by the Initialization Engine. The ECNs status is then retrieved by the Eligibility Check Engine to be further used for choosing a list of eligible ECNs for the rest of the training process. The ECN status may include information such as the amount of data available for training, and the availability of computational resources. The list of selected ECNs is sent to the Training Scheduler in order to send the model and training configuration to each of the eligible ECNs. All the trained models are received from the ECNs and the model aggregation is performed by the Model Aggregation module to build a global model out of the received locally trained models. This updated global model is now ready for validation which is carried out by the Model Version Control Engine on the validation dataset available on the TCN. The obtained model is later versioned according to the status of the training round. A copy of the obtained global model is also stored in the model database. In addition, there is a Monitoring Dashboard, located on the TCN as a surveillance service to provide observability on the training procedure to the end-user.

The ECN is also composed of different sub-modules, namely, ECN Serving Engine, and Training Execution Engine. The ECN Serving Engine is responsible for binding the ECN entity to a specific port and address, which can be accessible by the TCN through the network connection. The main body of the training lays in the Training Execution Engine. As an example, for the case of supervised training of Neural Networks, after retrieving the global model and training configuration it executes a training loop for a certain number of epochs as predefined in the training configuration. The loop begins by outputting some predictions using the training dataset available on the ECN. Then, the loss function is computed using the corresponding ground-truth. Next the gradients are obtained by means of backpropagation. Finally, the model parameters are updated. Once the loop is over, the locally updated model is sent back to the TCN.

As illustrated in Figure 19, the training workflow comprises four general stages which are carried out in an iterative manner for each training round:

1. Checking the eligibility of the ECNs. Once the eligibility check step is performed, a list of eligible ECNs is returned to contribute to the training round.
2. Distributing the model and training configuration among ECNs.
3. Reporting the locally obtained models to the TCN.
4. Updating the local models with the newly obtained Global Model.

There is an important step that must be taken prior to initiation of any training process, which we call ECN Serving. In this stage, each ECN binds to its specific IP/Port and starts serving the TCN. A JSON file must be provided to the system on each ECN with information such as its corresponding IP/Port, path to the data, and dataset tag. Another JSON file must be provided on the TCN site with information such as the list of all potential ECNs contributing in the training or validation.

During each training round, four main executions are performed (see Figure 20):

1. **Initialization**: TCN is responsible for the initialization of the training. This process comprises establishing the connection with all the potential ECNs and query their status. Prior to this stage, all the ECNs should be available in serving mode.
2. **Scheduling**: The TCN is responsible to schedule the training process considering the number of eligible nodes available to contribute to the training as well as to send the model and configuration files to the eligible ECNs. Scheduling refers to prioritizing the ECNs for the training process based on their time dependent resource availability. In the solution provided here, this condition is relaxed, and scheduling is reduced to listing the ECNs in the training configuration file of the TCN, based on the output of the eligibility check stage. Note, for the TCN to start the training, all eligible ECNs should remain available in serving mode. In case of the unavailability of some of the ECNs, they will be taken out after a specified time-out period.
3. **Training**: Once the ECNs receive the configuration and global model from the TCN, they start the training. After the training is completed, each updated model is sent back to the TCN.
4. **Model Aggregation**: Once the TCN receives the locally updated models, it performs model aggregation (e.g., Federated Averaging) to obtain a new global model.

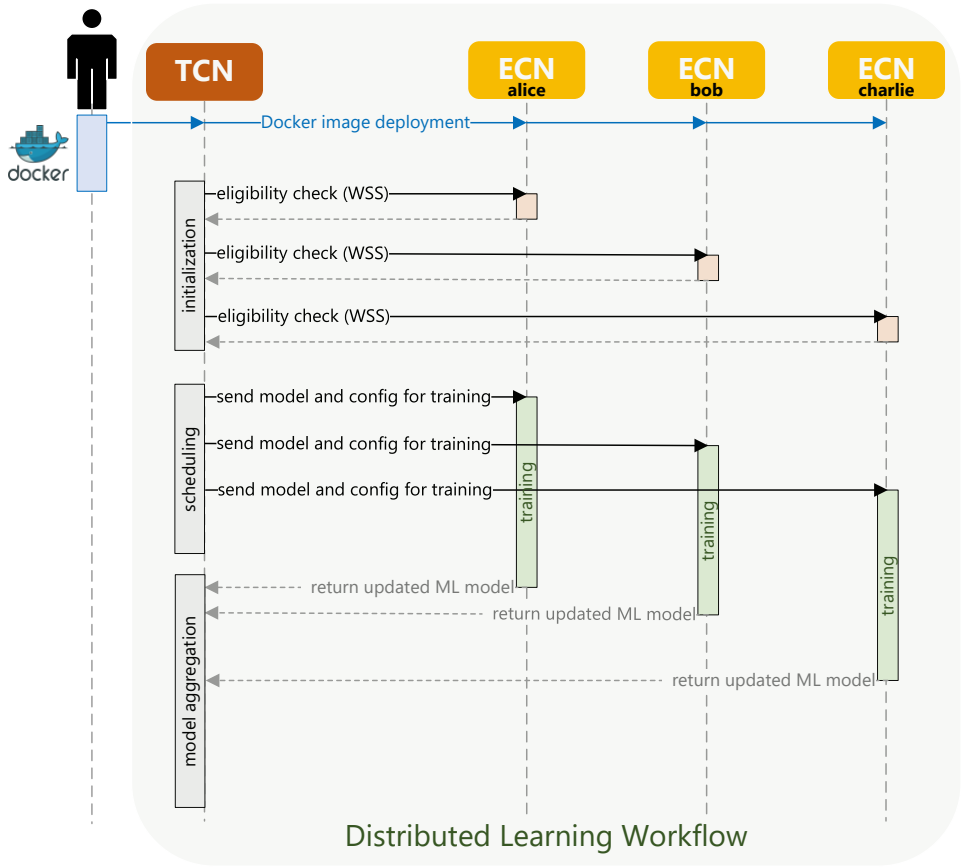
As explained before, prior to starting the training procedure by the TCN, each ECN must be ready to serve the training. ECNs accomplish this mission by binding to their specific IP/Port address. In this stage, the dataset which will be further used for the training is introduced to the serving ECNs with their specific data path and its corresponding tag on the ECN storage disk.

It is possible to divide the ECN processes into two separate phases, namely Eligibility Check phase and Training Phase. In the eligibility check phase, the ECN receives the query from the TCN and then it responds to the TCN with the local metrics. In the training phase, the training notification is received together with the configuration file from the TCN. The ECN substitutes its local model with the global model pulled from the TCN. It further sets the hyper-parameters such as batch size, training epochs, and learning rate in its local computation environment. Then, it starts the training on the locally available labelled data. It utilizes local training by using, e.g., gradient-descent. Once the maximum number of epochs is reached, the updated Local Model is ready to be sent back to TCN. It sends the obtained Local Model or the updated parameters of the Local Model to the TCN.

The TCN side workflow starts with an initialization process and continues to the training phase. For the training process, a maximum number of training rounds T is considered. Initiating the communication protocol with IP/Ports and certificate-keys (for secure connection) and querying the status of the ECNs are the main parts of the initialization process.

Each Round of the training phase comprises a sequence of actions. The TCN checks the eligibility of each ECN. It then receives notifications from ECNs. If enough number K of ECNs are available, then the TCN starts the training round by sending the model configuration files to the eligible ECNs. For this stage, the TCN initializes the model parameters using a new configuration file or restoring an old global model configuration file to proceed with the training. It then waits for the ECNs to complete their training and receives gradients from all eligible nodes. After collecting all the necessary updates from the ECNs, the TCN performs model aggregation to construct a global model out of the collection of locally trained models. The obtained global model is then subject to a validation process on the dataset available on the TCN. This validation, which we call global validation, is possible if some data is available on the TCN site for model assessment, otherwise we need to perform the validation process on each

of the ECNs. If the validation is not successful or the maximum number of rounds *T* is reached, then the training is complete.



TCN: Training Contributor Node, ECN: Edge Contributor Node, WSS: Web Socket Secure

Figure 20: Communication Workflow of different elements of the Distributed Learning Platform using Docker images.

For the TCN and ECNs to communicate with one another, a secure communication protocol will be adopted. The envisioned solution was developed over WebSocket. This protocol enables a two-way communication between a client running a code in a controlled environment to a remote host that has opted-in to communication from that code. The protocol has two parts: a handshake and the data transfer. Once the client and the server have both sent their handshakes, and if the handshake was successful, then the data transfer part starts. This is a two-way communication channel where each side can, independently from the other, send data at will (see Appendix A).

3 Conclusion & next steps

In the Deliverable D4.4, our efforts aimed to create a high-level architecture for the data infrastructure utilized in the Zero-SWARM architecture. The high-level architectures introduced in the preceding sections will serve as a basis for developing software components that will implement both the main data structure of Zero-SWARM and the infrastructure for training machine learning models that will utilize data locality in order to avoid transferring sensitive data over the network.

Moreover, the entire system will be protected by cybersecurity mechanisms aimed at preventing unexpected data leaks and unwanted data sharing.

Additionally, we introduced the paradigm of MLOps to automate and manage the entire lifecycle of a machine learning model, i.e., an automatic paradigm for Machine Learning training.
Finally, we introduced the OPC-UA information model concept that will be used as a base for design and develop the information data models that will be stored and used by the Zero-SWARM ecosystem. Future deliverables as “D4.2 - Distributed stream computing within the Edge-Cloud”, “D4.3 - Self-learning modules for robotic and human behaviours” will detail the information models to be stored and employed in training through the data infrastructure and will describe more in details the training steps in the project.

References

[1]. Zero-Swarm "D2.1 – Eco designed architecture, specification & benchmarking"

[2]. OPC Foundation, *OPC Unified Architecture Part 1: Overview and Concepts,* 2022.

[3]. OPC UA 10000-3: UA Part 3: Address Space Model UA Part 3: Address Space Model (opcfoundation.org)

[4]. ISA-95 Job control OPC UA companion specifications: ISA-95 Common Object Model (opcfoundation.org) and ISA-95-4 Job Control (opcfoundation.org)

[5]. R Dennis Cook. 1977. Detection of influential observation in linear regression. Technometrics 19, 1 (1977), 15–18.

[6]. Amirata Ghorbani and James Zou. 2019. Data shapley: Equitable valuation of data for machine learning. In International Conference on Machine Learning (ICML).

[7]. Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. 2020. Coresets for dataefficient training of machine learning models. In International Conference on Machine Learning (ICML).

[8]. Peilin Zhao and Tong Zhang. 2015. Stochastic optimization with importance sampling for regularized loss minimization. In International Conference on Machine Learning (ICML).

[9]. Xiao Zeng, Ming Yan, and Mi Zhang. 2021. Mercury: Efficient On-Device Distributed DNN Training via Stochastic Importance Sampling. In ACM Conference on Embedded Networked Sensor Systems (Sensys).

[10]. Jaemin Shin, Yuanchun Li, Yunxin Liu, and Sung-Ju Lee. 2022. FedBalancer: data and pace control for efficient federated learning on heterogeneous clients. In ACM International Conference on Mobile Systems, Applications, and Services (MobiSys). 436–449.

[11]. Anran Li, Lan Zhang, Juntao Tan, Yaxuan Qin, Junhao Wang, and Xiang-Yang Li. 2021. Sample-level Data Selection for Federated Learning. In IEEE International Conference on Computer Communications (INFOCOM).

[12]. Gong, C., Zheng, Z., Wu, F., Shao, Y., Li, B., & Chen, G. (2023, April). To Store or Not? Online Data Selection for Federated Learning with Limited Storage. In Proceedings of the ACM Web Conference 2023 (pp. 3044-3055).

[13]. IEC Asset Administration SHell for industrial applications, IEC Standard 63278-1, 2022.

[14]. Plattform Industrie 4.0, Details of the Asset Part 1 - The exchange of information between partners in the value chain of Industrie 4.0 (Version 3.0RC02), 2022.

[15]. Industrial Digital Twin Association, "https://industrialdigitaltwin.org," [Online]. Available: https://industrialdigitaltwin.org/en/content-hub/submodels. [Accessed 13 04 2023].

[16]. OPC Foundation, OPC UA for Asset Administration Shell (AAS), 2021.

[17]. Zero-Swarm, "D2.3. Cybersecurity implementation templates and methodological approach"

[18]. Standardization landscape for cybersecurity https://ecs-org.eu/ecso-uploads/2022/10/5a31129ea8e97.pdf

[19]. ENISA Threat Landscape. https://www.enisa.europa.eu/publications/enisa-threat-landscape-2022

[20]. "NIST-800-53 Rev5.1," [Online]. Available: https://csrc.nist.gov/Projects/risk-management/sp800-53-controls/release-search#/.

[21]. "https://syc-se.iec.ch/deliveries/cybersecurity-guidelines/security-standards-and-best-practices/iec-62443/," [Online].

[22]. K. Cao, Y. Liu, G. Meng and Q. Sun, "An Overview on Edge Computing Research," in *IEEE*

[23]. T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiquzzaman and D. O. Wu, "Edge Computing in Industrial Internet of Things: Architecture, Advances and Challenges," in IEEE Communications Surveys & Tutorials, vol. 22, no. 4, pp. 2462-2488, Fourthquarter 2020, doi: 10.1109/COMST.2020.3009103.

[24]. X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan and X. Chen, "Convergence of Edge Computing and Deep Learning: A Comprehensive Survey," in IEEE Communications Surveys & Tutorials, vol. 22, no. 2, pp. 869-904, Secondquarter 2020, doi: 10.1109/COMST.2020.2970550.

[25]. W. Sun, S. Lei, L. Wang, Z. Liu and Y. Zhang, "Adaptive Federated Learning and Digital Twin for Industrial Internet of Things," in IEEE Transactions on Industrial Informatics, vol. 17, no. 8, pp. 5605-5614, Aug. 2021, doi: 10.1109/TII.2020.3034674.

[26]. OpenStack Foundation, "Accelerating NFV Delivery with OpenStack: Global Telecoms Align Around Open Source Networking Future", White Paper, 2016

[27]. M Yannuzzi, et al., "Toward a converged openfog and etsi mano architecture," in 2017 IEEE Fog World Congress (FWC), IEEE, 2017.

[28]. OpenStack Foundation. (2021). OpenStack. [Online]. Available: https://www.openstack.org/

[29]. OSM (Open Source MANO). (2021). [Online]. Available: https://osm.etsi.org/

[30]. ETSI NFV, "NFV Release 4 Description", ETSI, 2020

[31]. GitLab CI/CD Documentation, [Online]. Available: https://docs.gitlab.com/ee/ci/" .

[32]. Z. Liu e P. Bellot, "OPC UA PubSub Implementation and Configuration". em 2019 6th International Conference on Systems and Informatics (ICSAI), 2019 pp. 1063-1068. DOI: 10.1109/ICSAI48974.2019.9010445

[33]. *Access*, vol. 8, pp. 85714-85728, 2020, doi: 10.1109/ACCESS.2020.2991734.

[34]. AutomationML OPC UA companion specification: AutomationML (opcfoundation.org)

[35]. Tobacco Machinery OPC UA companion specification: Tobacco Machinery (opcfoundation.org)

Appendix A

The WebSocket specification defines an API establishing socket connections between a web browser/client and a server such that there is a persistent connection between the client and the server and both parties can start sending data at any time.

The WebSocket protocol has lower overhead than the half-duplex alternatives such as HTTP polling, facilitating real-time data transfer from and to the server. This is made possible by providing a standardized way for the server to send content to the client without being first requested by the client, and allowing messages to be passed back and forth while keeping the connection open.

WebSocket is a framed protocol which means that a chunk of data is divided into a number of discrete chunks, with the size of the chunk encoded in the frame. The frame includes a frame type, a payload length, and a data portion. Once both parties acknowledge that the WebSocket connection should be closed, the TCP connection is torn down.

The WebSocket protocol is an independent TCP-based protocol. However, its only relationship to HTTP is that its handshake is interpreted by HTTP servers as an Upgrade request. WebSocket uses port 80 for regular WebSocket connections and port 443 for WebSocket connections tunnelled over Transport Layer Security (TLS).

Appendix B

RPK high level modelling structure that will be described in the Deliverable D4.2 and D4.3