D2.1 Reference system architecture and validation planning of the implementation scenarios

1st version

Revision: v.1.0

Work package	**WP 2**
Task	Task 2.1
Due date	28/02/2025
Submission date	09/04/2025
Deliverable lead	FIWARE Foundation e.V
Version	1.0
Authors	Sofia Polymeni, Vasileios Pitsiavas, Dimitrios Gerakas, Christos Zavitsanos, and Georgios Spanos (CERTH), Fernando López, José I. Carretero, and David Campo (FIWARE), Andreas Ekelhart and Walid Fdhila (SBA), Stathis Zaragkas (NCSRD), Axel Vick and Isabella Grossart (FhG), Sergio Lembo (VTT), Ioannis Kefaloukos and Stamatis Kostopoulos (HMU), Francesco De Pellegrini, Cleque Mboulou, and Younes Ben Mazziane (UAVIGN), Charalampos Marantos (INTRA), Angel Cataron (SIEMENS), Georgios Koutroulis (AVL), Dionysis

Grant Agreement No.: 101135930
Call: HORIZON-CL4-2023-DATA-01.
Topic: HORIZON-CL4-2023-DATA-01-04
Type of action: HORIZON Research and Innovation Actions

	Skordoulis and Shengli Liu (AXON), Matej Posinković and Matjaž Breskvar (BEYOND), Savvas Ouzounidis, Marios Siganos, Zacharenia Lekka, Akis Nousias, Nikos Moschos, Vasileios Gavresis, and Ioannis Boukas (K3Y), Giannis Ledakis (UBI), Luciano Riccio, Enzo Caputo, and Sara Dana Kabl Talabani (MEDITECH), Han Yang and Qiang Ni (ULANCS)
Reviewers	Isabella Grossart (FHG) Georgios Spanos (CERTH)
Abstract	First version of CoGNETs architecture, including a description of the technologies to be used, the partners capabilities and the KPIs specifications in respect to the topologies and business models of each targeted PUC.
Keywords	Cloud-Edge-IoT Continuum, DNN, RNN, FIWARE, Game optimization strategies, Cognitive computing and programming models, Federated learning mechanisms, Swarm-wise distributed security paradigms, Data manageability, scalability and adaptability mechanisms

Document Revision History

Version	Date	Description of change	List of contributor(s)
0.1	29/11/2024	• ToC + 1st edit	FIWARE
0.2	23/02/2025	• Contribution to sections 1, 3, 5, 7 • Integration of contributions, edition, and generation new version	CERTH, MEDITECH, FIWARE, K3Y, VTT, INTRA, UAVIGN, HMU, ULANCS, SBA
0.3	27/02/2025	• Contribution to sections 1, 2, 3, 5, 6, 7 • Integration of contributions	FIWARE, CERTH, ULANCS, MEDITECH, FhG, SIEMENS, INTRA, SBA, UBI, BEYOND, K3Y
0.4	11/03/2025	• Contributions to sections 3, 4, 5 • Integration of PUC1 and PUC2 • Integration of contributions	FIWARE, FhG, ULANCS, VTT, BEYOND, SIEMENS, AVL, UAVIGN, AXON, SBA
0.5	25/03/2025	• Contribution to sections 3, 5, 6, 7, 8, 9 • Integration of PUC3 • Integration of changes • Final formatting and edition	FIWARE, K3Y, FhG, HMU, ULANCS, MEDITECH, CERTH
0.91	06/04/2025	• 1st review	FhG
0.92	07/04/2025	• 2nd review	CERTH
1.0	09/04/2025	• Resolution of reviewers' comments • Final Edition	FIWARE, FhG, CERTH, INTRA

Funded by the European Union

Project funded by

DISCLAIMER

Funded by the European Union

Project funded by

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Swiss Confederation

Federal Department of Economic Affairs, Education and Research EAER
State Secretariat for Education, Research and Innovation SERI

Funded by the European Union (CoGNETs, 101135930). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

This work has received funding from the Swiss State Secretariat for Education, Research and Innovation (SERI).

COPYRIGHT NOTICE

Project funded by the European Commission in the Horizon Europe Programme		
Nature of the deliverable:	**R**	
Dissemination Level		
PU	*Public, fully open, e.g. web (Deliverables flagged as public will be automatically published in CORDIS project's page)*	✓
SEN	*Sensitive, limited under the conditions of the Grant Agreement*	
Classified R-UE / EU-R	*EU RESTRICTED under the Commission Decision No2015/ 444*	
Classified C-UE / EU-C	*EU CONFIDENTIAL under the Commission Decision No2015/ 444*	
Classified S-UE / EU-S	*EU SECRET under the Commission Decision No2015/ 444*	

* R: Document, report (excluding the periodic and final reports)
DEM: Demonstrator, pilot, prototype, plan designs
DEC: Websites, patents filing, press & media actions, videos, etc.
DATA: Data sets, microdata, etc.
DMP: Data management plan
ETHICS: Deliverables related to ethics issues.
SECURITY: Deliverables related to security issues
OTHER: Software, technical diagram, algorithms, models, etc.

EXECUTIVE SUMMARY

This document, titled "D2.1 – Reference system architecture and validation planing of the implementation scenarios" presents an in-depth exploration of the CoGNETs project, focusing on the technologies and tools to architect the CoGNETs system, including KPIs specifications in respect to the topologies and business models of each targeted Pilot Use Cases. This document will be used as a reference by all technical tasks and WP3, WP4, and WP5.

Key aspects include:

- **Project Overview:** CoGNETs aims to advance the Cloud-Edge-IoT (CEI) continuum through innovative technologies and frameworks.
- **Architecture Definition:** The document outlines the methodology for defining the architecture, including stakeholder identification, requirement gathering from partners, view creation, and model development.
- **Technology Considerations:** It explores the infrastructural and technological capacity of partners, the state-of-the-art in relevant areas (like IoT-Edge-Cloud, Game Optimization, Federated Learning, and Security), and expert analysis.
- **Pilot Use Cases (PUCs):** The document defines the PUCs and their specific requirements, the input and output as well as the initial identification of the data to be managed, which will drive the architecture's development. Additionally, it is defined the corresponding KPIs to measure the success of the platform.
- **Key Components:** Several key components are identified, including the Dashboard (UI), Cognitive AI Service Repository (CSR), DevSecOps Platform (DOP), Middleware Swarm Context, and Middleware Node Context.
- **Security:** Security is a major concern, with considerations for hardware, system, application, and AI-level security.

In conclusion, the goal of the document is to provide a comprehensive understanding of the CoGNETs logical building blocks, its requirements and KPIs laying the groundfloor for the successful implementation of the CoGNETs platform.

Funded by the European Union

Project funded by

TABLE OF CONTENTS

Funded by the European Union

Project funded by

Funded by the European Union

Project funded by

Funded by the European Union

Project funded by

Funded by the European Union

Project funded by

Funded by the European Union

Project funded by

LIST OF FIGURES

Funded by the European Union

Project funded by

Funded by the European Union

Project funded by

LIST OF TABLES

Funded by the European Union

Project funded by

Funded by the European Union

Project funded by

Funded by the European Union

Project funded by

ABBREVIATIONS

ACO	Ant Colony Optimization
AI	Artificial Intelligence
AIOTI	Alliance for Internet of Things Innovation
API	Application Programming Interface
AWS	Amazon Web Services
BDVA	Big Data Value Association
BTC	Business Technical Requirements
BUS	Business Requirements
CEI	Cloud-Edge-IoT
CFL	Collaborative Federated Learning
CNN	Convolutional Neural Network
CPS	Cyber-Physical Systems
CPSL	Cluster-based Parallel Split Learning
CSA	Coordination and Support Action
CSR	Cognitive AI Service Repository
DAG	Direct Acyclic Graph
DDAG	Distributed Directed Acyclic Graph
DDM	Distributed Data Manager
DID	Decentralized Identifier
DLT	Distributed Ledger Technology
DMF	Data Management Framework
DNN	Deep Neural Network
DOP	DevOps Platform
DRM	Distributed Resource Manager
DSM	Distributed Service Manager
DWM	Distributed Workload Manager
ESCO	Enhanced Chicken Swarm Optimisation

Funded by the European Union
Project funded by

ETSI	European Telecommunications Standards Institute
FNC	Functional Requirements
FPGA	Field Programmable Gate Array
GDPR	General Data Protection Regulation
GHG	Greenhouse Gas
HW	Hardware Platform
IoT	Internet of Things
JADE	Java Agent Development Framework
JITA-4DS	Just-in-Time Architecture for Data Science Pipelines
KPI	Key Performance Indicator
KRR	Knowledge Representation and Reasoning
MAS	Multi-Agent Systems
MEC	Multi-access Edge Computing
MEL	MicroELements
MFA	Multi-Factor Authentication
MILP	Mixed-Integer Linear Programming
ML	Machine Learning
NCE	Node Manager - Component Executor
NDM	Node Manager - Data Manager
NDMo	Node Manager - Device Monitoring
NDR	Node Manager - Device Registration
NDS	Node Manager - Device Storage
NFN	Non-Functional Requirements
NGA	Node AI Game Agent
NGSI	Next Generation Service Interfaces
NN	Neural Network
NWO	Node Manager – Workload Orchestrator
OS	Operating System

Funded by the European Union
Project funded by

PN	Production Need
PUC	Pilot Use Case
PUF	Physical Unclonable Functions
QoS	Quality of service
RBAC	Role-Based Access Control
RISC	Reduced Instruction Set Computer
RL	Reinforcement Learning
ROI	Return on Investment
ROS	Robot Operating System
RvE	Robustness via Elasticity
SAREF	Semantic Annotation for Real-world Entities Framework
S-AI	AI-Level Security
S-APL	Application-level Security
SDN	Software Defined Network
SFL	Split-Federated Learning
SGA	Swarm AI Game Agent
SGA	Swarm AI Game Agent
S-HW	Hardware-level Security
SLA	Service Level Agreement
SLO	Service Level Objective
SRPMA	Secure Routing Protocol based on Multi-objective Ant colony optimization
S-SL	System-level Security
SSO	Single Sign-On
TLS	Transport Layer Security
TMS	Thermal Management System
TRL	Technology Readiness Level
UI	User Interface
VDC	Virtual Data Center

Funded by the European Union
Project funded by

WSNs	Wireless Sensor Networks

Funded by the European Union

Project funded by

1 INTRODUCTION

The rise of edge computing and the movement of processing intelligence closer to end users have made large-scale private edge deployments more accessible. However, realizing the full potential of edge intelligence requires more than just boosting edge capacity with commodity and specialized processors near various Internet of Things (IoT) devices. It is needed the creation of a Middleware Framework that enables IoT, Edge, and Cloud devices to autonomously organize dynamic IoT-to-Cloud swarm continuum for optimal data processing and seamless service delivery.

CoGNETS aims to facilitate IoT-to-Cloud swarm continuum through an on-demand opportunistic approach, focusing on "dynamic swarm continuum" that incorporates both fixed and temporary infrastructural elements (such as IoT and Edge devices that can join or exit the swarm in real-time). This approach allows cognitive resources to be organized and utilized as locally as possible, ensuring optimal and secure management of services and data in self-managed heterogeneous environments. These environments may include legacy enterprise networks, resource-constrained IoT systems, and Edge-Cloud devices with varying connectivity types and data volumes.

Additionally, current AI technologies can only partially realize the potential of an autonomous and dynamic computing paradigm. While there are technologies to deploy AI on IoT and resource-constrained devices, these devices still lack capabilities for "self-organization" and "collaborative learning." (e.g., unable to autonomously recognize or respond to the constantly changing availability of data and computing resources). CoGNETs will develop a context-aware mesh network of cognitive resources managed by a Middleware that facilitates the formation of dynamic swarms and coordinates their activities. As a result, it will enable a robust set of end-to-end security measures, decentralized identities, and collaborative federated learning practices, ensuring efficient, transparent, and reliable service provisioning.

1.1 DELIVERABLE PURPOSE & OBJECTIVES

This document comprises the CoGNETs deliverable 2.1 (D2.1) and aims to provide an initial definition of the CoGNETS architecture, including the requirements. To do so, the document defines the Pilot Use Cases (PUCs), the necessary hardware and software components to develop them, their requirements from the CoGNETs platform, as well as the KPIs and their corresponding way of measurement. These PUCs will help to evaluate the relevance of the CoGNETs platform. Moreover, the data to facilitate the interoperability of the information between the different PUCs are identified. Finally, the document defines for each of the PUCs the corresponding KPIs and the corresponding way of measurement of them to evaluate the improvement of the adoption of the CoGNETs solution.

1.2 MAPPING COGNETS TASKS

The purpose of this section is to map CoGNETs Grand Agreement commitments, both within the formal deliverable and Task description, against the project's respective outputs and work performed.

Funded by the European Union

Project funded by

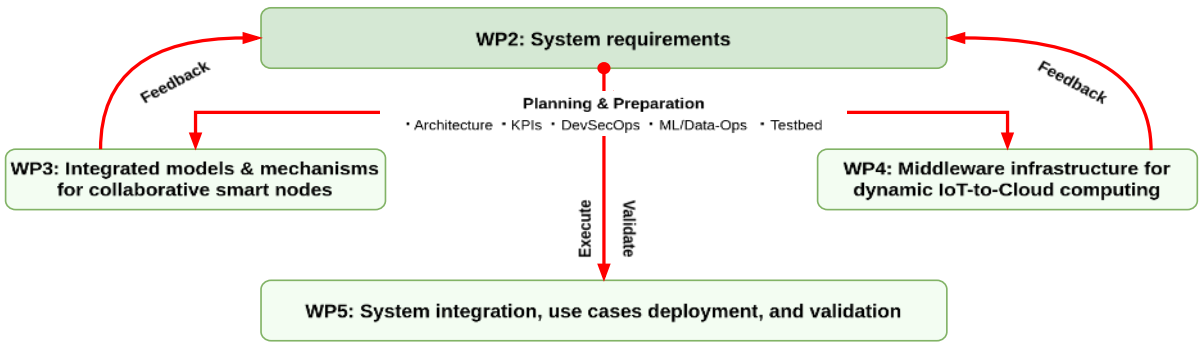
Table 1: Adherence to CoGNETs tasks descriptions.

Project GA Component Title	Project GA Component Outline	Chapter(s)
Task 2.1: Identification of technologies and reference architecture	This task aims to compile recent literature insights to develop a feasible architecture based on the proposed system structures. This includes the definition of the technological capabilities of the partners as well an exploring advancements in IoT-Edge-Cloud swarm continuum architectures, optimization strategies, cognitive computing models, federated learning mechanisms, distributed security paradigms, and data management techniques. The research will include desk studies, stakeholder interviews, and market analysis to enhance the architecture with effective methods for improving computing efficiency, AI accuracy, and updating outdated modules.	Sections 2, 3, 4, 5
Task 2.2: Specification of use case scenarios, KPIs, and business models	This task defines the Pilot Use Cases (PUCs), KPIs, and business models for three vertical sectors, evaluating the participants, requirements, actor interactions, and expected outcomes for each PUC, while considering stakeholders' facilities, production needs, data privacy regulations, and security policies.	Section 6

1.3 INTERRELATIONS WITH OTHER WORK PACKAGES

This document is the basis for the implementation of the Middleware Layer - Swarm Context, Middleware Layer - Node Context, Middleware Layer - Security, and Application Layer to be developed in WP3 and WP4 as well the definition of the PUCs to be executed in WP5. For this purpose, D2.1 includes the initial definition of the requirements, hardware and software components of the three PUCs to be used in the context WP5.

Figure 1: Relationship with other work packages



Funded by the European Union
Project funded by

Therefore, D2.1 will provide valuable inputs to subsequent deliverables, including:

- **D2.2** will define the DevSecOps and ML/Data- Ops methodology specification based on the tools and requirements introduced in the Application definition of the CoGNETs platform.
- **D2.3** will define the Lab-based testbed deployment with the requirements defined in this document.
- **D3.1** will be focused in the definition of the first version of the integrated software mechanisms for collaborative smart nodes taking into account the requirements of the Game Agents and architecture definition in this document.
- **D3.3** will be focused in the definition of the first version of the integrated software mechanisms for collaborative smart nodes in terms of Collaborative Federated Learning and the end-to-end security based on the requirements and architecture defined in this deliverable.
- **D4.1** will specify the first version of the implementation of the Logical Building Blocks related to the middleware infrastructure for dynamic IoT-to-Cloud swarm computing related to Broker Runtime and functional components, including the corresponding APIs and aligned with the requirements identified in this deliverable.
- **D4.3** will specify the first version of the implementation of the Logical Building Blocks related to the middleware infrastructure for dynamic IoT-to-Cloud swarm computing related to external tools, AI services and Datasets generated in the project.
- **D5.1** will be focused on the data management, system integration, evaluation plan and more details definition of the KPIs in terms of defining and obtaining the metrics of them based on the definition of the PUCs described in this document.
- **D5.2** will be focused on the Pilot Use Case deployment and implementation and how the CoGNETs Building Blocks are use on them.

1.4 DELIVERABLE STRUCTURE

This document is organised as follows:

- **Section 1** defines the introduction of the content developed in the document including the relationships with the tasks inside the WP2 and the relationships with other work packages.
- **Section 2** defines the architecture's definition methodology adopted in the WP2 to recover the requirements and define the Pilot Use Cases (PUCs). This activity includes the identification of the Logical Building Blocks to be introduced in the architecture.
- **Section 3** defines an overview of the hardware capacities of the partners to be considered in the execution of the CoGNETs platform as well as proper scalability of the solution to be implemented.
- **Section 4** defines a deep analysis of the state of the art regarding Cognitive computing & programming models, Data manageability, scalability and adaptability mechanisms,

Funded by the European Union

Project funded by

Federated Learning mechanisms, Game Optimization strategies, IoT-Edge-Cloud swarm continuum architectures, Swarm-wise distributed security paradigms.

- **Section 5** is focussed on the interviews with experts in several areas related to the project to extract requirements to be considered as well in the implementation to the CoGNETS platform.
- **Section 6** defines the project motivation, challenges and objectives to be achieved, the initial introduction of the CoGNETS Logical Building Blocks, the main technologies and tools to architect CoGNETs as well as the architecture views of the project, defining the high level overview of the Logical Building Blocks.
- **Section 7** defines the requirements list for each Logical Building Block generated by the CoGNETs partners taking into account Technical, Non-Technical, Business, and Business Technical requirements.
- **Section 8** defines the PUCs in terms of the solution to be adopted as well as the requirements that need to be implemented by the CoGNETs architecture to resolve the improvement based on distributed AI Models. This section will include also the definition of KPIs in order to measurements the success of the adoption of the CoGNETs solution.
- **Section 9** concludes this document and discusses the next steps in the project development process.

Funded by the European Union

Project funded by

2 ARCHITECTURE'S DEFINITION METHODOLOGY

This section presents the methodology used to define the CoGNETs architecture. A systematic approach has been followed, based on the ISO/IEC/IEEE 42010 Standard. This standard, first published in 2011, provides guidelines for defining the essential elements and considerations when defining an architecture.

The methodology used in defining the CoGNETs architecture involved several critical steps. Stakeholders were identified and the tools and pilot use-cases were defined by understanding the interconnection technologies. Technical details and user requirements were specified. Another important step towards defining architecture included matching the project tools with architecture components and establishing the gathered requirements in the Pilot Use Cases.

The initial architecture vision included high-level diagrams and component descriptions. The integration and implementation pipeline was planned to detail component integration and testing, considering hardware requirements and connectivity needs. The final architecture design incorporated feedback and validation activities, ensuring complete documentation.

The following paragraphs will provide details on the methodology overview, architecture definition phases and key activities.

2.1 METHODOLOGY OVERVIEW

Before defining the architecture, the scope and objectives of the CoGNETs solution were thoroughly discussed and clarified in the early stages of the project. The defined architecture aims to provide a comprehensive implementation of complex IoT (Edge to Cloud) systems that fully supports the research objectives of CoGNETs. The scope of the defined architecture includes defining all processes, data management protocols, component interactions, and the technology infrastructure.

From the beginning of the project, the key stakeholders (partners), including the researchers, the industry partners, and the end-users (PUCs), were actively engaged. Regular biweekly meetings were held to ensure that their concerns were addressed and the user and technical needs and requirements were successfully collected. Business, Functional, Non-Functional, and Technical requirements were collected through discussions and interviews. These requirements were then validated by all partners to ensure they accurately reflected their needs and were included in shared documents within the project's repository.

2.2 ARCHITECTURE DEFINITION PHASES AND ACTIVITIES

The architecture definition process for the CoGNETs project was divided into several phases and activities, as presented in this paragraph. Initially, stakeholders and their concerns related to the CoGNETs solution were identified during the Stakeholder Identification and Concern Identification phase. Following this phase, requirements were gathered and validated in the Requirement Gathering and Validation phase to ensure alignment with the CoGNETs objectives. During this phase, several activities ran simultaneously. Various views were developed to represent different aspects in the Architecture View Creation phase. Detailed definition of the components along with their interactions were created during the Architecture Model Development phase. Finally, the CoGNETs architecture is produced in the Architecture Description and Evaluation phase to ensure that it meets the established requirements.

Funded by the European Union
Project funded by

1. **Stakeholder Identification and Concern Identification:**

This phase involves identifying stakeholders and their concerns, as well as defining the scope and objectives of the architecture. During this phase, the key tools, processes and structures were defined/outlined and the PUCs were discussed to ensure that each component and sub-system supports the research activities and aligns with the overall objectives of CoGNETs.

The initial step included assessing the CoGNETs solution's offerings within distributed IoT infrastructures deployed into the Edge-Cloud Continuum environment. The interconnected technologies and their application to PUCs were taken into account. Once the CoGNETs' role in the Edge-Cloud Continuum was clarified, several activities (2nd phase) proceeded simultaneously.

2. Requirement Gathering and Validation:

This phase focuses on collecting and validating requirements from stakeholders and choosing viewpoints to address these requirements. To facilitate this process, shared documents and questionnaires were created to list all the requirements that the architecture must meet. Comprehensive discussions and interviews took place to ensure alignment with PUCs needs and project objectives.

Key stakeholders were actively engaged (academic partners, industrial partners, end-users) and their primary concerns were documented.

The list of requirements includes seven related topics:

a) IoT-Edge-Cloud swarm continuum architectures

b) Game optimization strategies

c) Cognitive computing & programming models

d) Federated Learning mechanisms

e) Swarm-wise distributed security paradigms

f) Data manageability

g) Scalability and adaptability mechanisms

The key activities carried out simultaneously during this phase are the following:

a) **Middleware Operations**: This activity focuses on gathering requirements and defining the interfacing characteristics for Middleware Operations including the Swarm Context and the Node Context with the main purpose to establish the network plane execution and data plane synchronization. This activity included also the requested requirements regarding the performance to be achieved by the platform and restrictions that might be requested by the different PUCs. This activity covered the services included in the Middleware Layer (Swarm Context and Node Context).

Funded by the European Union

Project funded by

b) **Game Intelligent Agent Requirements**: Another activity establishes the corresponding requirements and definition of the DNNs to be executed in the Pilot Use Cases associated to the Game Intelligent Agent. The purpose was to extract the corresponding requirements in terms of precise algorithms to use and which kind of information is required in order to develop the pruning and splitting process as well as the communication between the different nodes or leaf of the graph. The services involved are the following:

- **Middleware Layer (Swarm Context) - Swarm AI Game Agent (SGA)**.
- **Middleware Layer (Node Context) - Node AI Game Agent (NGA)**.
- **Middleware Layer (Security Context)**.
- **Application Layer - Cognitive AI Service Repository (CSR)**.

The next activity covers the security configuration of the Node Context with the purpose to connect to the Edge-Cloud continuum and provide secure communications.

c) **Data Stamping and Security**: This activity included the corresponding data stamp of the partial results of the splitting DNN and RNN. Besides, a complete security layer is provided in terms of hardware and system level security as well AI level security. The services involved are the following: Node Manager - Device Registration (NDR), Node Manager - Device Storage (NDS), Node Manager - Data Manager (NDM), Hardware-level Security (S-HW), System-level Security (S-SL), Application-level Security (S-APL), AI-level Security (S-AI). Another activity covered the process to provide a UI access to the user in order to facilitate the Human-in-the-loop activity to select the corresponding management of the PUCs process and the selection of the corresponding Cognitive AI Service from the repository (CSR). These requirements include the corresponding data model of the information requested by the DNN and RNN to be translated to the Edge-Cloud Continuum to be executed.

d) **Physical Layer Requirements**: The last activity was related to the Physical Layer in order to get requirements from the concrete hardware used in the project and which type of connectivity do we have in order to access the information needed to execute the DNN and RNN. These requirements are expected to give details about KPIs, constraints, computational capacities of devices, and network connectivity of the solution.

3. Architecture View Creation:

This phase involves developing views to represent the architecture and defining integration points and interfaces. In this phase, the Consortium was focused on identifying software/ hardware components from the related tasks in CoGNETs' DoA and their interactions. Integration points and interfaces are defined to ensure seamless interaction between different sub-components. In addition, data models, storage and security requirements were established.

Specific viewpoints were defined and developed to address different aspects of the CoGNETs architecture, such as the research, the technical, and the innovation viewpoint. This process provided a high-level overview of the architecture, outlining its alignment with project goals and the overall vision of CoGNETs for collecting feedback from all the partners.

The CoGNETs components (presented in the rest of this document), along with the corresponding tasks are presented in Table 2.

Funded by the European Union

Project funded by

Table 2: CoGNETs components and corresponding Tasks.

Component	Task involved
Dashboard (UI)	T4.4
Cognitive AI Service Repository (CSR)	T4.1, T4.4, T3.3
DevSecOps Platform (DOP)	T2.3, T2.4, T5.1
Swarm AI Game Agent (SGA)	T3.2, T4.2
Distributed Resource Manager (DRM)	T4.2
Distributed Service Manager (DSM)	T4.2
Distributed Workload Manager (DWM)	T4.2, T3.3
Distributed Data Manager (DDM)	T4.2, T3.3
Node AI Game Agent (NGA)	T4.3, T3.2, T3.3
Node Manager - Device Monitoring (NDMo)	T4.3, T3.2, T3.1
Node Manager - Device Registration (NDR)	T4.3, T3.2
Node Manager - Device Storage (NDS)	T4.3, T3.2
Node Manager - Data Manager (NDM)	T4.1, T4.3, T3.2
Node Manager - Workload Orchestrator (NWO)	T4.3, T3.2
Node Manager - Component Executor (NCE)	T4.3
Hardware-level Security (S-HW)	T3.4
System-level Security (S-SL)	T3.4, T4.3
Application-level Security (S-APL)	T3.4, T4.4
AI-level Security – Adversarial shield (S-AI)	T3.4

4. Architecture Model Development:

This phase (which is more related with T2.3 and T2.4) includes the procedure for implementing the architecture, providing a pipeline for the Continuous Integration and Continuous Deployment (CI/CD). This process aims to set the ground for specifying also the infrastructure, supporting the CoGNETs' needs.

5. Architecture Description and Evaluation:

This phase involves documenting the CoGNETs architecture, including the detailed models, diagrams, and descriptions, providing a comprehensive view of the CoGNETs architecture.

Funded by the European Union
Project funded by

This definition methodology ensured that the architecture was consistent and aligned with both the research and innovation objectives of the project. The architecture, delivered in D2.1, provides a strong base for the future development and validation activities.

2.3 KEY EVENTS

The key events that significantly contributed to the definition of the architecture are as follows:

1) **Online Kickoff Meeting (M1)**

   The project was initiated with an online kickoff meeting, where representatives from all partner organizations were present and preliminary discussions were held. In this meeting an initial alignment on **CoGNETs** goals was discussed.

2) **1st Plenary Meeting (M2)**

   The first plenary meeting provided an opportunity for face-to-face discussions of the initial architecture concepts. Technical partners presented their components, and initial approaches and concepts were discussed.

3) **Biweekly WP Meetings**

   Regular biweekly WP2 meetings were established to ensure continuous communication and alignment between all the partners. In these meetings the progress of each task was shared, the challenges were discussed and collective decisions were made. The goal of these meetings was to facilitate ongoing collaboration, issue resolution, challenges mitigation and to understand the project's individual components and the connection between them.

4) **Shared Requirements Document Creation**

   A collaborative document was established and shared to the partners (both technology and PUCs providers) to systematically collect requirements. This living document played the role of the central repository/documentation, where all business, functional, non-functional requirements and technical specifications were collected.

5) **Questionnaires and Interviews**

   Questionnaires were shared to PUC providers and experts, along with detailed interviews. This procedure facilitated and enabled the collection of specific requirements, the investigation of technical constraints and the clarification of PUC specifications leading to detailed and validated requirements, constraints and specifications.

6) **All-Day Architecture Meeting (M7)**

   A dedicated meeting was held to dive deep into the **CoGNETs'** architecture technical details. The meeting focused on resolving complex architectural challenges, clarifying the role of each component, connecting components and corresponding tasks, defining the interfaces between components and establishing the integration strategy.

7) **Discussions for finalizing Sub-Components' Role**

Funded by the European Union

Project funded by

Focused (bilateral) meetings were held to finalize the roles and responsibilities of some of the sub-components within the **CoGNETs** architecture. The purpose of these discussions was to ensure clear boundaries, define the interfaces between components and finalize the roles and responsibilities for sub-components.

8) **2nd Plenary Meeting (M9)**

The final architecture was presented and validated during the **CoGNETs** 2nd Plenary Meeting, while the organization and the final inputs to the present Deliverable were presented. Partners' feedback was collected and final adjustments were made before submission. The **CoGNETs** architecture was finally validated and approved by the partners.

9) **Architecture Submission (as defined in the GA)**

The official submission of D2.1.

Funded by the European Union

Project funded by

3 INFRASTRUCTURAL AND TECHNOLOGICAL CAPACITY OF PARTNERS

The section 3.1, is oriented to the hardware/software capacity in terms of identifying where the platform can be run. The idea is to identify the different platforms that exist from the different partners in order to install or deploy the components. The final idea is to identify where these components will be running and therefore which kind of requirements are needed in order to use those platforms (e.g., connectivity, security access, HW restrictions if exists, etc.).

3.1 CERTH

Within CoGNETs, CERTH brings extensive expertise and advanced infrastructure to support the deployment and execution of the proposed platform. As a research and technology organization, CERTH provides strong software capabilities catered to the development needs requested for both the CoGNETs middleware and its vertical demonstrators.

CERTH will be actively engaged in advancing adversarial AI mechanisms to enhance the security and robustness of all AI-based applications within the CoGNETs platform, ensuring reliability and robustness under adversarial conditions. Its existing research in adversarial security mechanisms, including adversarial training and secure Collaborative Federated Learning (CFL) techniques, will ensure that the deployed AI models within the CoGNETs platform are resilient against data poisoning and adversarial attacks, ultimately enhancing the platform's ability to maintain trustworthiness in distributed AI applications.

By focusing on AI-based security measures, CFL and real-time processing, CERTH's role is equally important in the development of the PUC3 – "Connected Healthcare (Collaborative AI for Medical Data Analytics in Health 4.0)" scenario, offering both software expertise in Edge computing, IoT integration and an existing advanced health agent, as well as hardware resources including Edge (e.g., Raspberry Pi 4) and IoT wearable devices (e.g., Samsung Galaxy Watch series 4). The provided infrastructure is designed to support the dynamic and scalable nature of the CoGNETs platform, ensuring efficient deployment across diverse environments. The available computational resources facilitate large-scale AI model training and adaptive analytics that enhance near real-time decision-making processes using secure wireless connectivity. To ensure data privacy and system resilience, advanced security measures are also integrated, including end-to-end encryption, multi-factor authentication and real-time intrusion detection, protecting sensitive patient data and ensuring high compliance with regulatory requirements.

3.2 FIWARE

FIWARE Foundation has a limited number of resources that consists of a small set of 3 Linux servers. Their configuration is:

- 32 CPUs per server
- 160Gb RAM per server
- 2Tb SSD per server

There are four extra Tb of data for backups.

Funded by the European Union
Project funded by

Regarding Software, FIWARE Foundation is one of the main actors providing solutions based on the ETSI NGSI-LD API such as Orion-ld Context Broker, which is capable of implementing the distribution and synchronization of NGSI-LD entities amongst a number of Orion-ld Context Brokers distributed across networks. Despite of the NGSI-LD broker, the FIWARE Foundation can provide a number of IoT Agents able to collect data from different sources like IOT devices or ad-hoc sources and seamless deliver the updated data to the Context Broker. This is ideal to spread the data between the different Kubernetes clusters and the CoGNETs Swam node. The communications between brokers or IoT Agents and brokers can be secured using PEP proxies and other additional software with this specific purpose.

3.3 SBA

Within CoGNETs, SBA contributes its extensive expertise in the area of cybersecurity and privacy. SBA is leading the task of end-to-end security, identity, privacy and resilience mechanisms to protect the CoGNETs middleware in a holistic manner. In particular, SBA supports the HW implementation partners in modelling attacks at the HW and firmware level. SBA's research expertise in decentralised identity systems, including the evaluation of self-sovereign identity architectures, will be helpful in performing a security assessment and proposing mitigation strategies for the chosen DID method. In addition, SBA will apply its knowledge of threat analysis and mitigation to application security. Given SBA's experience from various EU projects on ML-driven healthcare solutions, SBA will contribute its knowledge on (federated) machine learning security for adversarial threat analysis and mitigation of the distributed learning concepts (Adversarial Shield).

3.4 NCSRD

Within CoGNETs, the National Centre for Scientific Research "Demokritos" (NCSRD) contributes its expertise in network slicing through the Katana Slice Manager, an Open-Source platform for end-to-end network slicing management. While NCSRD does not provide dedicated hardware resources for deployment, Katana offers software-based capabilities that can be leveraged for dynamic resource allocation, orchestration, and automated slice management. This tool can support CoGNETs by enabling flexible and programmable network slicing across different infrastructures, ensuring optimized resource utilization and seamless integration with other partners' platforms. However, any deployment requiring additional computational resources or specific hardware dependencies will need to be hosted externally.

3.5 FHG

The Fraunhofer Institute for Production Systems and Design Technology (FHG-IPK) is located at the Production Technology Center Berlin (PTZ). The PTZ includes a central machine hall with various machine tools, industrial robots and other industrial equipment. In addition, several types of industrial control systems are available for prototyping and experiments.

During the CoGNETs project, FHG will use the following industrial equipment for building the manufacturing testbed and the pilot use case 1 (PUC1):

- Mobile Robot Platform MiR100
- Collaborative Lightweight Robot Universal Robot UR5
- Form-flexible Vacuum Gripper Formhand

Funded by the European Union

Project funded by

- 2D/3D Camera MS Azure Kinect v2
- Industrial Robot Arm KUKA Agilus
- Conveyor Belts with Servo Drives V90 and G120C
- Pneumatic linear cylinders with Festo Valves
- PLC Siemens S7-1200
- PLC Siemens S7-1500
- Soft-PLC CodeSys
- Mini-PC Intel NUC
- Edge Server Celsius
- Edge Cloud on premise (available in FHG-network)
- Cloud Server (available through Internet)

Given the above mentioned equipment, FHG will contribute with sophisticated control architectures and prototypes. The mobile robot and lightweight robot together with the 2D/3D camera and form-flexible gripper are combined into a basic mobile manipulation system. This system is controlled by a distributed external control infrastructure built on ROS2 and deployed between an onboard Mini-PC and the Edge Server. The industrial robot is combined with two conveyor belts and the pneumatic drives into a handling and assembly cell. The cell is controlled by the Codesys Soft-PLC running on a Mini-PC, the safety system is running on the S7-1200 PLC. The Soft-PLC is connected to the Edge Cloud for orchestration services.

FHG will further contribute with expertise in design, implementation and evaluation of distributed control systems for industrial equipment using the CoGNETs middleware and ecosystem.

3.6 VTT

VTT brings AI/ML expertise to support the development of the collaborative federated learning framework.

3.7 ULANCS

Within CoGNETs, ULANCS brings its extensive expertise and relevant infrastructure in Cyber Security, Edge, Fog, Cloud, IoT, SDN, machine learning, game theoretical modelling analysis and optimisation techniques.

Specifically, ULANCS will bring its expertise and infrastructure in game theoretical modelling and optimisation tool, GPUs computation capacity to support the development of CoGNETs game-intelligent agent systems for the autonomous decision-making.

Funded by the European Union

Project funded by

3.8 HMU

Within CoGNETs, the Hellenic Mediterranean University (HMU) will contribute by leveraging its strong technological background, advanced infrastructure, and proven expertise in cybersecurity. As a leading research institution with a well-established track record in network security, edge computing, and privacy-aware AI, HMU will be a key player in ensuring the resilience, efficiency, and security of the CoGNETs platform across its vertical applications.

As an integral partner in PUC3 – "Connected Healthcare (Collaborative AI for Medical Data Analytics in Health 4.0)", HMU will provide both software and hardware capabilities, hence facilitating secure, AI-powered health analytics. Additionally, complementing these efforts, HMU's deep knowledge of data manageability will ensure that the systems are scalable, adaptable, and secure, hence maintaining integrity and confidentiality of critical information in dynamic environments.

Beyond its technological capabilities, HMU will support CoGNETs with its computational infrastructure, which is designed to support large-scale data processing, AI model training, and experimental validation. Its high-performance computing environment includes a Dell R960 server, three Dell R640 servers, two Fujitsu Primergy TX150 S7 systems, and an NVIDIA A100 accelerator, providing the necessary power for scalable machine learning workflows and federated AI training.

Finally, HMU will utilise a pseudo-anonymised and fully compliant with privacy regulations dataset of network flows for AI-driven anomaly detection within its premises in its dedicated computing environment, ensuring that these processes are executed efficiently, securely, and in accordance with stringent data protection policies.

3.9 UAVIGN

At UAVIGN, there is a computing cluster, which is used for scientific computation. The characteristics of the cluster are the following:

- Compute servers: 27
- CPU: 552 cores
- Total RAM: 5.6 TB
- GPU cards: 71
- Total VRAM: 1424 GB
- Task scheduler: Slurm

3.10 INTRA

INTRA possesses the necessary infrastructural and technological capacity to support the deployment and operation of the CoGNETs testbed components. INTRA's infrastructure hosts the environments that will be utilized by the CI/CD processes, as they will be defined within WP2, to instantiate the appropriate DevOps & MLOps pipelines. The following specs are currently foreseen for the CI/CD server of CoGNETs:

Funded by the European Union
Project funded by

Sample HW Specs:

- CPU: 4x
- RAM: 8GB
- HDD: 160GB

Besides the CI/CD server, INTRA also hosts the underlying communication middleware Streamhandler, that will be utilized by the project.

Sample-starting specs for a deployment of Streamhandler: CPU: 8x - 16x

- RAM ≥ 16GB
- HDD ≥ 240GB

Resource scaling will be periodically reviewed and adjusted based on project needs, ensuring flexibility and responsiveness to evolving requirements.

3.11 MEDITECH

Meditech has a FIWARE node “MediHub” made of an OpenStack cluster with 8 Dell servers and a synology of about 100TB. The cluster is dedicated to the projects of Meditech's network partners, it will be possible to define one or more tenants to be possibly dedicated to the CoGNETs project, but to do so, the tenant characteristics would have to be received from the project. The tables below show the cluster components and the network and support components installed.

Table 3: MEDITECH Cluster components

Role	Quantity	Brand - Model	Description
MGMT-NODE	2	DELL - R440	DELL - R440
COMPUTE-NODE	9	DELL - R440	DELL - R440
Storage	1	Synology - RS1221+	Synology Single Power 8 Slot
MGMT-SWITCH	1	DELL - N1524	DELL N1524 24P x Gigabit
IDRAC-SWITCH	1	ARUBA HP2530	ARUBA HP2530
EROG-SWITCH	2	DELL - S4112T (Sx)	DELL S4112-ON (12PX10GB)
Firewall	2	FGT - 100F	Fortinet Firewall
UPS	2	DELL	UPS

Funded by the European Union
Project funded by

Table 4: MEDITECH Hardware characteristics

Q.ty	Model	CPU1	CPU2	RAM	Disk	NIC1	NIC2
3	PowerEdge R440	Intel® Xeon® Silver 4216 CPU 2.10GHz 16 Core	Intel® Xeon® Silver 4216 CPU 2.10GHz 16 Core	16GB	2TB 7.2 SATA	Broadcom Gigabit Ethernet BCM5720	Broadcom Adv. Dual 25Gb Ethernet
8	PowerEdge R440			20GB	900GB 15 SATA	Broadcom Gigabit Ethernet BCM5720	Broadcom Adv. Dual 10GBASE-T Ethernet
1	Synology RS1221+			4GB	8TB SATA Seag-ate Iron-Wolf		

Table 5: MEDITECH installed components

Role	Version	Name
OS	22.04 Jammy Jellyfish	Ubuntu
IAAS	6.0.0	OpenStack-Ansible
Compute Management Service	26.1.0	NOVA
Image Service	4.1.0	GLANCE
Identity Service	22.0.0	KEYSTONE
Network Management Service	8.1.0	NEUTRON
Placement Service	PLACEMENT	
Block Storage	9.1.0	CYNDER
Dashboard	23.0.0	HORIZON

3.12 SIEMENS

Siemens brings expertise in industrial computing infrastructure management to the CoGNETs project, emphasizing DevOps practices such as infrastructure automation, Continuous Integration and Continuous Deployment (CI/CD) and monitoring. By applying these methodologies, Siemens ensures a seamless integration of the CoGNETs middleware components in the manufacturing pilot use-case, while ensuring sustainability by reduced energy consumption and efficient resource allocation.

Funded by the European Union
Project funded by

3.13 AXON

AXON LOGIC (AXON) is an information and technology company that provides reliable support services and consultancy to various sciences, such as applied mathematics, quantum physics, electronics and communications engineering. AXON accurately delivers new technology and business information, ideas, and insights to customers worldwide.

Within CoGNETs, AXON provides extensive technological expertise in game theory optimisation to model strategic interactions between autonomous agents that make decisions leveraging RL and Deep RL machine learning paradigms. At the same time, enabling decentralised model training and preserving data privacy via collaborative federated learning technologies at the Edge.

In addition, AXON enhances and adapts its Open Service Repository platform to accommodate a selection toolkit of the newly derived CoGNETs AI services as an external middleware component named Cognitive Service Repository (CSR).

3.14 BEYOND

Hardware security guarantees require an actual (non-virtual) physical component which enables on one side immutability of digital logic and on the other its programmability / upgradeadility to support research and development. To achieve these somewhat conflicting needs an FPGA (“programmable chip”) is going to be used. The content of the FPGA will be loaded with specific SoC (system on chip) design comprising a RISC-V processor, PUF and other necessary hardware digital logic. It is expected that such hardware component will be provided in the form of a USB dongle (subject to size & power requirements) with USB interface to enable ease of connectivity with the servers and edge devices. The API of the hardware component, which will be defined by BEYOND, is expected to be minimalistic to reduce the attack surface and minimize the hardware complexity, power consumption and footprint.

3.15 K3Y

K3Y is an innovative SME specializing in Cybersecurity and Artificial Intelligence (AI), bringing expertise as an AI and Information Technology (IT) specialist. The company has extensive experience in Explainable AI (XAI) and has developed an XAI tool through a previous EU-funded project. Additionally, K3Y excels in privacy-preserving Machine Learning (ML) techniques, including Federated Learning (FL), enabling secure and decentralized AI model training while ensuring data privacy and compliance.

3.16 UBI

UBI is a software house that delivers state-of-the-art applications in various domains while it also has strong experience in cloud computing and cloud-edge continuum technologies, data processing and cybersecurity. Various cloud technologies have been investigated, including Kubernetes and Knative, while an internally developed cloud orchestrator (MAESTRO) and a cloud for hosting services and experimentation are owned. The cloud testbed includes more than 350 cores (~1400 vCPUs) at 2.6GHz with 2.6TB RAM and NAS with 166.4TB effective capacity (block storage & POSIX-based filesystem). Provisioning is made via Proxmox. Attached with an Edge/Fog computational cluster with x86, ARM and RISC-V devices.

Funded by the European Union

Project funded by

4 STATE OF THE ART ANALYSIS

This section describes the SotA of the different topics to be implemented in the project, taking into account the analysis of the literature. Hence, this section includes desk research on IoT-Cloud-Edge Continuum architectures, game optimization strategies, cognitive computing and programming models, federated learning mechanisms, swarm-wise distributed security paradigms, and data manageability, scalability and adaptability mechanisms.

4.1 IOT-CLOUD-EDGE CONTINUUM ARCHITECTURES

The increasing adoption of wearable interconnected devices and other IoT technologies have led to the need of new infrastructure which will have the capabilities to manage the huge amount of devices and the correlated data produced and transferred. Cloud-Edge-IoT (CEI) continuum, is the new concept introduced to manage this evolution, where IoT nodes interact seamlessly with edge devices and cloud systems to ensure efficient data processing and task execution. As authors in [1] noted, due to the limited battery and computational capacities of IoT devices, the majority of data processing is held on external systems, more specially on edge servers or in the cloud. This approach minimizes latency and ensures timely responses, which are critical for modern applications.

In addition, as highlighted in [2], due to the evolution of IoT devices, cloud computing models are challenging to manage the applications and the data they have to transfer. In this context, Osmotic Computing emerges as a novel paradigm, providing deployment and migration strategies tailored to the infrastructure and application requirements across Cloud, Edge, Fog, and IoT layers. Software abstractions like MicroELements (MEL), enable smarter orchestration of these systems which encapsulate resources, services, and data necessary for IoT applications.

A novel approach to enhancing programmability and adaptability in the CEI continuum has been proposed in [3] through RVE Accelerators. These accelerators enable elastic adaptation and resource optimization by supporting end-to-end programming, monitoring, and configuration for swarm-based applications. By coordinating elasticity across layers, this framework ensures robustness, scalability, and cost efficiency, addressing the dynamic requirements of modern IoT ecosystems.

As discussed in [4], recent advancements in CEI architectures have led to new solutions for efficiently managing and executing data science workflows. One approach is the Just-in-Time Architecture for Data Science Pipelines (JITA-4DS) framework. This framework focuses on providing flexible and on-demand resource allocation by creating Virtual Data Centers (VDCs) tailored to the specific needs of each data processing pipeline. These VDCs are dynamically assembled based on the system's requirements, ensuring that the goals of the pipeline, known as Service Level Objectives (SLOs), are met effectively. JITA-4DS takes advantage of disaggregated data centre configurations, proving it can pool resources from multiple locations and allocate them as needed. This makes it possible to achieve a balance between competing priorities, such as reducing energy consumption and maintaining strong performance for big data tasks. By supporting such dynamic and scalable resource management, the framework addresses many of the challenges faced by IoT-edge-cloud systems, particularly those involving workloads that require high adaptability and resource efficiency. This makes JITA-4DS an important contribution to the development of modern IoT-edge-cloud architectures.

In addition, authors in [5] note that managing CEI Continuum systems presents significant challenges due to their complex and multilayered architectures, as well as the heterogeneity

Funded by the European Union
Project funded by

of their components. Alongside, the integration of IoT, Edge, Cloud computing, and artificial intelligence requires advanced solutions to manage scalability, security, and privacy concerns. For these reasons authors [5] proposed a Data Management Framework (DMF), which provides a unified approach to overcome these challenges by incorporating differential privacy, energy efficiency and data visualization tools into a single, adaptable environment. The proposed framework enables developers to design, configure, and validate systems while it ensures secure, scalable, and privacy-compliant deployments. Key features include a differential privacy plugin, a traceability plugin for monitoring data flows and tools for real-time power consumption monitoring to enhance energy efficiency. It is validated through real-world applications, such as hydro-monitoring in water treatment systems and broadcast tower monitoring, the DMF demonstrates its potential for improving the management and security of IECC systems across diverse industries like healthcare, smart cities and industrial IoT.

Furthermore, the continuous increasing demand for scalable and secure data exchange has led to innovative solutions that address data interoperability and privacy. A notable contribution in this area is the IoTFeds platform, which combines an open-source interoperability framework with blockchain technologies to enable decentralized federation management and secure marketplace operations. As authors described in [6], IoTFeds allows IoT data providers and consumers to share, exchange, and monetize IoT data services through trusted transactions. The platform's architecture adopts a microservices-based, containerized design, ensuring scalability and high performance for managing federations and marketplaces.

The CEI continuum represents a transformative shift in managing the increasing complexity of interconnected devices and data-driven systems. Through advancements like Osmotic Computing, RVE Accelerators, JITA-4DS, and the IECC DMF, researchers are addressing critical challenges in scalability, privacy, security, and resource optimization. These frameworks and tools collectively enable the seamless integration of IoT, Edge, and Cloud layers while improving efficiency, adaptability, and robustness in a wide range of applications, from healthcare to industrial IoT.

4.2 GAME OPTIMIZATION STRATEGIES

In today's dynamic computing landscape, resource allocation is increasingly performed in distributed environments such as cloud and edge computing. At the intersection of economics and computer science, game theory provides a robust framework to analyse strategic interactions among rational agents [7]. Auctions, a class of market mechanisms rooted in game theory, are used to allocate scarce resources in an efficient and fair manner. This document is organized into five parts. First, we introduce the basics of game theory, outlining its core concepts and relevance to strategic decision-making. Second, we describe auctions as specific mechanisms for resource allocation and how they embody game-theoretic principles. Third, we present an overview of edge and cloud computing, highlighting the challenges of resource distribution in these environments. Fourth, we explore the connections between auctions and game theory, illustrating how auction models serve as a practical application of game-theoretic ideas. Finally, we discuss how game theory is applied specifically to edge computing, with emphasis on auction-based resource allocation mechanisms that address the unique characteristics of decentralized networks. Together, these five sections provide a comprehensive survey of the role of auctions and game theory in modern distributed computing systems.

Funded by the European Union

Project funded by

Game theory

Game theory is a branch of optimization theory that studies mathematical models of strategic interaction among rational agents [8], [9]. It helps predict outcomes in competitive situations where players (agents) make decisions based on the choices of others.

The fundamental components of game theory include:

- Players: The individuals or entities involved in the game.
- Strategies: The options or actions available to each player.
- Payoffs: The rewards each player receives as a result of the chosen strategies.
- Nash Equilibrium: A set of strategies where no player can improve their payoff by changing their strategy unilaterally [10].

Game theory provides a framework for understanding how participants interact and make decisions in environments where outcomes depend on the actions of others, making it highly applicable to competitive resource allocation scenarios such as auctions [11].

Auctions

Auctions are fundamental market mechanisms that facilitate the efficient allocation of scarce resources by allowing buyers to place bids, with the highest (or in some cases, the lowest) bid determining the transaction outcome [12]. Auctions have been extensively studied in economic theory due to their ability to balance supply and demand, maximize revenue, and ensure fair competition among participants [11]. The study of auctions dates back to early economic literature, with formalized models developed in the mid-20th century. Vickrey's seminal work introduced the second-price sealed-bid auction, demonstrating that truthful bidding is a dominant strategy [13]. Milgrom and Weber later expanded the field by analysing auction behaviour under asymmetric information and risk aversion [12].

Several auction formats have been proposed in the literature, each designed to address different allocation problems and strategic behaviours:

- English Auctions: Also known as open ascending-price auctions, bidders publicly announce increasing bid amounts until no higher bids are placed. This format is commonly used in traditional and online marketplaces, such as eBay [14], and is widely analysed in auction theory due to its transparency and price discovery properties [11].
- Dutch Auctions: In contrast to English auctions, Dutch auctions follow a descending-price format, where the auctioneer starts with a high asking price and lowers it until a participant accepts the offer [15]. These auctions are efficient for perishable goods and high-speed trading environments, such as stock exchanges [11].
- Sealed-Bid Auctions: In sealed-bid auctions, bidders submit private bids without knowing the bids of competitors. The highest bid wins in a first-price auction, whereas in a second-price auction (Vickrey auction), the highest bidder pays the second-highest bid amount [13]. This format is commonly used in government contracts, markets of natural resources, and real estate [16], [17].
- Combinatorial Auctions: In many real-world applications, items are complementary, meaning their value in combination is higher than the sum of their individual values.

Funded by the European Union
Project funded by

Combinatorial auctions allow bidders to place bids on bundles of items, making them particularly useful for spectrum allocation, transportation logistics, and cloud computing resource distribution [18], [19].

Game theory for edge computing

Edge computing, due to its decentralized nature and limited resources, presents unique challenges for resource allocation. Unlike traditional cloud computing, where centralized resource allocation mechanisms can be implemented with global information, edge computing requires distributed decision-making approaches that account for heterogeneous devices, varying network conditions, and real-time constraints [20], [21].

Game theory provides a mathematical framework for modelling competitive and cooperative interactions among rational agents. In the context of edge computing, game-theoretic models help design incentive-compatible, scalable, and efficient resource allocation mechanisms. Unlike static resource allocation approaches, auction-based game theory models enable dynamic pricing and adaptive decision-making, ensuring efficient resource utilization in decentralized edge environments [22].

Challenges in Edge Resource Allocation. The unique characteristics of edge computing introduce several challenges in resource allocation, which are also part of the CoGNETs agenda.

1) Decentralized Decision-Making: Traditional centralized auctions may not be suitable for edge environments, where re-sources are distributed across multiple entities with no global coordinator [23].

2) Scalability and Real-Time Constraints: Edge computing requires fast and adaptive mechanisms for resource pricing and allocation to handle fluctuations in demand [22].

3) Security and Privacy Concerns: Secure allocation mechanisms are needed to protect user data and prevent malicious behaviour in auction-based bidding [24].

Game-Theoretic Auctions for Edge Computing

Game-theoretic auction mechanisms have been proposed to address these challenges by ensuring efficient, fair, and secure resource allocation. These models provide the following advantages:

- Distributed Resource Allocation: Auction models help allocate edge resources efficiently without requiring a central controller. Mechanisms such as double auctions and combinatorial auctions enable fair and market-driven allocation [23], [25].

- Dynamic Bidding Mechanisms: Edge computing environments experience dynamic fluctuations in resource availability. Game-theoretic, dynamic auctions allow participants to adjust their bids in real time, ensuring optimal pricing and allocation [22].

- Privacy and Security Considerations: Secure blockchain-based auction frameworks have been proposed to enhance transparency and prevent fraudulent bidding behaviour in edge computing [24], [26].

Several studies have explored auction-based game-theoretic approaches for edge computing:

1) Auction-Based Resource Allocation: Chen et al. [25] and Wang et al. [23] analysed auction-based models for efficient resource allocation in edge and cloud computing.

Funded by the European Union
Project funded by

2) Game-Theoretic Models for Edge Networks: Xu et al. [22] proposed adaptive pricing strategies for dynamic edge computing environments.

3) Blockchain-Based Auctions: Avasalcai et al. [26] studied how decentralized blockchain-enabled auctions improve trust and security in edge computing.

Task offloading and bidding

One of the applications of game theory for resources sharing is task offloading. In edge computing this refers to the process of dynamically deciding whether computational tasks should be executed locally on edge devices or offloaded to edge servers or cloud data centres [26], [27], [28]. Unlike traditional cloud computing, where resource allocation is managed in a centralized manner, edge computing introduces decentralization and resource constraints making task offloading a complex optimization problem [20], [21]. Game theory is effective to capture strategic decision-making in edge computing environments, where multiple users, edge nodes, and service providers interact in a competitive or cooperative manner to optimize latency, energy efficiency, and resource utilization [22], [23]. By leveraging game-theoretic models, task offloading decisions can be made dynamically, efficiently, and fairly while ensuring stability in resource allocation. In CoGNETs, we will use task offloading as a benchmark use case for the application of our bidding solutions, as developed in WP3.

Challenges

Task offloading in edge computing introduces several challenges that are addressed using game-theoretic solutions:

1) decentralized resource management: edge computing lacks a centralized authority, requiring distributed decision-making mechanisms where multiple players (devices, servers) compete for limited resources [29];

2) heterogeneous computing environments: edge devices and servers have varying computational power, energy capacity, and bandwidth availability necessitating adaptive offloading strategies [30];

3) latency and energy constraints: offloading decisions must balance minimizing latency (for real-time applications) and reducing energy consumption (for battery-powered edge devices) [27];

4) security and privacy issues: offloading tasks to external edge nodes poses risks related to data integrity, trust, and privacy preservation which game-theoretic models can address [31].

Game-Theoretic Models for Task Offloading

A structured approach for optimizing task offloading decisions in decentralized edge computing environments has been developed in the literature via different strategic framework.

Non-Cooperative Game Models: when multiple self-interested edge devices compete for limited resources, non-cooperative game models (e.g., Nash equilibrium-based strategies can be used to determine optimal task offloading strategies [32].

Cooperative Game Models: edge nodes can form coalitions to share computing resources, reducing latency and improving efficiency. Coalition formation games and bargaining models have been proposed to enable collaborative offloading [33].

Funded by the European Union
Project funded by

Stackelberg Game Models: in hierarchical edge-cloud environments, Stackelberg games model the interaction between a leader (edge server) and followers (edge devices) to determine optimal pricing and workload distribution [23], [31].

4.3 COGNITIVE COMPUTING AND PROGRAMMING MODELS

Cognitive computing, which leverages artificial intelligence (AI), cloud computing and advanced data analytics to emulate human cognition and reasoning, is transforming how computing systems process and respond to vast datasets. These intelligent systems enable automated decision-making, pattern recognition, and adaptive learning in real time. As digital ecosystems grow more complex, the need for seamless interoperability between diverse programming models becomes increasingly critical. This is especially true in distributed environments where data is generated and processed across multiple layers. Within the IoT-Cloud-Edge continuum, the challenge lies in ensuring that cognitive computing models can operate consistently across heterogeneous infrastructures-ranging from resource-constrained IoT devices to high-performance cloud servers [34], [35], [36]. Achieving interoperability across these layers is vital for enabling intelligent, real-time applications while minimizing latency and maintaining data privacy.

A core enabler of cognitive computing in this continuum is the evolution of programming models that support modularity, scalability, and cross-platform execution. Traditional centralized computing approaches are giving way to more distributed paradigms such as serverless computing [37] and microservices [38]. These models facilitate the deployment of AI workloads across different environments while maintaining flexibility. However, the diverse nature of hardware and software in IoT, edge, and cloud environments requires interoperable frameworks. Initiatives such as Open Neural Network Exchange (ONNX) and Eclipse Kura are addressing these challenges by providing open standards for AI model execution and device-level management. These frameworks ensure that cognitive applications can seamlessly transition from the cloud to the edge, enabling real-time processing closer to data sources while maintaining coordination with centralized systems. Another example is WebAssembly (Wasm), which offers a platform-independent binary format that enables efficient execution of complex algorithms on both low-power IoT devices and high-performance cloud environments [39]. By facilitating the seamless transfer and execution of cognitive workloads, these interoperability frameworks are breaking down barriers between disparate computing environments.

The interoperability of programming models also relies on orchestration and resource management across the IoT-Cloud-Edge continuum. As cognitive workloads become more dynamic, efficient scheduling and workload distribution are essential for optimizing performance and resource usage. Kubernetes [40] has emerged as a dominant orchestration platform, providing automated deployment, scaling, and management of containerized applications. Extensions like KubeEdge enable Kubernetes to extend its orchestration capabilities to edge devices, allowing AI models to be deployed and managed seamlessly across different layers of the continuum. Furthermore, Eclipse ioFog provides edge-specific orchestration that integrates with cloud-native systems, ensuring that cognitive applications can dynamically allocate resources based on real-time demands. These frameworks are essential for maintaining operational efficiency and ensuring that cognitive processes are executed where they are most needed.

Interoperability challenges also extend to legacy system integration and cross-domain collaboration. Many industrial, healthcare, and automotive systems rely on proprietary protocols and legacy infrastructure, which can hinder the deployment of modern cognitive applications. Middleware solutions like Eclipse Kura bridge the gap by providing APIs and abstraction layers that connect legacy devices with modern cognitive computing frameworks. Similarly,

Funded by the European Union
Project funded by

OPC Unified Architecture (OPC UA) [41] provides a standardized communication protocol for industrial automation, enabling cognitive systems to interact with legacy equipment in manufacturing environments. In healthcare, FHIR [42] (Fast Healthcare Interoperability Resources) provides a standardized framework for exchanging patient data across digital health platforms, allowing cognitive models to access and analyse medical information securely. These interoperability frameworks are essential for enabling cognitive computing across sectors where legacy infrastructure is prevalent.

Looking forward, advancements in neuromorphic computing [43] and heterogeneous computing will further impact cognitive systems' design and interoperability. Neuromorphic architectures, inspired by biological neural networks, offer low-power, real-time processing capabilities suited for edge environments. Meanwhile, Gaia-X [44] is leading efforts to establish a federated data infrastructure for secure and interoperable collaboration across the IoT-Cloud-Edge continuum. These initiatives point toward a future where cognitive computing systems operate seamlessly across diverse environments, unlocking new possibilities for autonomous intelligence. As the complexity of distributed architectures continues to grow, ensuring robust programming model interoperability will remain a foundational challenge and opportunity for the next generation of intelligent, interconnected systems.

4.4 FEDERATED LEARNING MECHANISMS

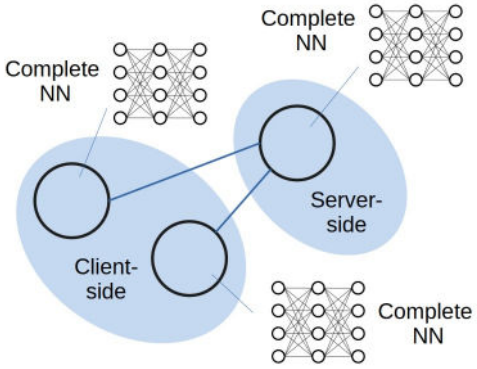
The training workflow in traditional machine learning frameworks requires to centralize the training data into a server. This setting presents a challenge when the goal is to produce a model that generalizes for training data that belongs to multiple owners and it is regarded as confidential. Federated learning [45] is a collaborative machine learning framework designed to keep the privacy of the training data. It is based on the principle of decentralizing the training and keeping the training data local in the domain of their owners. Federated learning allows multiple participants, referred to as clients, to contribute to the training of a machine learning model under the coordination of a central server.

In the typical federated learning framework, the model to be trained on the client-side and server-side is the same (complete model). Clients train locally and share model updates to the server. The server aggregates the models trained by the clients. Then, the aggregated models are shared to the clients. A typical setting is the so-called cross-silo federated learning, in which multiple organizations collaborate in the training of a model. Figure 2 shows cross-silo federated learning for the case of a machine learning model implementing a Neural Network (NN). The same model (complete NN) is trained at the client-side and at the server-side.

Funded by the European Union

Project funded by

Figure 2: Cross-silo federated learning.



In the case of large models, federated learning may require an amount of resources that small computing devices constrained in computing resources, such as IoT or mobile devices, cannot provide. Examples of the limitations of federated learning in the context of these devices are the following. (1) Federated learning requires full model training on the client-side. (2) Federated learning may not be feasible for IoT or mobile devices due to computing and other resource limitations to train a complete model (e.g. a whole NN). (3) The size of the model may be an issue in wireless networks, due that for large models the amount of data to communicate requires a suitable bandwidth.

In the context of the limitations of federated learning for small computing devices, a paradigm so-called split learning presents the possibility to divide a machine learning model into small sub-models. In the context of a model based on a NN, the NN is split into small sub-networks, resulting in parts with fewer layers than in the original NN. With such an arrangement, the key idea is to allocate for the training a few of the initial layers of the original NN to the clients, and the remaining part of the NN to a server.

Federated learning and split learning have their own benefits and drawbacks in terms of security, required computing resources, communications overhead, etc. Ongoing research in this field of study is focusing on creating hybrid approaches that combine the benefits from the two frameworks, as well as looking for alternative solutions. In the subsections that follow, we briefly discuss the different mechanisms based on federated and split learning.

4.4.1 Split Learning

Split learning [46] is a collaborative learning framework. It is presented as an alternative to federated learning for devices constrained in computing resources, such as IoT. Actually, other limitations on resources may be a factor to adopt split learning, for example, in terms of the energy source to power the device implementing the model (e.g. battery powered devices).

In the contexts of models based on a NN, split learning is based on splitting the NN and placing one part (sub-network) at the client-side (e.g. end-device side), and the remaining part of the NN at the server-side (for example, an edge server or core-cloud). Figure 3 shows an example of split learning with two clients and one server.

Funded by the European Union
Project funded by

Figure 3: Split learning example.

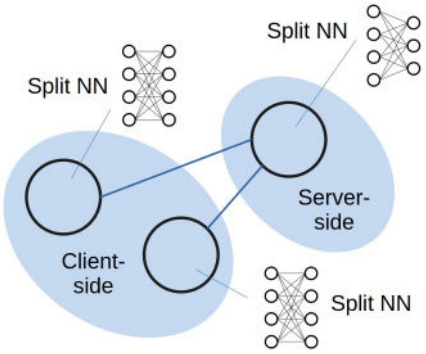
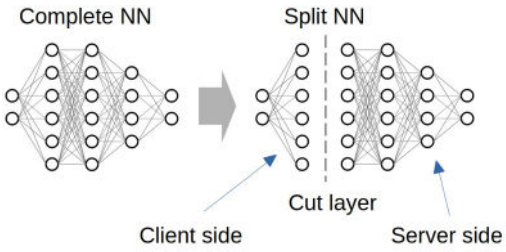


Figure 4 depicts a complete NN and its split version. The NN is split at a so-called cut layer. The corresponding entities assigned to process each part, namely client and server sides are shown in the figure. In clients constrained on resources, the part of the NN allocated to them is typically a small part of the complete NN. The remaining and major part of the NN is allocated to the servers (edge server or core-cloud). Thus, with such an arrangement, the major effort of the training workload is offloaded to the servers. The outputs at the cut layer is referred to as smashed data.

Figure 4: Split NN at the cut layer.



Split learning is targeted to protect data privacy. Data privacy is based on keeping a few layers and the raw training data at end/local devices for privacy preservation (data ownership). As a feature proper to this framework, split learning allows a better resource utilization distributed from the end-devices to the core-cloud. Thus, split learning can be used also in contexts where data privacy is not the main requirement, but the main requirement is a distributed computing effort for the training and inference of the machine learning models.

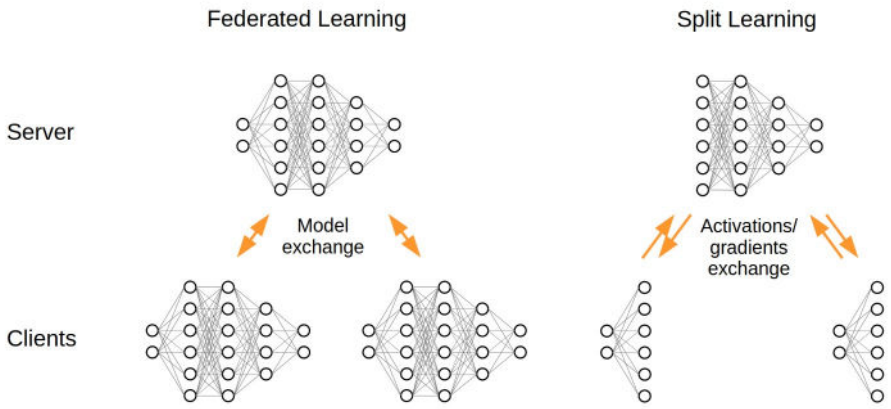
The training in split learning is sequential. A client performs the forward propagation of the training process first in the client-side, and then in the server-side. Then, backward propagation is performed in the opposite direction. Next, the process is repeated in another client. Thus, only one client is active at a time, the others are idle. The larger the number of clients, the larger the training latency.

Funded by the European Union
Project funded by

4.4.2 Federated Learning vs. Split Learning

Federated and split learning are shown and compared side by side in Figure 5 for an example case with two clients. In federated learning the whole model is exchanged between the clients and the server. In contrast, in the case of split learning, only the activations and gradients from the smashed data of the cut layer are exchanged.

Figure 5: Comparing federated and split learning.



Federated learning allows parallel model training, in contrast, split learning operates sequentially. However, in terms of communication overhead, in split learning the overhead is reduced since a fraction of the model and smashed data is communicated between a client and a server. *The choice between federated learning and split learning is discussed in [47].* If the computing capability of end devices is low (such as in the case of IoT devices) the choice is to use split learning. Communication wise, when the training dataset from the clients is large, federated learning may be a better option. However, split learning becomes a viable option when large volumes of smashed data are proportional to the size of the dataset. Split learning is more communication-efficient when the size of the model is larger than the size of the smashed data.

4.4.3 Federated Learning and Split Learning variants

A variant of the plain split learning, so-called U-shape split learning [46], keeps local the training ground-truth labels, so that these are not transferred to the server-side.

Split learning operates sequentially, trains the model for one client at a time. In contrast, Split-Federated Learning (SFL) [48] allows parallel forward propagation in all of the models of the clients and server. Then, parallel backward propagation is performed in the server and clients. The server updates its model by weighted averaging of gradients from clients' smashed data. Each client performs backward propagation on their client-side model and the gradients are computed. The gradients from the clients are sent to an auxiliary server (Fed server) that averages the gradients from the clients, and sends them back to all of the clients. A mechanism of differential privacy is used to make the gradients of the clients private. A variation of SFL is proposed in [49], allowing parallel processing without client-side model synchronization, and lower communications overhead than in SFL.

Cluster-based Parallel Split Learning (CPSL) [50] aims to reduce the training latency by combining parallel and sequential training stages. Clients are associated to clusters. Clients in

Funded by the European Union
Project funded by

the clusters are trained in parallel, and aggregate local models at the server. Then, the training of the global model across clusters is performed sequentially.

4.5 SWARM-WISE DISTRIBUTED SECURITY PARADIGMS

The swarm-wise distributed systems nowadays, evolved as one of the most evolutionary technologies due to their adaptability, efficient operation and resilience. Also, swarm-wise distributed security paradigms have emerged as a promising solution for safeguarding modern decentralized systems. Inspired by collective behaviours in nature, these paradigms offer scalability, adaptability, and resilience. This section reviews the current advancements, challenges, and gaps in this domain to establish the foundation for further exploration.

As authors mention in [51], swarm-wise intelligent security systems offer significant advantages as they are decentralized and self-organized and have adaptive behaviours. These systems have the ability to handle dynamic and complex environments, such as IoT networks, as they provide robustness, scalability and fault tolerance. Their distributed nature reduces reliance on centralized control, enhancing resilience against single points of failure and they improve response times to emerging threats. Additionally, there are swarm intelligence algorithms that enable efficient resource management, adaptive routing and proactive threat detection. These algorithms are ant colony optimization and particle swarm optimization, which are highly suitable for addressing evolving security challenges in modern interconnected systems.

Authors in [52] highlighted potential threats to swarm distributed systems. While many of these threats are similar to those affecting other technologies, swarm applications also face unique security challenges. For instance, security vulnerabilities have been observed in smaller devices due to resource constraints, such as limited storage, communication bandwidth, computational power, and energy. In the event of an attack, these limitations could lead to a loss of availability. Additionally, these constraints restrict the types of security measures that can be applied. Another potential threat is the compromise of a swarm intelligence security credential, which could impact other members within the same environment.

Despite the growing complexity and the significant advancements in IoT technology, numerous challenges have led to crucial issues regarding security and efficiency issues. Indeed, authors in [53] found out that security challenges in IoT systems extend to cloud-based mechanisms, which often may affect data storage and processing. Alongside this, security and authorization mechanisms designed for enterprise devices may be not compatible with cloud infrastructures and require significant adaptation. Also, another cloud challenge that authors mentioned is the difficulty of connectivity investigations due to the lack of physical access to system hardware and this may affect the availability and reliability of cloud services. In addition, critical risks have been observed in virtual machine's security, more specially in multi-tenant architectures where susceptibility of hypervisors or virtual machines could compromise multiple users on the same physical server.

The aforementioned challenges underscore the need for robust, decentralized, and adaptive security solutions capable of addressing the complexities of modern IoT and cloud ecosystems. Resource-constrained environments, such as wireless sensor networks (WSNs), have effectively addressed security challenges in distributed systems by employing innovative algorithms. For example, the authors in [54] proposed a secure routing protocol based on multi-objective ant colony optimization (SRPMA) to enhance WSN security while minimizing energy consumption. This approach improves the ant colony algorithm by incorporating multiple optimization objectives, such as node residual energy and the trust of routing paths. Additionally, it has demonstrated robust performance against black hole attacks in WSN routing [54]. Similarly, swarm intelligence-inspired approaches, such as Ant Colony Optimization

Funded by the European Union
Project funded by

(ACO) and other bio-inspired techniques, have proven effective in optimizing WSN routing and security, balancing trade-offs between energy efficiency, scalability, and reliability [55]. These strategies highlight the potential of leveraging decentralized and adaptive mechanisms to address critical challenges in WSNs and IoT environments.

Swarm-wise distributed systems and security paradigms offer scalable, adaptive, and resilient solutions to the growing challenges existing in CEI continuum ecosystems. Their decentralized nature reduces reliance on central control, enhances robustness, and enables efficient responses to threats. Despite advancements, challenges such as resource constraints, credential vulnerabilities, and cloud-specific security risks persist. Algorithms like Ant Colony Optimization (ACO) have shown ability in addressing these issues by balancing energy efficiency, scalability, and reliability [55].

Furthermore, authors in [56] proposed a novel swarm-based feature selection algorithm to enhance attack detection in Cyber-Physical Systems (CPS) integrated with IoT. Their approach utilises Enhanced Chicken Swarm Optimisation (ECSO) with self-learning capabilities to select relevant features from preprocessed data, which are then processed utilising ensemble classifiers in the cloud. This method which was tested on the NSL-KDD dataset, demonstrated strong performance in improving attack detection in CPS across multiple statistical measures. Finally, authors in [57] aim to address the challenge of securing IoT devices under energy and funding constraints through the introduction of a pricing model for security services provided by smart gateways. They formulated the security optimisation problem as a Mixed-Integer Linear Programming (MILP) problem and proposed a swam intelligence-based task scheduling scheme to efficiently solve it, significantly outperforming existing methods in security enhancement and task feasibility. These strategies highlight the potential of leveraging decentralized and adaptive mechanisms to address critical challenges in WSNs and IoT environments.

Swarm-wise distributed systems and security paradigms offer scalable, adaptive, and resilient solutions to the growing challenges existing in CEI continuum ecosystems. Their decentralized nature reduces reliance on central control, enhances robustness, and enables efficient responses to threats. Despite advancements, challenges such as resource constraints, credential vulnerabilities, and cloud-specific security risks persist. Algorithms like ACO have shown ability to address these issues by balancing energy efficiency, scalability, and reliability.

Additionally, the Internet of Things (IoT) is connecting billions of smart devices, enabling real-time data exchange and intelligent automation [35]. This number is expected to surpass 32 billion by 2030, significantly outpacing traditional identity management capabilities [58]. Current Identity and Access Management (IAM) systems, designed primarily for human-centric interactions, fall short in addressing the unique challenges of IoT ecosystems. One major issue is the lack of standardized protocols for device identification. Manufacturers often resort to proprietary systems for naming and managing devices, resulting in siloed ecosystems that hinder interoperability and scalability. As the number of IoT devices grows exponentially, centralized systems struggle with performance, often leading to bottlenecks or high operational costs. Another critical gap lies in lifecycle management. IoT devices require identity provisioning, updating, and revocation throughout their lifecycle. Legacy IAM systems are not equipped to handle these dynamic and resource-constrained requirements effectively. The absence of a unified framework leaves IoT devices vulnerable to identity spoofing, unauthorized access, and botnet attacks, as seen in high-profile incidents like the Mirai botnet, which exploited weak authentication mechanisms in IoT devices.

Decentralized Identifiers (DIDs) may offer a robust alternative for device identification and authentication, by removing the dependency on central authorities, enhancing scalability

Funded by the European Union
Project funded by

through distributed ledger technologies as trust anchors, and ensuring privacy and integrity through cryptographic methods. DIDs enable IoT devices to have autonomous control over their identities, facilitating secure, direct communication and authentication, thereby establishing a decentralized public key infrastructure (DPKI). DIDs are a novel type of identifier, standardized by the World Wide Web Consortium (W3C) [59], designed to enable verifiable, self-sovereign digital identities within decentralized systems. Unlike traditional identifiers that rely on centralized or federated registries and authorities, DIDs are created, managed, and controlled by their owners without intermediaries. Think of DIDs as globally unique identifiers that are cryptographically generated and operate independently of central authorities. Each DID points to a corresponding DID document (e.g., JSON-LD), which encapsulates essential information such as public keys for authentication, verification methods, and service endpoints.

Core Concepts of DIDs

- DID Syntax: did:<method>:<identifier>,
  e.g., did:iota:0x0a6cf85kzfaff3c4c9097ce91d84b1df4pu75r439a64a5b6e-f30476cekj83ed3:

- DID Document: Contains public keys, verification methods, and service endpoints. Like DNS maps URLs to IP addresses, DID resolution maps a DID to its DID Document.

- DID Method: Defines CRUD operations (Create, Resolve, Update, Deactivate) for the DID.

DID methods vary based on their approach to core operations (create, read, update, deactivate), their governance structures, network design, and registry mechanisms. Some methods rely on permissionless networks, allowing open participation, while others use permissioned systems with restricted access, impacting decentralization [60], [61]. Additionally, methods vary significantly, particularly in areas critical for IoT ecosystems, such as security, scalability, interoperability and operational efficiency to authenticate and manage devices across dynamic and decentralized ecosystems.

The paper by Kortesniemi et al. [62] examines cryptographic algorithms used with DIDs on constrained IoT devices. It reveals that even devices with limited resources, such as 8-bit microcontrollers, can deploy DIDs directly. However, for devices with even more stringent constraints that face challenges with the computational overhead and storage requirements of public-key cryptography, or where the security implications of storing sensitive cryptographic keys on-device are significant, a proxy approach can be used. This method offloads the computationally intensive cryptographic operations to external, more capable systems, thereby allowing these resource-limited IoT devices to leverage the benefits of DIDs without bearing the full burden of cryptographic complexity.

An OAuth-based delegation method was also proposed by Lagutin et al. [63] to offload DID processing from resource-constrained IoT devices to authorization servers. This approach ensures even low-powered devices can benefit from decentralized authentication while maintaining operational efficiency. Fedrecheski et al. [64] introduced a lightweight DID implementation tailored for constrained IoT networks, i.e., swarm DID method. They employ Optimized DDO (DID Document Object) Serialization using CBOR-DI (Concise Binary Object Representation for Decentralized Identifiers), a compact binary format that reduces the size of the DID document, thus improving both storage efficiency and transmission speed. They also introduce DIoTComm, a protocol designed for low-latency, low-bandwidth communication in IoT networks. The method demonstrated a fourfold reduction in metadata size, enabling better performance in constrained environments.

Funded by the European Union
Project funded by

The paper by Fan et al. [35] introduces DIAM-IoT, a decentralized identity and access management framework for IoT ecosystems. It breaks IoT application silos using blockchain as a bridge for decentralized data authorization, and integrates Decentralized Identifiers (DIDs) and Verifiable Credentials (VCs) enabling unified identity management and cross-application interoperability. The framework also allows device owners to define user-specific rules for granting data access requests.

Ansay et al. [65] propose Gnomon, a DID-based system, for secure 5G IoT device registration and software updates. Unlike traditional PKI systems, Gnomon uses Decentralized Identifiers (DIDs) and verifiable credentials to ensure that devices can securely authenticate software updates without risk of credential revocation or expiration. A prototype called Ionic, based on the Microsoft ION network, was developed to enable memory-constrained devices to verify and download software updates securely.

Han et al. [66] analysed SRAM-based PUFs to generate unique signatures for DIDs in large-scale IoT networks. They demonstrated the feasibility of using PUFs for distributed device identification, ensuring uniqueness and scalability across IoT environments, considering both chip-by-chip and block-by-block uniqueness. They also showed that SRAM-PUFs maintain high uniqueness and reliability even in extensive deployments.

Su et al. [67] developed a secure decentralized identity management system for IoT devices, integrating blockchain, Physical Unclonable Functions (PUFs), and True Random Number Generators (TRNGs). Their approach introduces Decentralized Machine Identifiers (DMID) that enhance device security by preventing unauthorized access and software tampering. The system also includes a hardware module that leverages TrustZone to isolate security-sensitive operations. This approach minimizes the risks of single points of failure and DDoS attacks, ensuring secure device registration, authentication, and software updates without relying on centralized authorities.

Javaid et al. [68] proposed using PUFs in combination with Ethereum blockchain smart contracts to establish secure device identities and ensure data provenance in IoT environments. This approach uses PUFs' unique hardware-based security to ensure that each device has an authentic, unforgeable identifier. The Ethereum blockchain adds a layer of tamper-resistant storage, providing data integrity and a secure audit trail. By integrating both technologies, their system enables robust device authentication and data provenance without relying on traditional centralized infrastructures.

Patil et al. [69] proposed a privacy-preserving authentication protocol that integrates PUFs with blockchain-based smart contracts. The protocol aim is to enhance security and reduce authentication overhead in IoT networks. The method is used for ensuring data provenance and transparency while maintaining lightweight operations suitable for constrained IoT devices.

4.6 DATA MANAGEABILITY, SCALABILITY AND ADAPTABILITY MECHANISMS

This section provides an overview of the state of the art overview for Data manageability, scalability and adaptability mechanisms supported by Next Generation Service Interface with Linked Data (NGSI-LD) API, standardized by ETSI. This has emerged as a foundational framework for managing context information in FIWARE ecosystems. Its adoption in smart cities, IoT, and digital twin applications underscores its technical robustness in addressing data interoperability, scalability, and adaptability challenges.

Funded by the European Union

Project funded by

Data Manageability in ETSI NGSI-LD API

NGSI-LD's core data manageability arises from its property graph model, where entities, properties, and relationships are modelled as linked data using JSON-LD. This approach enables dynamic updates, subscriptions, and queries while ensuring semantic consistency. FIWARE Context Brokers implement this model to provide robust and federated data management across heterogeneous sources. Most relevant key aspects are:

- **Context Broker Architectures**: FIWARE brokers manage the full lifecycle of context data—from ingestion and update to query and subscription notification. In practical deployments, these brokers can operate as stand-alone nodes or be federated across domains to handle high data loads. A federated architecture allows each broker to manage a partition of the data while providing a unified interface for global queries. Sanchez et al. [70] demonstrated that a distributed broker architecture can efficiently manage data from large-scale IoT deployments by partitioning data among brokers without compromising the unified NGSI-LD interface.

- **Semantic interoperability**: The Smart Data Models Initiative, supported by FIWARE, provides domain-agnostic schemas (e.g., Satellite Imagery, Risk Management) that harmonize terminology and enable cross-domain data sharing. Li et al. [71] highlighted that mapping JSON-LD @context definitions to standardized URIs is crucial for achieving cross-domain interoperability and for enabling federated queries in complex smart city scenarios. This semantic consistency is critical when federated queries span multiple context brokers or when integrating data from diverse sources such as IoT devices, satellite imagery, or environmental sensors.

- **Entity Linking and Data Curation**: Advanced techniques for automatic entity linking and enrichment help mitigate issues such as data redundancy and quality control. In the context of IoT, Perera et al. [72] emphasized that robust data curation and the chaining of relationships between entities are key to reliable context management, thereby reducing the need for manual intervention during data integration.

Scalability mechanisms

Scalability in NGSI-LD systems is critical for handling the massive data volumes generated by IoT networks. Several mechanisms ensure that the system can grow in a modular and efficient manner. NGSI-LD's RESTful API and federated context broker design enable horizontal scaling. Starting from a single context broker, systems can evolve into a distributed network where multiple brokers are deployed across the Edge-Cloud continuum. Botta et al. [73] describe how a federated context broker approach distributes the processing load by assigning different data partitions to different brokers. This enables the system to maintain low latency even when the number of concurrent entities grows significantly. The use of technologies such as Apache Kafka, as reported by Sanchez et al. [70], further supports high-throughput data ingestion and real-time notifications.

To mitigate latency, caching solutions (e.g., Redis) and load balancing (using tools like NGINX or Istio) are deployed to serve frequently accessed context data. In a study by Aazam et al. [74] on distributed IoT systems, such caching mechanisms were shown to significantly reduce response times and balance the computational load across the network, thereby ensuring that the NGSI-LD API remains responsive under heavy load.

Funded by the European Union
Project funded by

The asynchronous publish/subscribe model inherent in NGSI-LD enables the system to manage thousands of real-time data streams. By decoupling data ingestion from processing—often via serverless functions or stream processing frameworks—systems can ensure that context brokers remain lightweight. This architectural choice, discussed in detail by Botta et al. [73], is critical for supporting real-time applications in smart cities and industrial IoT environments.

Adaptability strategies

Adaptability mechanisms ensure that NGSI-LD systems can evolve as new technologies and domain requirements emerge. FIWARE's IoT Agents translate legacy communication protocols (e.g., MQTT, LoRaWAN) into NGSI-LD–compliant messages. This not only facilitates the integration of existing devices but also supports incremental migration to modern data architectures. Perera et al. [72] note that this protocol agnosticism is critical for maintaining interoperability in heterogeneous environments.

The microservices approach allows the replacement or upgrade of individual system components without impacting the overall architecture. Li et al. [71] document that modularity in FIWARE ecosystems enables dynamic reconfiguration—such as swapping data enrichment modules or time-series databases—which is essential for adapting to sector-specific requirements or evolving standards.

Dynamic enrichment techniques—using external knowledge bases or semantic web tools—enable real-time context augmentation. Sanchez et al. [70] illustrated that by incorporating dynamic context enrichment, systems can integrate additional data sources (e.g., weather data, traffic information) on the fly, thereby enhancing situational awareness. Additionally, cross-domain federation mechanisms ensure that context data can be shared securely and consistently across different administrative boundaries.

Funded by the European Union

Project funded by

5 ANALYSIS OF THE EXPERTS

This section provides the answers obtained from the experts for the same topics presented in the previous section. The name of the experts is not published unless we have received an explicit confirmation from them for GDPR compliance. All experts except one have given this explicit confirmation. The expert who didn't give the confirmation to publish his name will be titled as expert X. Besides, a minor introduction of each expert is presented in order to justify the selection of the persons. The answers provided by the experts facilitated the requirement elicitation for the CoGNETs project.

5.1 ANALYSIS OF DR. IGNACIO LACALLE ÚBEDA

Ignacio Lacalle Úbeda is a Senior Researcher at UPV and serves as the deputy project co-ordinator for the O-CEI project. Due to his expertise, we believe he is well-suited to address questions related to IoT-Edge-Cloud swarm continuum architectures.

1. ***How do you manage dynamic task allocation and resource optimization across the IoT-Edge-Cloud continuum in swarm-based architectures, especially in latency-sensitive applications?***

   Currently, we handle the resource optimization by complying with the following premises:

   - We keep an updated record of the current status of the resources available per node that form the swarm (members of the continuum). We achieve this via the federation of Context Brokers and a hierarchical domains structure.
   - We take the decision of workload allocation based on such status, and the explicit requirements (SLA – Service Level Agreement) and the components description of a containerized application.
   - Such decision undergoes an AI process. In particular, we apply two different algorithms, that are selected by the user:
     - A reinforcement learning approach, that increases the performance as the continuum evolves.
     - A multi-parameter (KPI) optimization algorithm drawing from a pre-profiling of the workload and the potential impact in the continuum.

2. ***What strategies or frameworks do you recommend for maintaining data consistency and security as data moves between IoT devices, edge nodes, and the cloud in a swarm environment?***

   We recommend the establishment of:

   - Data Products: A footprint of the metadata associated to a dataset/data source that includes information about lineage, sovereignty, size, format, and other parameters that allow to check the consistency across the data chain to avoid manipulation and losses.

Funded by the European Union

Project funded by

- Apply immutability commodities such as IOTA Tangle, that enables the registry in a Direct Acyclic Graph every time that is required in the data chain (via API transactions).
- The federation of nodes, so that data can be accessed ubiquitously across the continuum.

3. ***What are the key challenges in achieving seamless interoperability and orchestration among heterogeneous devices and nodes in a swarm continuum architecture, and how can they be addressed?***

   To our understanding, the key challenges are:

   - A uniform and appropriate description of the workloads to be orchestrated.
   - Maintenance of a replicable and reliable status of the swarm.
   - Select a data exchange format that can be easily accessed, used, interpreted and converted into from disparate languages and platforms.
   - The capacity to encompass a wide range of devices: IoT (not Linux-based, diverse microprocessors and programming frameworks…), edge (MEC; fog…) and cloud; and have them working together.
   - Difficulties in network routing (public IPs unavailability, tunnels, cloud-native network…).
   - Lack of standardization and uniformity in data exchange mechanisms.

5.2 ANALYSIS OF DR. USMAN WAJID

Dr. Usman Wajid is the Director of ICE and the technical manager of the sister project ENACT, which is also part of the Cognitive Computing Continuum cluster. Due to his expertise, we believe he is an ideal candidate to respond to questions related to cognitive computing and programming models.

1. ***How do current programming models handle the integration of cognitive computing capabilities, like reasoning and decision-making, into distributed systems?***

   Current programming models support the integration of cognitive computing capabilities in distributed systems through a combination of techniques, such as:

   - Artificial Intelligence Techniques: Distributed systems can leverage AI and machine learning techniques for intelligent and dynamic decision-making in different scenarios. Existing frameworks like TensorFlow and PyTorch offer distributed training capabilities, which allow AI models to be trained across multiple nodes of the distributed system. The trained models can be deployed to carry out cognitive tasks such as reasoning, classification, and predictions. Similarly, federated learning techniques can be used to train AI models across decentralised data sources without moving data to a central node. In federated learning, each node performs local decision-making and shares only model updates, which can be used for reasoning and cognitive tasks while maintaining privacy and reducing latency.

Funded by the European Union
Project funded by

- Knowledge Representation and Reasoning: KRR techniques can be applied to represent distributed knowledge as ontologies, which can be used by distributed systems to reason over large or decentralised datasets. Tools like Apache Jena can enable reasoning over distributed data. Moreover, distributed systems can implement rule-based reasoning engine, such as Drools to allow for distributed reasoning across different components of the distributed system. Drools allow the processing of programming logic or rules to be distributed across various nodes to handle real-time decision-making in a decentralised manner.

- Multi-Agent Systems: MAS techniques enable the development of distributed systems by modelling system functionalities and/or cognitive tasks as multiple intelligent agents. In the MAS-based distributed system, each agent can represent an independent entity capable of reasoning and decision-making. These agents can interact with each other to solve complex problems and make decisions in a collaborative or competitive environment. Frameworks like JADE (Java Agent Development Framework) and ROS (Robot Operating System) are often used for developing multi-agent systems in distributed setups.

- Edge and Fog Computing: In edge or fog computing environments, cognitive tasks (e.g., decision-making based on real-time data) are handled closer to the data source (e.g., IoT devices). This reduces latency and allows for local data processing and reasoning, which in turn reduces the resource requirements and latency on centralised processing and reasoning. In the edge and fog environments, distributed systems and AI models can be deployed to perform cognitive functions on different edge nodes, and these nodes can collaborate with others in the system to reach decisions.

- Microservices Architectures: Microservice and serverless system architectures can support cognitive computing capabilities in distributed systems by enabling different functions to be executed on demand. The on-demand execution of distributed functions allows for scalable and distributed execution of cognitive tasks, like decision-making, which can be associated with different microservices that can be deployed across a distributed (edge or cloud) infrastructure.

- Event-Driven Architectures: Distributed systems can integrate with event-driven architectures to support distributed reasoning and cognition in different components. In the event-driven architecture, when certain conditions are met, the decision-making functions can be triggered to process data and perform relevant actions. Open-source frameworks, such as Apache Kafka or AWS Lambda provide support for incorporating event-driven architectures in distributed systems.

The aforementioned techniques and frameworks allow for developing distributed systems with scalable, real-time, and intelligent decision-making capabilities, which are considered vital for many application domains.

2. ***What challenges do developers face when designing cognitive applications that can adapt and learn over time, especially in dynamic environments?***

   Designing cognitive applications that can adapt and learn over time, especially in dynamic environments, presents several challenges for developers. These challenges span technical, architectural, and operational dimensions. One such challenge faced by the developers relates to the limitation of existing application programming models

Funded by the European Union

Project funded by

to support the development of dynamic applications. Existing application programming models such as Multi-Agent Systems, microservice or RESTful model or Cloud Application Programming model describe the architectural aspects of programming distributed application. However, these models do no provide basic constructs or ready to use functionalities that can help developers in handing dynamism in the environment or the cognition and adaptation capabilities of the application. For example, the Multi-Agent Systems community can make use of standard interaction protocols to enable communication between distributed agents. However, enabling dynamic behaviours and handling uncertainty is left on individual programmers to develop in their own way. The availability of such standardised yet reconfigurable functionality can allow programmers to efficiently address the need for dynamism arising from uncertainty in the operating environment or the lack of relevant data.

Moreover, in dynamic environments, data sources are often changing, incomplete, or noisy. Cognitive applications must be able to handle missing or uncertain data and still make reasonable decisions. Ensuring that the system can cope with this uncertainty without making incorrect inferences therefore represents a significant challenge. The lack of programming support or reconfigurable functionality in the existing programming models means the programmers need to program the functions from scratch, which not only adds to programming overhead but also results in non-standardised approaches for addressing the same or similar problems. For example, cognitive systems in dynamic environments often need to make decisions in real-time and therefore developers must design new algorithms that can quickly adapt and update their models without introducing unacceptable delays. This requires efficient processing of incoming data, which can be challenging in environments with high throughput or real-time constraints. The availability of such algorithms and training datasets can ease the burden on application developers and also pave the way for developing standardised applications. The availability of standard, model driven, functionality would also help in addressing the complexity of Designing Self-Adaptive Systems.

Similar support for incorporating standardised security, privacy and ethical and governance principles in the cognitive application, making them accountable and compliant by design for typical concerns, such as bias, fairness, trustworthiness and other regulatory aspects.

Designing cognitive applications from scratch is a complex and multifaceted challenge. Developers must tackle issues related to data quality, real-time adaptation, scalability, security, privacy, and system stability. Additionally, ethical concerns, integration with legacy systems, and maintaining trust in the system add further layers of complexity. Addressing these challenges can be helped by developing architectural enhancements and standardised functionalities that ensure that developing cognitive applications remain efficient, reliable and compliant with relevant principles over time. In this respect, the Application Programming Model currently being developed in the EC funded ENACT project aims to address the limitations of existing frameworks by providing ready to use and reconfigurable functionalities that can be used to develop new cognitive applications or enrich existing applications with real-time adaptation functionalities.

3. ***With so many frameworks available for cognitive computing, how do you decide which programming models or tools work best for a specific use case?***

Deciding which programming models or tools for cognitive computing are best suited to a specific use case requires a detailed understanding of both the use case itself

Funded by the European Union

Project funded by

and the capabilities of the available frameworks. The selection process typically involves evaluating the technical requirements, performance considerations, and specific features offered by the frameworks in the context of the problem. Depending on the task, developers can make use of frameworks like Apache Kafka or Flink that support real-time decision-making and event-based systems that can be enriched with rule-based reasoning using Drools or Prolog. Similarly for tasks such as image classification, object detection, and image generation, TensorFlow, PyTorch, OpenCV, or Keras can provide the necessary structures and programming support.

An important aspect of developing cognitive applications is the balance performance with scalability. Scalability is particularly important in modern AI applications where there is a need to handle large datasets across distributed resources. To support such requirements, frameworks such as Apache Spark enable parallel processing across clusters of machines.

A common need in the research and innovation projects is to perform experimentation and prototyping of innovation solutions. Prototyping is also an approach that can help support the selection of suitable programming model or framework. In many cases, the best way to determine which framework works best is to build quick prototypes using several different frameworks and evaluate them against your use case's specific requirements. Performance benchmarks, ease of integration, and speed of development can be assessed during the prototyping phase. Similarly, feedback from end-users or domain experts who will interact with the cognitive system can help in the selection or approval of suitable framework. The insights of users can help identify which framework meets the use case's goals more effectively, particularly in areas like usability and interpretability.

In conclusion the right programming model or framework for a cognitive computing application depends on the nature of the use case, data characteristics, real-time processing needs, and deployment considerations. By evaluating frameworks based on task-specific capabilities, scalability, ease of use, security, and other factors, you can select the best tool to meet the performance, operational, and ethical requirements of your application. Additionally, prototyping and experimenting with multiple frameworks can provide valuable insights into their practical suitability for specific needs.

5.3 ANALYSIS OF JASON FOX, VICE-CHAIR ETSI ISG CIM

Jason Fox is Vice-Chair of the ETSI ISG CIM group responsible of the definition of the ETSI NGSI-LD API for Context Information Management enabling close to real-time (right-time) access to context/digital twin information coming from many different sources (not only IoT data sources). Due to his expertise in the Data manageability, scalability and adaptability, he is ideal to response to questions related to these topics.

Data Manageability

- **How should data be categorized and organized within the system?**

  Data entities should represent a digital twin of an asset or logical concept as found in the real world. In that regard every data entity requires a unique identifier and some form of type (or multiple type). Broadly speaking attributes of the entity can be subdivided into Properties and Relationships where a property holds the value of an attribute and a relationship forms a directional link between two separate entities.

Funded by the European Union

Project funded by

Linked data principles should apply, particularly in the case that data is shared between separate organizations. The advantage of Linked Data (such as JSON-LD) is that it allows for the attributes used internally within a system to differ in separate organizations and still offer unique URIs for classifying the meaning of the attributes in question.

Entity definitions should be broad and flat rather than deep - pragmatism should be used over a strict application of ontologies. It is easier for systems to query a single data entity with lots of attributes holding data rather than processing a series relationships and traversing the knowledge graph. This is particularly the case with complex queries involving multiple attributes simultaneously.

E.g., "Give me all Blue Cars over three years old"

Query: Entity type="Car"&color="Blue"&age>3

Vs

Query for an Entity that has attribute type which in turn has value relationship named type which has valueObject with value "Car" and the same entity has attribute which has value relationship named "color" which has valueObject with value "Blue" and an Entity has an attribute which has value relationship named "color" which has valueObject with value "Blue".

- **What metadata is required to manage the data effectively?**

For an Entity. - an id, type and temporal attributes such as createdAt and modifiedAt

For each Attribute - a type (e.g. Property/Relationship) a data type (e.g. Integer/Float/String) and for dynamic attributes, potentially temporal meta data such as observedAt. In addition, values should indicate a defined unit of measure using a predetermined set of unit codes (such as UN/CEFACT). Other well-defined meta data attributes may depend on the attribute being measured (e.g. precision, accuracy etc.) or the type of information held within the attribute (e.g. a location may require a coordinate reference datum point. Specialised data attributes should use a predetermined format for representation which can be referenced in the metadata - e.g. Quaternions for a pose element or GeoJSON for a location attribute or a langString http://www.w3.org/1999/02/22-rdf-syntax-ns#langString format for a Multilanguage attribute. Additionally meta data in the form of a cryptographic proof may be necessary to ensure consistency of data.

- **Who needs access to the data, and what levels of access are necessary?**

It is likely that separate roles will be required for readers and writers of any system. It may be the case that different users' data is stored separately from one another due to legal requirements. Such a system can be obtained using a multi-tenant system. To minimize complexity, access should ideally be at the entity level rather than at the attribute level. The number of roles and granularity within the system should be kept to a minimum.

- **What tools or interfaces are preferred for data retrieval and management?**

Access should be obtained through a well-defined interface following internationally accepted standards. To ensure ongoing flexibility such an interface itself should not

Funded by the European Union
Project funded by

dictate the security mechanisms involved - these may vary from use-case to use-case. Obviously support for the chosen standardised security mechanisms (the control plane) must be available for use across the chosen data-exchange interface (the control plane) and such mechanisms should be orthogonal to each other.

Regarding data management, again this is use-case dependent and will vary depending on whether there is a unique owner of the data in the system or multiple actors are involved across a data space. In the former case a centralised access control and identity management system is possible. In data spaces and more decentralized systems a mechanism supporting distributed trust may be required (e.g. a system based on verifiable credentials and trusted issuers).

- **What measures will be taken to ensure data accuracy and consistency?**

  A well-defined data schema can be used to pre-validate data prior to entry into the system. Consistent data models can be used to automate the creation of DataAccessObjects which will can be used to ensure entity data is complete

- **How will data validation and cleansing be handled?**

  Similar to the previous question, a data schema can be useful when checking the level or accuracy of potential input data feeding the system in a brown-field project. Potential input data can be sampled and input into a prototype system and check for accuracy in a manual or automated fashion. The input can be amended and re-run into an updated prototype to check that inconsistent data entry is kept to a minimum. Note that "dirty" input data could be funnelled through a series of automated cleansing steps and each step include validation itself prior to pushing to a "clean" system.

- **What policies and procedures should be in place for data management?**

  Policies must be agreed by a legally enforceable document before considering the necessary processes to be put in place. At a minimum all interactions will need to be logged to determine access and for use in auditing processes. The exact nature of the policies to be enforced will depend on the system to be created, and the sensitivity of the data involved.

- **How should compliance with data regulations be monitored?**

  Prior to commencing the project a compliance check should be conducted so that as far as possible compliance can be architected into the system as a whole. Obviously data (and the systems holding that data) can be classified as higher or lower risk and the efforts to maintain compliance can be concentrated on those most at risk. Any such system will need to be audited on a regular basis, since regulations change and compliance cannot be left purely in the hands of the administrator of the system. There will need to be mechanisms in place to facilitate the compliance process as necessary.

Scalability

- **What is the expected growth rate of users or data volume over time?**

Funded by the European Union

Project funded by

This will depend on the precise use case. Of particular importance is how (and whether) temporal data is to be stored since potentially every change will create a record of some sort. There needs to be a defined backup or storage policy and potentially retirement of data over time. For audit logs a rollover policy needs to be put in place.

- **Are there peak usage times that need to be considered for scalability?**

  Again, this will depend on the details of the use case. Throttling or smoothing may be required. It should be considered if they can or should be dropped if they cannot be processed. Within a microservice based infrastructure, scalable systems can be put in place to increase resources once a threshold has been reached and scale back down once a peak has passed. The mechanisms for doing this are dependent on the infrastructure provider chosen.

- **What performance benchmarks must be met as the system scales?**

  Load Testing, Stress testing and Endurance testing benchmarks. Changing the load from peak to trough (see above) and also inspecting at simulated maximum load to see how the system will break and checking consistency over time to ensure the system doesn't fall over due to resource constraints (e.g. memory leak) These tests should help define an upper bound based on current resources and therefore help to indicate when more resources are needed and identify where bottlenecks will occur first.

- **How should the system handle increased load or data volume?**

  This will depend on the details of the use case, in particular how fungible the data is. Ideally systems should be architected to be able to accept increased load, or at least fail in an acceptable and non-catastrophic manner. Reducing performance, removing subsidiary functions for periods of time and/or returning error responses to "please try later" may all be acceptable under some circumstances. Critical, time-limited and non-replayable data messages must be prioritised.

- **Are there preferences for cloud, on-premises, or hybrid solutions?**

  There can be arguments for all three solutions taking into account a multitude of factors. Cost, legal requirements of the jurisdiction in which the data must be held, a need for a highly robust high uptime infrastructure. Perceived trust in who has access to the data, the need for scaling and so on. The decision will be a payoff between the relevance of the various push factors depending on the value of the data and product involved. It may also be the case that the weight of these factors changes over time, meaning that a solution may need to transition from one preference to another.

Adaptability

- **How often do you anticipate changes in data requirements?**

  Once again this is a factor of the use-case. In general fixed requirements based on legal statue are less likely to need changes than unregulated requirements and commercial systems competing for users which should adapt according to user need.

- **What processes are in place for implementing changes to data structures?**

Funded by the European Union
Project funded by

In general, this is a function of the maturity of the use case itself. Ideally changes should be kept to a minimum, which can be done through using data structures which are already standardized within the domain. Where this cannot be achieved, data exchange should occur using an extensible generic API capable of parsing different payload types and the "type" held in the metadata so that new elements can be added to an existing system. The more that payloads can be self-documenting through the presence of appropriate metadata the better.

- **What other systems need to be integrated, and how flexible should the integration be?**

Existing brown field solutions will inevitably require integration of existing legacy systems. The key driving factor will be deciding how long those legacy systems will need to be maintained. If the legacy solution can be dropped quickly then a less reliable and less flexible integration system can be put in place since the assumption is that this solution is merely an ad-hoc interim solution, however, if it is envisaged that the legacy system will need be be maintained further then the ideal intermediate integration should be as flexible as possible allowing for multiple legacy clients to be serviced for a long period of time. A microservice based architecture can be helpful for this, allowing for different integration solutions to be accessed via different gateways.

- **How should the system adapt to changes in external data sources or APIs?**

API paths should be versioned according to software engineering best practice so that a fixed protocol can be used under a given path and a contract maintained so that data passed into that API shall conform to a given standard. Where necessary API payloads should indicate the payload version they are using and the return payload version they are willing to receive, through the use industry standard mechanisms for the API.

5.4 ANALYSIS OF TIM SMYTH

Tim Smyth is a senior developer of the FIWARE Foundation responsible of the implementation of the FIWARE Data Connector. He has high expertise in decentralized identity management as well as the use of EU Standards for data registration, data sharing, as well as federated environments. His answers in terms of swarm-wise distributed security is relevant for the CoGNETs project to define an Edge-Cloud computing decentralized security management.

- **How do you define "swarm-wise distributed security" in the context of your organization?**

We are focused on the aspect of decentralized access control and role handling using verifiable credentials to strengthen the idea of self sovereign identities. The driving force is the published data space connector component which follows the DSBA Technical Convergence recommendations.

- **What specific goals do you want to achieve with this security paradigm?**

Enabling the spread of sovereign data spaces with a standardized protocol to enable interoperability while not enforcing data planes restrictions or tying the technology down to specific use cases and data protocols.

- **What types of threats and vulnerabilities are you most concerned about?**

Funded by the European Union
Project funded by

By using established standards for cryptography and established protocols the threat of underlying logical vulnerabilities is greatly reduced. Implementation specific issues pose a higher risk but are mitigated by peer review of the code.

- **What types of data need protection, and what are the confidentiality, integrity, and availability requirements?**

  The developed components are data agnostic, so the data type requirements are to be defined by the data provider.

- **How should sensitive data be encrypted and managed?**

  The data space connector components define the protocol for handling authorization functionality but do not enforce restrictions on the data encryption and management.

- **What authentication and authorization mechanisms are preferred?**

  The FIWARE Data Connector utilizes authentication based on W3C DID with VC/VP standards and SIOPv2 / OIDC4VP protocols and authorization based on attribute-based access control (ABAC) following an XACML P*P architecture using Open Digital Rights Language (ODRL) and the Open Policy Agent (OPA). These standards were proposed due their widespread use and specific utilisation by the Data Spaces Standardization groups.

- **How should access be managed in a distributed environment?**

  Access should be granted to organisations based on decentralized identities while utilizing one or more trust anchors for managing basic participant management.

- **How should nodes communicate and collaborate to enhance security?**

  Communication between the nodes should use the described protocols for authorization and connection management while relying on strong underlying cryptographic functionality of secured network access to avoid common attack mechanisms.

- **What mechanisms are needed for nodes to detect and respond to security breaches?**

  The utilization of revocation lists for issued long-lasting tokens and the general use of short-lasting tokens enables an effective exclusion of compromised nodes.

5.5 ANALYSIS OF PROF. SOKRATIS KATSIKAS

Prof. Sokratis Katsikas is an internationally recognized researcher in cybersecurity and information security, currently serving as a Professor at the Norwegian University of Science and Technology (NTNU) and Director of the Research and Innovation Center at the Norwegian Center for Cybersecurity in Critical Sectors. With an extensive academic and research background in trust management, cryptographic security protocols, and distributed security mechanisms, he is highly qualified to provide insights on Swarm-wise Distributed Security Paradigms.

Funded by the European Union
Project funded by

- **How can swarm intelligence principles enhance distributed security mechanisms, particularly in dynamically changing environments where nodes join and leave unpredictably? (things to consider here: kinds of adaptive security measures or consensus protocols to employ, hence ensuring continuous integrity and resilience)**

  Swarm Intelligence Principles

  1. Swarm intelligence leverages decentralized, self-organized systems to solve complex problems. This principle can be applied to distributed security mechanisms to enhance resilience and adaptability.

  2. Inspired by the collective behaviour of social insects, swarm intelligence enables efficient navigation and problem-solving through local interactions. This can be used to coordinate security measures across multiple nodes.

  3. Swarm intelligence systems are inherently scalable and robust, making them suitable for dynamic environments where nodes frequently change.

  Adaptive Security Measures

  1. Utilizing real-time threat intelligence allows for continuous monitoring and analysis of network traffic to detect suspicious activities and respond promptly. This proactive approach helps in identifying and mitigating threats before they cause significant damage.

  2. Implementing machine learning and AI can analyse vast amounts of data to identify patterns and anomalies that may indicate potential security threats. These technologies enable adaptive security measures to evolve with changing threats.

  3. Managing Non-Human Identities (NHIs), which are machine identities, can provide end-to-end protection by proactively identifying and mitigating security risks. This approach enhances visibility, control, and compliance in dynamic environments.

  Consensus Protocols

  1. Byzantine Fault Tolerance (BFT)-based consensus protocols ensure that the system can reach an agreement even if some nodes are faulty or malicious. This is crucial for maintaining integrity in unpredictable environments.

  2. Dual-Mode Consensus Protocols, such as Flexico, offer both fast and backup modes for consensus. The fast mode operates under ideal conditions, while the backup mode takes over in non-ideal conditions, ensuring continuous operation without starting over.

  3. Federated Learning allows for decentralized training of models while preserving data privacy. This approach can be integrated with swarm intelligence to enhance decision-making processes in distributed security mechanisms.

- **What strategies or frameworks do you recommend for trust management and authentication in swarm-based security models? (nodes potentially may be operating in adversarial environments we need to consider: ZTA principles and**

Funded by the European Union

Project funded by

identity management to maintain security while minimizing computational overhead)

Trust Management Strategies

1. The DualTrust Model focuses on monitoring the trustworthiness of autonomic managers rather than individual swarming sensors. It enhances scalability and protects the swarm by ensuring that only trusted managers can make critical decisions.

2. In Reputation-Based Systems nodes can maintain a reputation score based on their behaviour and interactions. Trust decisions are made based on these scores, which helps in identifying and isolating malicious nodes.

3. Continuously performing Behavioural Analysis of nodes to detect anomalies and potential threats helps in identifying compromised nodes and taking corrective actions promptly.

Authentication Frameworks

1. Blockchain-Based Identity Management ensures a decentralized and tamper-proof system. Each node's identity is securely stored on the blockchain, and authentication is performed using cryptographic techniques.

2. Physical Unclonable Functions (PUFs) generate unique identifiers for each device, making it difficult for attackers to clone or impersonate devices. This approach enhances physical security and ensures trustworthy communications.

3. Implementing Group Authentication Protocols allows for efficient and secure authentication of multiple nodes simultaneously. This reduces computational overhead and ensures that only authenticated nodes can participate in the swarm.

Zero Trust Architecture (ZTA) Principles

1. Always authenticate and authorize based on all available data points, including user identity, device health, and location.

2. Limit user and device access to only what is necessary for their roles. This minimizes the attack surface and reduces the risk of unauthorized access.

3. Design the system with the assumption that breaches will occur. Implement continuous monitoring, micro-segmentation, and end-to-end encryption to detect and respond to threats quickly.

Identity Management in Adversarial Environments

1. Implement adaptive access controls that adjust based on the current threat level and context. This approach ensures that access permissions are dynamically adjusted to mitigate risks.

2. Use AI and machine learning to enhance identity verification processes. These technologies can analyse patterns and detect anomalies that may indicate potential threats.

Funded by the European Union
Project funded by

3. Implement deepfake detection mechanisms to prevent identity fraud and ensure the authenticity of communications.

- **What are the primary challenges in securing data exchange and decision-making processes within autonomous swarms and mitigation actions?**

  Challenges

  1. Dynamic Network Topology: The constantly changing structure of the swarm makes it difficult to maintain secure communication channels.

  2. Resource Constraints: Limited computational power and energy resources in individual nodes can hinder the implementation of robust security measures.

  3. Scalability: As the number of nodes increases, the complexity of securing communications and decision-making processes grows exponentially.

  4. Interference and Jamming: Swarms are susceptible to interference and jamming attacks, which can disrupt communication and coordination.

  5. Data Integrity and Authenticity: Ensuring that data exchanged between nodes is not tampered with and comes from legitimate sources is crucial.

  6. Latency and Real-Time Processing: Real-time decision-making requires low-latency communication, which can be challenging to secure without introducing delays.

  Mitigation Actions

  1. Implementing lightweight encryption protocols, such as SIMON and SPECK, can provide security without significant computational overhead.

  2. Using adaptive communication protocols that can adjust to changing network conditions helps maintain secure and efficient data exchange.

  3. Employing decentralized trust management systems, such as reputation-based models, ensures that trust decisions are made collectively by the swarm.

  4. Leveraging AI and machine learning for anomaly detection and adaptive security measures can enhance the swarm's ability to respond to threats in real-time.

  5. Utilizing multi-agent reinforcement learning frameworks for decision-making processes can optimize coordination and resilience against attacks.

  6. Implementing redundancy and self-healing mechanisms ensures that the swarm can recover from node failures and maintain secure operations.

5.6 ANALYSIS OF PROF. PANAGIOTIS TRAKADAS

Panagiotis Trakadas is an Associate Professor at the National and Kapodistrian University of Athens and the Director of Fourdotinfinity, an SME specializing in cutting-edge technological solutions. With extensive expertise in the Federated Learning research field, his knowledge

Funded by the European Union
Project funded by

is demonstrated through his active participation in several EU-funded research projects, including ENACT and CyclOps. Given his experience and contributions to these initiatives, we believe he is exceptionally well-suited to address inquiries related to Federated Learning Mechanisms.

- **What types of AI models (e.g., CNNs, DNNs) are most suitable for FL in IoT/ Edge environments, and why?**

  Federated Learning involves the exchange of model parameters (e.g. weights and biases) between a central server (the model aggregator) and clients (e.g. IoT/Edge devices). In that sense, FL can be applied to any type of model that needs to be trained (e.g., CNNs, DNNs, etc.).

  Therefore, in the IoT/Edge scenario, the only limitations as to which model to use, depends mostly on the available local resources which are typically limited. Deploying large DNNs in FL on IoT/Edge devices is challenging for this reason, not because certain model types cannot be trained in FL.

  Having said that, FL may not be suitable in case of models sensitive to data distribution, i.e. models that assume that training data is independently and identically distributed (IID). In FL, data across clients is often non-IID, i.e. data distributions can vary significantly between clients. Models that are sensitive to such distribution disparities may struggle to converge or perform optimally in FL settings.

- **Which key performance indicators (KPIs) should be used to evaluate the success of the FL system (e.g., accuracy, latency, convergence time, resource usage)?**

  Main motivation for FL in our experience is (1) the preservation of privacy and security of data, and (2) the efficiency of bandwidth utilization across the network. Both are enhanced with FL by sending only the parameters instead of the actual data in the network.

  Besides the above, our results until now show that other important KPIs include:

  - Accuracy/Loss
  - Total training time (i.e. local training + data transfer + aggregation).
  - Convergence time (i.e. time required to perform the necessary iterations to reach the desired accuracy)
  - Total energy consumption (clients + server)
  - Resource usage (to ensure the FL process is sustainable on clients since these typically have limited resources)

- **What types of IoT/Edge devices should be considered for participation in the FL process, and how each of them affects the learning process?**

  The choice of IoT and edge devices for Federated Learning does affects aspects such as training efficiency, communication and energy consumption. High-power edge devices enable fast local training and can handle complex models efficiently, making them well-suited for FL. However, they do consume more energy and may

Funded by the European Union
Project funded by

not always be available in distributed settings. In contrast, low-power IoT devices, such as sensors and microcontrollers, have limited computational capacity, often leading to straggler effects that slow down training, especially in Synchronous FL setups. Solutions include Asynchronous FL or selective client participation to mitigate this issue in diverse environments.

5.7 ANALYSIS OF EXPERT X ABOUT GAME OPTIMIZATION STRATEGIES

This expert serves as a professor with an extensive academic and research background in Game Theory.

- **How do Game Optimization Strategies balance the interests of attackers and defenders in the trade-off between Security and Quality of Service (QoS), ensuring security while optimizing system performance?**

Game Optimization Strategies Balance Security and QoS by modelling attacker-defender interactions using game theory. Key approaches include:

1. Trade-off Management – Stronger security reduces QoS (e.g., latency), while higher QoS may expose vulnerabilities. The goal is optimal resource allocation.

2. Game-Theoretic Models – Zero-sum games (minimizing worst-case losses), Stackelberg games (proactive defence), and Bayesian games (handling uncertainty).

3. Optimization Techniques – Adaptive security policies, resource-aware defences, and multi-objective optimization ensure minimal QoS impact while maintaining security.

- **What are the common game optimization methods used in the security-QoS trade-off? How do these methods adjust strategies to adapt to different attack models and system requirements?**

Common game optimization methods in the security-QoS trade-off include:

1. Zero-sum Games: Where the gain of one party (e.g., attacker) is the loss of another (e.g., defender). The system optimizes to minimize losses while managing QoS.

2. Bayesian Games: Used when attackers' intentions are uncertain, allowing for adaptive security measures based on probabilities of different attacks.

3. Stackelberg Games: The defender moves first, setting a security strategy, and the attacker responds. This helps in proactive defence planning with minimal QoS impact.

4. Evolutionary Games: These adjust strategies over time based on continuous learning from system performance and attack patterns.

Funded by the European Union

Project funded by

- **How can game optimization strategies enhance security without significantly degrading QoS in real-world network systems or intelligent IoT environments? Are there any typical application cases demonstrating their effectiveness?**

Game optimization strategies enhance security without significantly degrading QoS by dynamically adjusting security measures based on threat levels and system requirements. For example:

1. Adaptive Defence: Security mechanisms (e.g., firewalls, intrusion detection) are applied selectively, increasing only when a threat is detected, minimizing resource usage and QoS impact.

2. Lightweight Security: In IoT environments, lightweight encryption and anomaly detection are used to secure devices without overloading them.

3. Multi-objective Optimization: Balancing security and QoS through techniques like Pareto optimization, ensuring neither is overly compromised.

Application cases include:

1. Smart Home Systems: Using adaptive security protocols that adjust based on detected attack risks while maintaining QoS for devices like thermostats or security cameras.

2. Cloud Networks: Dynamically allocating security resources (e.g., threat monitoring) to critical areas, avoiding unnecessary performance hits on non-critical parts of the system.

Funded by the European Union

Project funded by

6 ARCHITECTURE DESCRIPTION

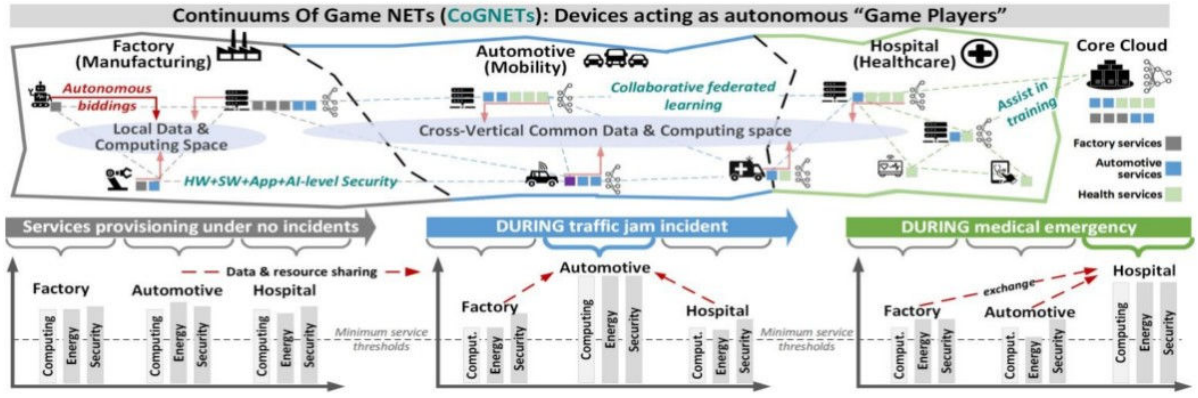
6.1 PROJECT MOTIVATION, CHALLENGES AND OBJECTIVES

The purpose of the CoGNETs framework is to develop a Middleware Framework that empowers IoT, Edge, and Cloud devices, playing a crucial role in creating a cohesive ecosystem. This framework will autonomously organize a dynamic IoT-to-Cloud swarm continuum, facilitating seamless communication and collaboration between devices at various levels. By leveraging this architecture, optimal data processing can be achieved, ensuring that information flows efficiently across the network. Additionally, the framework will enable seamless service provisioning, allowing devices to respond quickly to user demands and environmental changes. This holistic approach not only enhances operational efficiency but also paves the way for innovative applications in smart environments.

To this end, the project's ambition is to foster a shift toward an on-demand, opportunistic approach, embracing a model that allows for flexibility and responsiveness to emerging needs. This paradigm enables a continuum that operates without predefined orientations, allowing for fluidity in decision-making and resource allocation. Within this framework, a dynamic swarm continuum emerges, characterized by the integration of both fixed and temporary infrastructural elements. This duality facilitates a more adaptable system—capable of swiftly responding to changes, optimizing resources in real-time, and fostering innovation through collaborative efforts.

Several key observations must be considered in the definition of the architecture. The first is the analysis of current AI technologies, which can only partially support the potential of an autonomous and dynamic computing paradigm. While advancements have been made, the integration of concepts such as **self-organization** and **collaborative learning** —the second key observation— among running AI processes on IoT devices remains a significant challenge. These concepts aim to enhance the efficiency and adaptability of AI systems, enabling them to learn from one another and self-organize in response to changing environments. However, current limitations in processing power, data interoperability, and real-time communication hinder the full realization of this vision, highlighting the need for further innovation and development in the field.

Figure 6: CoGNETs approach and targeted concept



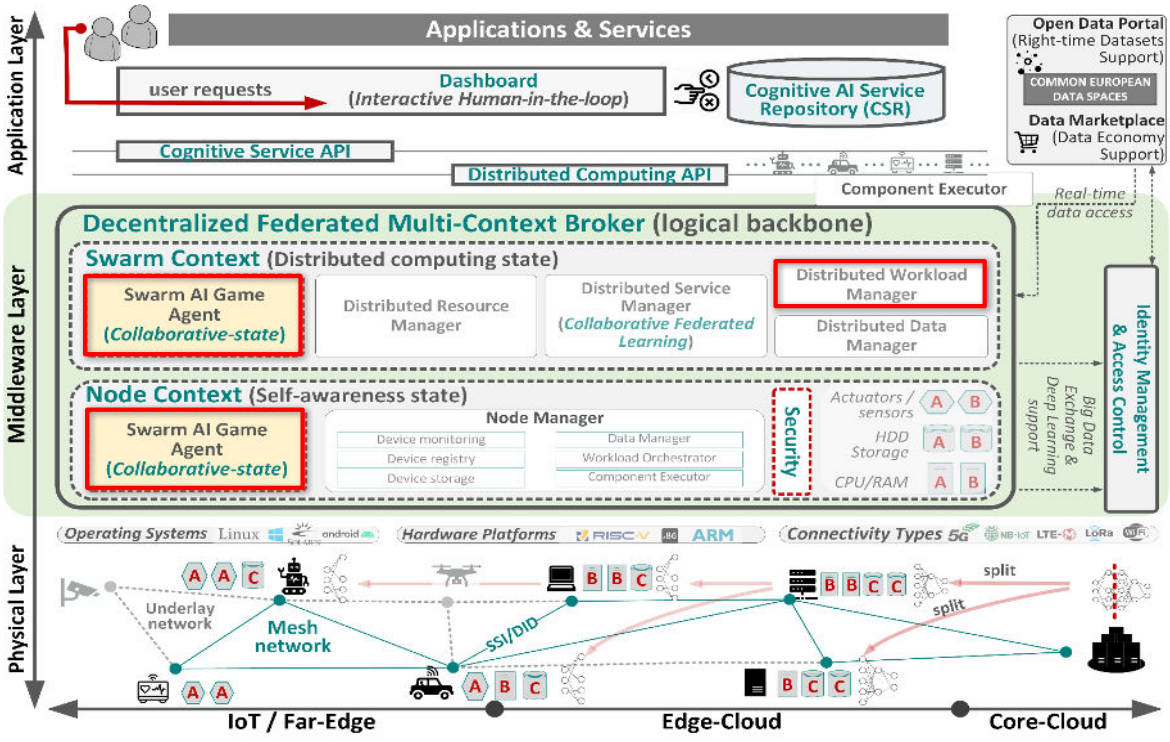
Funded by the European Union
Project funded by

Taking into account these key observations, the next point to analyse is what needs to be implemented within the CoGNETs platform. This includes developing intelligence as an integral component of each device — an essential functionality that supports every activity, process, and decision-making operation. This approach encompasses a wide range of capabilities, from the self-organization of on-board datasets and hardware resources to collaborative learning within mesh local clouds. By embedding intelligence directly into devices, they can autonomously manage their data and optimize resource usage, thereby enhancing efficiency and responsiveness. Furthermore, through collaborative learning, devices can share insights and experiences, fostering a networked intelligence that adapts and evolves in real time—ultimately leading to more informed decision-making and improved overall performance.

These ideas will be translated into a set of specific objectives for implementation within the CoGNETs project. The **first objective** consists of building intelligent game agents that enable self-organization and decision-making capabilities at the Edge. This also involves leveraging feature-based heterogeneity models and asymmetric competitive games to optimize data and resource sharing at the device level. These agent systems can autonomously assess IoT, Edge, and Cloud devices, determining how to effectively share data and resources to enhance overall performance while meeting diverse end-user requirements, such as speed, data rates, and accuracy.

The ambition is to autonomously maximize the data computing capacity of the IoT-to-Cloud continuum, while ensuring that energy efficiency, security, and sustainability are not compromised. A key innovation in this approach is the introduction of novel versions of Decentralized Network Intelligence, which rely on self-adaptive reasoning and knowledge acquisition—allowing devices to autonomously participate in common computing tasks. This strategy aims to improve service effectiveness while reducing energy consumption and minimizing vulnerabilities. The verification of these concepts will be carried out through Tasks T3.1–T3.2 (WP3) and documented in Deliverables D3.1a–D3.1b.

Figure 7: CoGNETs first objective

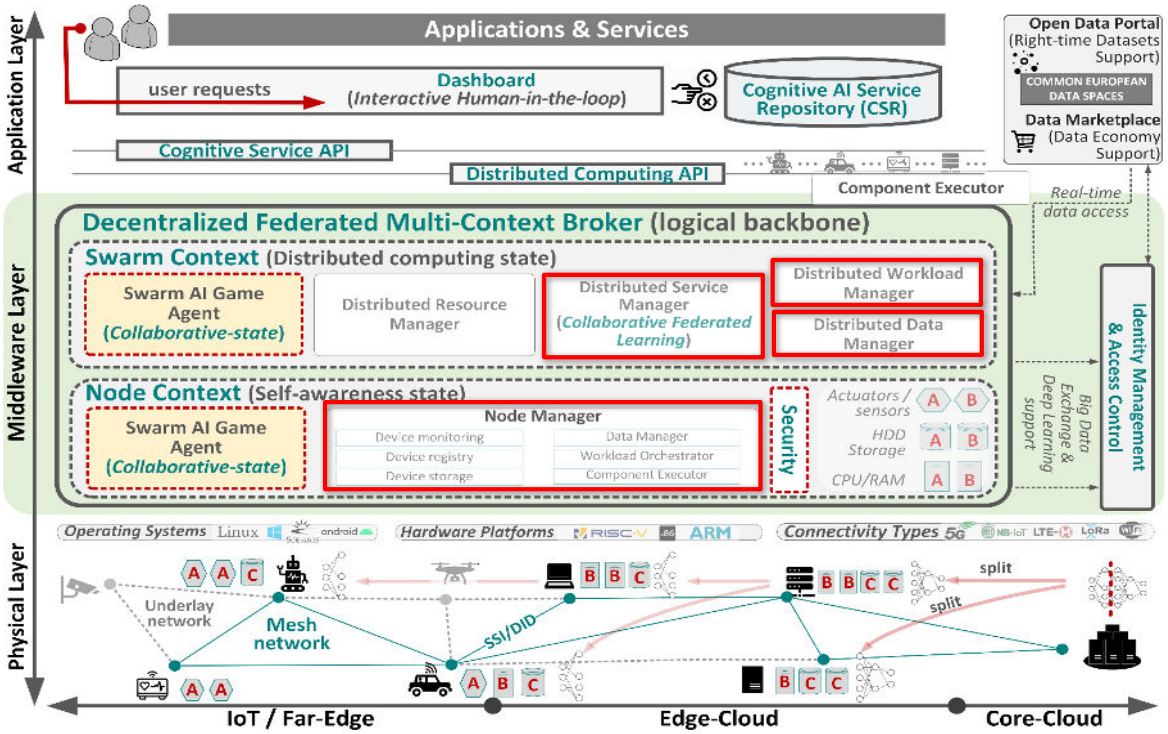


Funded by the European Union

Project funded by

The **second objective** deal with the building of a distributed middleware framework for co-ordinating dynamic IoT-to-Cloud swarms of autonomous data processing. For this purpose, we will utilize analytical model distribution functions and delocalized federated multi-Context Broker architectures based on the functionalities offered by the ETSI NGSI-LD Brokers. The baseline involves developing a middleware that interprets Game Agent functions as integrated services, leveraging Distributed Directed Acyclic Graph (DDAG) and Distributed Ledger Technology (DLT), complemented by a multi-context Broker Runtime that executes Game functions in harmony with core data orchestration and federated learning routines. The implementation of DLT will depends on the performance requirements of the platform which will be translated to similar technology to resolve any limitation of requirements in this aspect. Our ambition is to enhance compute-connect functionalities beyond existing operating systems by creating middleware that fully harnesses available data and computing resources through locally organized swarms. A key innovation lies in enhancing the ETSI NGSI-LD Broker for dynamic IoT-to-Cloud swarms via decentralized games, extending the traditional DAG DLT model to a novel DDAG DLT framework that supports improved self-verified data structures. Additionally, we will introduce a new Context Registry for direct data discovery and registration within highly distributed swarms. Verification of these advancements will be carried out through tasks T4.1-T4.4 (WP4) and deliverables D4.1a-D4.1b and D4.2a-D4.2b.

Figure 8: CoGNETs second objective

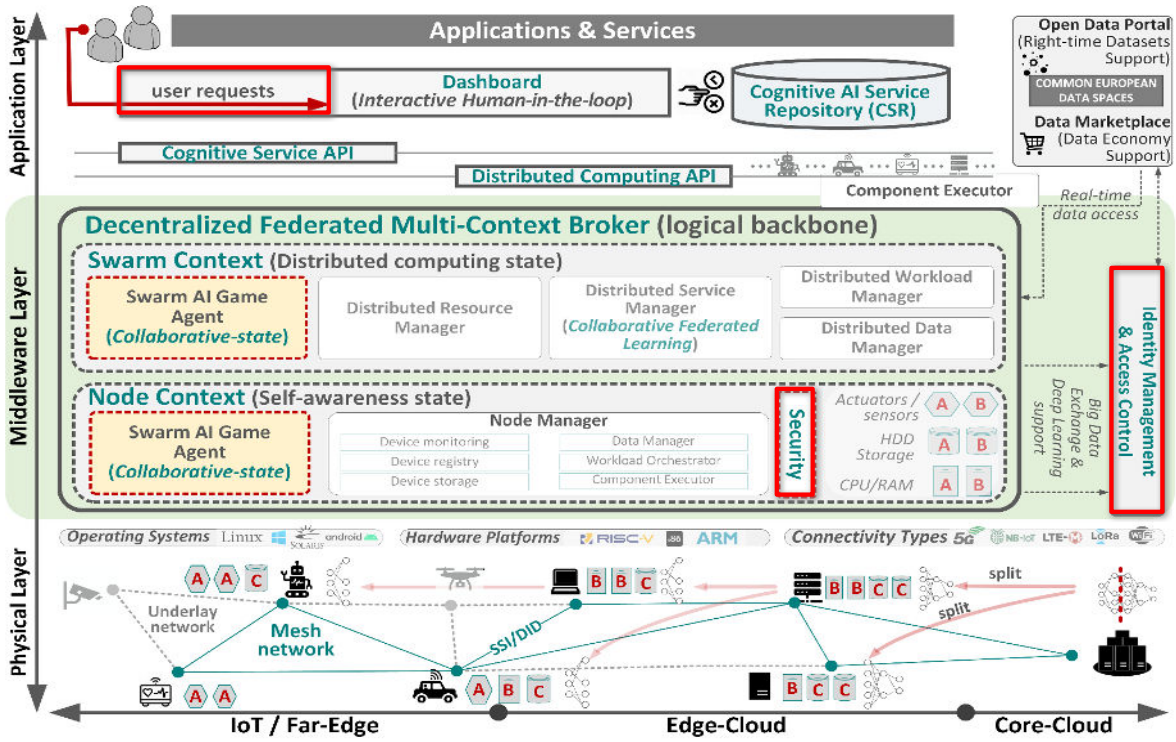


The **third objective** deal with the construction of an end-to-end security, identity, privacy, and resilience mechanisms that address swarm-centric threats across all system, application, and AI levels. We will employ low-overhead Self-Sovereign Identity (SSI) and Decentralized Identifiers (DID) secured by RISC-V architecture and adversarial shielding. The baseline involves enhancing our middleware with embedded security mechanisms that are divided into isolated domains. This integration will include SSI/DID, adversarial AI shields, anomaly detection, and rapid recovery protocols to fortify swarm security. Our ambition is to bolster the resilience of dynamic swarm continuum against evolving threats by developing RISC-V processor hardware that guarantees identity and security, effectively shifting swarm security

Funded by the European Union
Project funded by

responsibilities from end terminals to Edge and Cloud devices while ensuring end-to-end data confidentiality and integrity. A key innovation is the combination of hardware-assured end-to-end identity and security guarantees with automated security and privacy management for individual devices within IoT-to-Cloud swarms. This comprehensive approach will encompass message-level, transport-level, service-level, application-level, and node-level security techniques, facilitating automatic limitation, detection, and recovery from device compromises. Verification of these advancements will be conducted through tasks T4.1-T4.4 (WP4) and deliverables D4.1a-D4.1b and D4.2a-D4.2b.

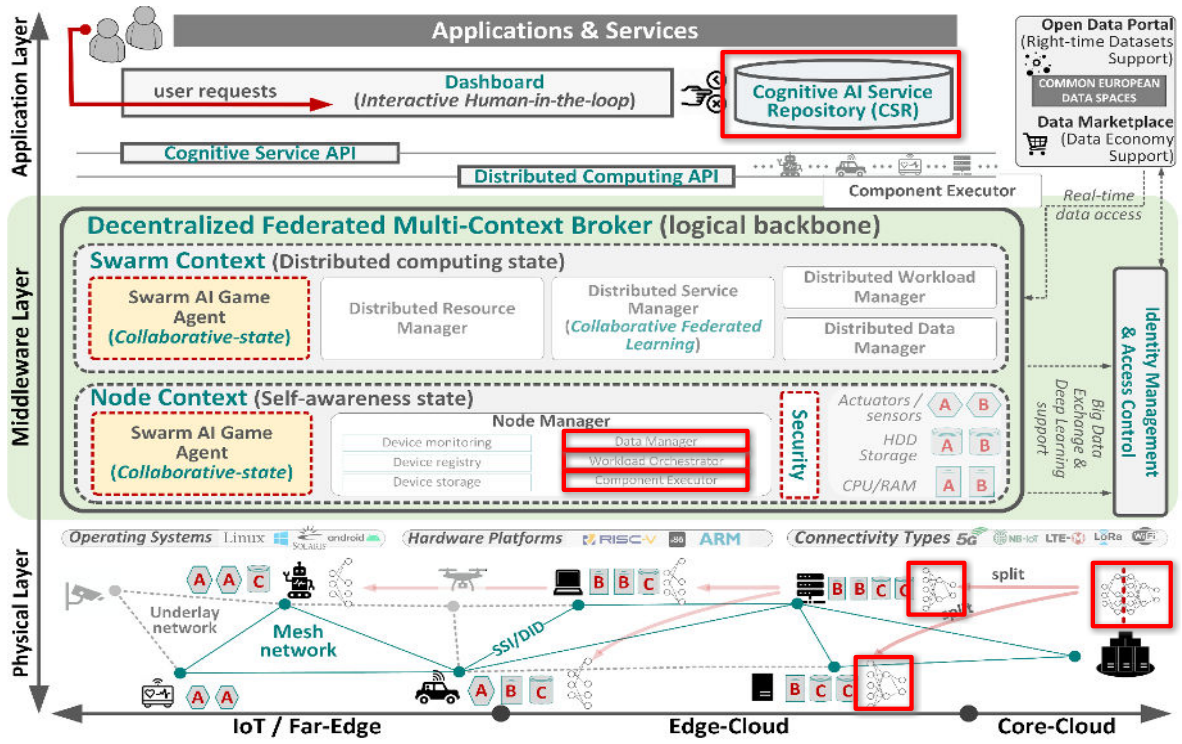
Figure 9: CoGNETs third objective



In the **fourth objective**, the CoGNETs platform builds a collaborative federated learning mechanisms that enhance AI service response locally while leveraging Edge-Cloud resources to improve training accuracy. The idea is the use of the pruning/splitting technologies for the neural network to be adopted by the Pilot Use Cases. The baseline involves enhancing the middleware's service provision through Collaborative Federated Learning (CFL) techniques, which coordinate AI model training by employing layer pruning and splitting, all while relying solely on local raw data. Our ambition is to develop a middleware that is Edge-responsive and resilient to cognitive service requests, effectively utilizing the Cloud to support training and ensure guaranteed service accuracy. A key innovation of our approach is the balance between centralized and decentralized learning, achieved by splitting Convolutional Neural Network (CNN) and Deep Neural Network (DNN) model layers across multiple IoT, Edge, and Cloud devices, thus eliminating the necessity for explicit data sharing. This method not only enhances service effectiveness but also safeguards user data privacy and improves energy efficiency. Verification of these advancements will be executed through tasks T3.3-T3.4 (WP3) and deliverables D3.2a-D3.2b.

Funded by the European Union
Project funded by

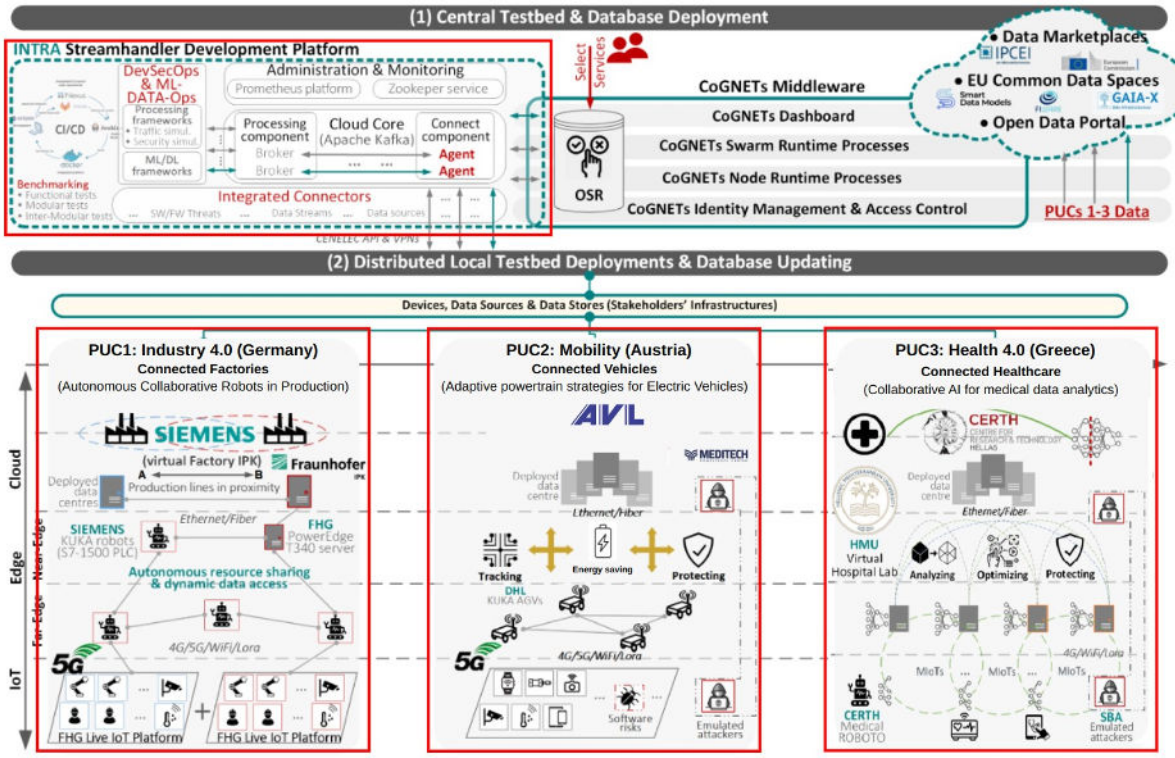
Figure 10: CoGNETs fourth objective



As a **fifth objective**, the CoGNETs platform has to develop a TRL5 testbed deployment that integrates the outcomes of objectives #1–#4 and validates its proof of concept across emerging sectors such as Industry, Mobility, and Health. We will implement the IoT-to-Cloud dynamic swarm computing paradigm in a realistic setting. This will be achieved by adopting a joint DevSecOps and ML/Data-Ops methodology, ensuring proper documentation, integration, and evaluation of impacts on the supply chains within these sectors. Our ambition is to establish an operational IoT-to-Cloud testbed infrastructure that facilitates the development, integration, and validation of future computing and connecting dynamic continuum, alongside their cognitive services and integrated security models. A key innovation of this initiative will be the validation of immersive EU vertical applications within realistic supply chain contexts, focusing on three primary use cases: PUC1 – Manufacturing (Industry 4.0), PUC2 – Mobility (Automotive), and PUC3 – Health (Health 4.0). Verification of these efforts will be conducted through tasks T5.1-T5.4 (WP5) and deliverables D5.1, D5.2, and D5.3a-D5.3b.

Funded by the European Union
Project funded by

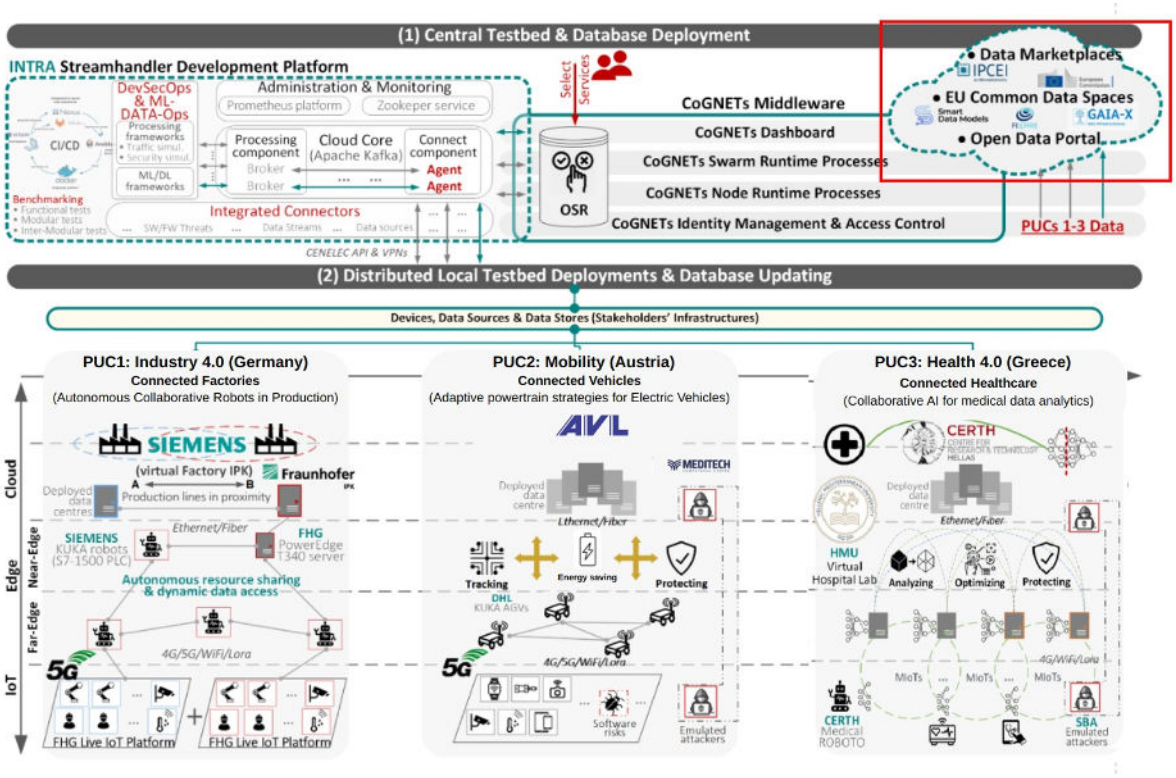
Figure 11: CoGNETs fifth objective



Finally, the **sixth objective** is focused on the identification of the EU social, ethical, legal, and privacy policy aspects of the CoGNETs system, where we aim to ensure its promotion to the Important Projects of Common European Interest (IPCEI) and Digital Europe Programme (DEP), as well as to national and international academic and industrial communities. The baseline involves making the CoGNETs middleware and its testbed infrastructure accessible to EU and international researchers, practitioners, and stakeholders, fostering collaboration and knowledge sharing. Our ambition is to position our solution as a transformative force in the EU data and computing technology landscape, enabling the development of more dynamic, scalable, interoperable, and energy-efficient systems with secure and trusted services. A key novelty of our approach is the creation of a productivity-focused IoT-to-Cloud solution that prioritizes people and their work needs, aiming to streamline the future network for greater accessibility and ease of use. Verification of these objectives will be conducted through tasks T1.4 (WP1) and T6.1-T6.4 (WP6), alongside deliverables D1.1-1.2 and D6.1x, D6.2x, D6.3x, D6.4x.

Funded by the European Union
Project funded by

Figure 12: CoGNETs objective 6



The application of these objectives go to the proper identification of the CoGNETs logical building blocks that we introduce in the following section together with the high level communication process between them in the section 6.3 and the complete list of requirements of them in section 7.

6.2 INTRODUCTION OF COGNETS LOGICAL BUILDING BLOCKS

This subsection is focused on the brief definition of each of the CoGNETs building blocks that we have identified in the project with the idea to provide a clear overview of their purpose to facilitate the understanding of the following sections in the document. We have identified a total of nineteen logical building blocks and we have classified them into four groups:

- **Application Layer** contains all logical building blocks that are connected to the final users of the CoGNETs platform. Application Layer includes three identified logical building blocks, Dashboard, Cognitive AI Service Repository, and DevOps platform, which are described as:

  - **Dashboard (UI)**, the dashboard will grant users the ability to request the execution of a CoGNETs AI module within a Pilot Use Case (PUC). Upon receiving a request, the system will translate this request to the Distributed Service Manager, which will initiate the process to execute the CoGNETs AI module. Additionally, a user-friendly interface will be provided to visualize the status of the CoGNETs AI module execution, allowing users to monitor progress and outcomes in real-time.

Funded by the European Union
Project funded by

- **Cognitive AI Service Repository (CSR)** facilitates the articulation and storage of CoGNETs AI service modules and submodules, including comprehensive metadata descriptions for each algorithm. It will establish a connection to the Distributed Service Manager (DSM), supporting both classical Convolutional Neural Networks (CNN) and Deep Neural Networks (DNN), as well as Q-Learning versions for Collaborative Federated Learning. To effectively manage these algorithms, it is essential to define a corresponding Data Model that accurately represents both CNN/DNN and Q-Learning frameworks, utilizing a Distributed Directed Acyclic Graph (DDAG) structure to ensure clarity and efficiency in data representation and processing.

- **DevOps and MLOps Platform (DOP)**. DevOps is a practice that integrates software development and IT operations to enhance collaboration, streamline processes, and improve application delivery through automation and continuous integration and deployment (CI/CD). MLOps extends these principles to the machine learning domain, focusing on the end-to-end lifecycle of AI/ML models, from development and training to deployment and monitoring. It emphasizes collaboration among data scientists and engineers, automation of workflows, version control for datasets and AI/ML models, and continuous performance monitoring to ensure model accuracy over time. Together, DevOps and MLOps create a cohesive framework that enables organizations to deliver high-quality software and insights more efficiently and reliably. CoGNETs adopts these principles to create a platform that facilitates the management of the AI modules and submoludes, taking into account both training and execution phase of them.

- **Middleware Layer (Swarm Context)** includes all the logical building blocks related to the Swarm cloud environment and the deployment of the AI modules and submodules to be executed in the Edge environment. Middleware Layer (Swarm Context) compounds five logical building blocks as described below:

  - **Swarm AI Game Agent (SGA)** plays a crucial role in executing key game functions such as "Pricing," "Bidding," and "Auctioning." By effectively managing these processes, the agent interprets the game results into actionable network functions. This transformation empowers each device to operate as an autonomous "Game Player," enabling them to actively participate in dynamic swarms. As a result, devices can collaborate and adapt in real-time, optimizing their interactions and enhancing overall network performance in a competitive environment.

  - **Distributed Resource Manager (DRM)** is pivotal in establishing the Trust List of Nodes, ensuring a secure and reliable network environment. It facilitates Decentralized Identity Management and the safeguarding of Identity Secrets for newly added nodes, which is essential for maintaining trust and security. Additionally, the DRM coordinates the initialization of these nodes to deploy various requested services, including Device Monitoring (NDMo), Device Registration (NDR), Device Storage (NDS), Data Management (NDM), Workload Orchestration (NWO), and Component Execution (NCE). By overseeing these processes, the DRM ensures that nodes are correctly configured and operational, enabling efficient service delivery within the network.

  - **Distributed Service Manager (DSM)** is designed to establish the functionalities of Swarm and Node Contexts, playing a vital role in the overall network architecture. It is responsible for synchronizing workload and data activities through the Distributed Directed Acyclic Graph (DDAG) with the various nodes, ensuring

Funded by the European Union
Project funded by

seamless communication and coordination. To achieve this, the DSM must maintain a connection with each of the Node Managers, allowing for effective oversight and management of distributed resources. This interconnectedness enables the DSM to optimize performance, enhance reliability, and facilitate the dynamic operation of services across the network.

- **Distributed Workload Manager (DWM)** is tasked with the critical role of assigning various AI modules to specific nodes, guided by the results generated by the Swarm Game Intelligent Agent. By analysing these results, the DWM ensures that each node is optimally utilized, enhancing efficiency and performance across the network. This targeted allocation of AI resources allows for more effective processing and decision-making, ensuring that the capabilities of the nodes are aligned with the demands of the swarm environment. Through its strategic assignments, the DWM contributes significantly to the adaptive and responsive nature of the distributed system.

- **Distributed Data Manager (DDM)** plays a crucial role in maintaining the integrity and relevance of data within the network. It is responsible for updating the results of the execution of the AI module and distributing this information to the rest CoGNETs nodes, ensuring that all participants have access to the latest insights. Additionally, the DDM must define synchronization rules within the data plane to facilitate timely updates of the outputs generated by the nodes. By establishing these rules, the DDM ensures consistency and coherence in data handling, enabling nodes to operate with the most current information and enhancing overall network performance. For this purpose, DDM will be based on the Distributed Operations defined by ETSI NGSI-LD API to facilitate this operation.

- **Middleware Layer (Node Context)** includes the logical building blocks related to the execution and monitoring of the AI modules and submodules on the Edge environment. Middleware Layer (Node Context) is compound of seven logical building blocks described below:

  - **Node AI Game Agent (NGA)** is tasked with gathering critical information from the nodes, including data on resources, security status, and other relevant metrics. This information collection is essential for maintaining an accurate overview of the network's operational health. Additionally, the NGA must communicate with the Distributed Service Manager to specify and/or update the execution of CoGNETs AI services through the instantiated DDAG. This interaction ensures that the services are aligned with the current state of the nodes and optimally deployed, allowing for effective resource management and enhanced performance within the distributed environment. The update of the DDAG will be produces only if the corresponding submodule is not under execution therefore there is a specific property in the DDAG that indicates that the AI submodule is under execution in order not to update the content.

  - **Node Manager – Device Monitoring (NDMo)** is responsible for collecting essential metrics from the Node Context, providing valuable insights into the performance and status of each device within the swarm. To enhance its analytical capabilities, the NDMo formulates closed-form mathematical utility functions that correlate the dynamics of data, resources, tasks, and requirements for each device. This comprehensive approach allows the NDMo to generate a detailed ranking of the devices based on their Computing, Security, and Energy perform-

Funded by the European Union
Project funded by

ance. By evaluating these interrelated aspects, the NDMo aids in optimizing resource allocation and ensuring that each device operates efficiently within the network.

- **Node Manager – Device Registration (NDR)** is responsible for the security registration of components and overseeing the DevOps operations necessary to initialize and integrate each node into the CoGNETS Swarm-Node network. This includes implementing security protocols to ensure that new nodes meet the required standards before joining the network. Additionally, the NDR manages the automatic configuration of the distributed network, ensuring that the network plane is set up correctly for optimal performance and connectivity. By facilitating secure integration and efficient network configuration, the NDR plays a vital role in maintaining the integrity and functionality of the CoGNETS ecosystem.

- **Node Manager – Device Storage (NDS)** is tasked with managing the storage of data within the node, specifically focusing on the updates and synchronization operations related to the DDAG and the Swarm Context. The NDS ensures that all relevant data is securely stored and readily accessible, facilitating efficient data retrieval and updating processes. By maintaining a well-organized storage system, the NDS supports the seamless flow of information within the network, enabling effective collaboration and coordination among nodes in the swarm. This role is crucial for ensuring that the data remains consistent and up-to-date across the entire distributed system.

- **Node Manager – Data Manager (NDM)** plays a critical role in ensuring the provenance of data within the node, overseeing the data integrity and reliability of the information processed. It is responsible for generating and managing the signature of the CoGNETS AI results, which helps establish trustworthiness in communications across the network. By composing a digital stamp in the DDAG, the NDM ensures that each piece of data can be traced back to its source, providing transparency and accountability. This functionality is essential for maintaining confidence in the system's outputs and facilitating secure interactions among nodes within the CoGNETS framework. It is based on the Data Integrity definition of the ETSI NGSI-LD API and W3C® Data Integrity specification [75], [76], [77].

- **Node Manager – Workload Orchestrator (NWO)** is a key component responsible for determining when a Node Context must execute a CoGNETs AI module. Upon identifying the need for execution, the NWO retrieves the corresponding Docker image of the AI module or AI submodule from the Container Service Registry (CSR) along with any necessary input data. It then calls the Node Component Executor (NCE) to launch the execution of this AI module or AI submodule. Once the execution is complete, the NWO coordinates the recovery of the output data, which is subsequently updated into the DDAG through the Data Manager. This orchestration ensures that these AI modules are executed efficiently and that their results are accurately reflected in the system, enabling continuous improvement and adaptation within the network.

- **Node Manager – Component Executor (NCE)** is responsible for executing the CoGNETS AI module or submodules, ensuring that the designated tasks are carried out effectively. Upon completion of the execution, the NCE returns the results to the DDAG data structure managed by the Data Manager. This process is crucial for maintaining the flow of information within the network, as it en-

Funded by the European Union
Project funded by

sures that the outputs from the AI module are accurately captured and integrated into the overall system. By facilitating the execution and result reporting, the NCE plays a vital role in the functionality and responsiveness of the CoGNETS framework.

- **Middleware Layer (Security Management)** includes several security levels applied depending of the context and level of application. We have identified four levels of security applied:

  - **Hardware-level Security (S-HW)** system is designed to provide robust protection through the use of Physically Un-clonable Functions (PUF) and various hardware mechanisms, effectively preventing the exfiltration of sensitive information even in the case of a complete breach of the software within the Decentralized Identity (DID) framework. Additionally, it ensures the secure execution of critical software code by implementing separation of software components and utilizing RISC-V remote control flow attestation and integrity checks. This approach significantly enhances the overall trustworthiness of the system, safeguarding against unauthorized access and manipulation. RISC-V and the PUF will be use as storage system of the credentials of the Node owner used in the DID authentication and authorization process as well as the Data Integrity signature process.

  - **System-level Security (S-SL)** will be supported by low-overhead Decentralized Identity (DID) mechanisms, serving as extensions of the DID Management framework for node identity and authentication. These mechanisms will be seamlessly integrated into the IoT-to-Cloud environment, ensuring secure and efficient identity verification and authentication processes across all nodes. This approach will enhance security while maintaining optimal performance, facilitating reliable interactions within the system. It will be based on the innovative adoption of M2M scenario of a data space mechanism into Edge-Cloud scenario.

  - **Application-level Security (S-APL)** will focus on conducting a thorough requirements analysis for the identity layer of the swarms concept. This will involve developing a comprehensive threat model using methodologies such as attack trees, STRIDE, and MITRE, facilitated by a novel application layer dedicated to monitoring and anomaly detection. Throughout all phases of the project, a continuous analysis of the security state of the system and its components will be performed to identify potential security and privacy risks. Special attention will be given to vulnerabilities associated with the interface between Physically Unclonable Functions (PUFs) and the DID layer, ensuring a robust framework that mitigates threats effectively.

  - **AI-level Security – Adversarial Shield (S-AI)** will encompass a comprehensive analysis of distributed learning and swarm intelligence to identify potential threats. This analysis will inform the development of an adversarial shielding mechanism aimed at strengthening the security framework and safeguarding the actions that directly influence the learning process. Additionally, the system will implement sanity checks for individual contributions, tailoring these checks to suit the specific learning environments of each CoGNETs pilot use case. A thorough risk analysis will be conducted to prioritize the most relevant threats and determine suitable mitigation strategies, ensuring robust protection for the AI components throughout their operational lifecycle.

Funded by the European Union
Project funded by

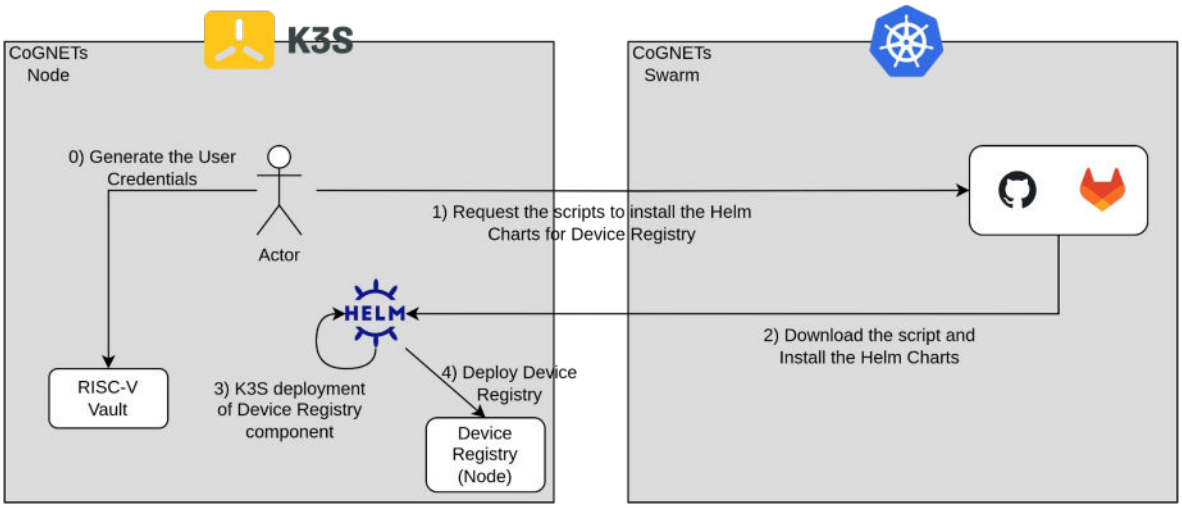
6.3 ARCHITECTURAL VIEW

The following architectural views aim to show how the system's functions and components are organized and interact with each other and partially showing some of the underlying technology that we'll use as building blocks. Each section will show how the different functions of the project are architected. The following subsections are focused on mapping the requirements specified in the previous section and the building blocks that we will implement in the architecture.

6.3.1 Registration process

This process shows how a new K3S registers to the CoGNETs Swarm in several steps, reflecting the requirements for the DRM and the CI/CD pipelines, showing how a device is registered to the DRM and showing the deployments of the software for the K3S.

Figure 13: Initial steps in the registration process

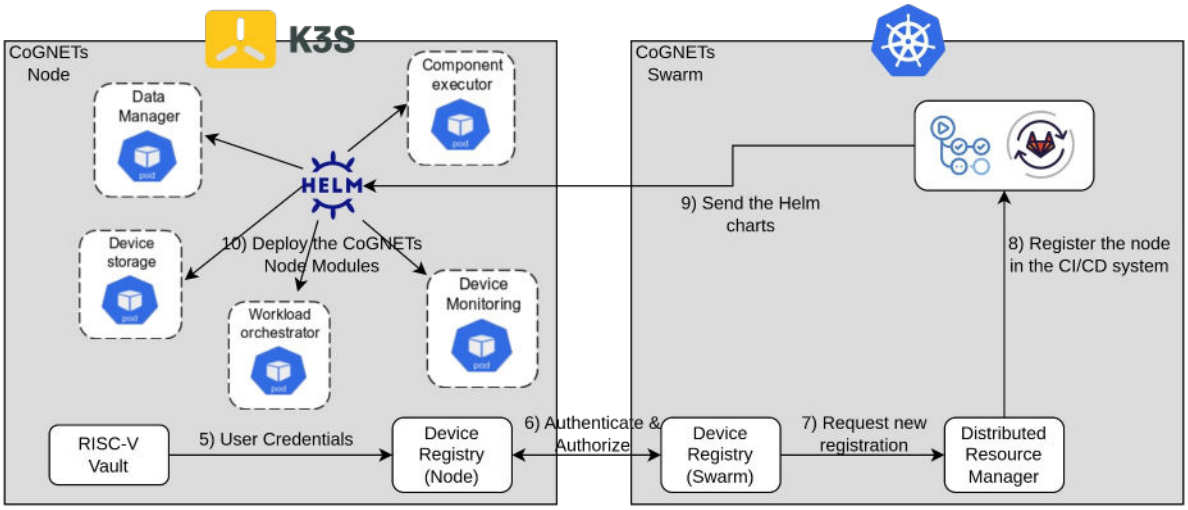


When a new K3S node joins the CoGNETs Swarm, it initiates the registration process by generating node credentials, securely stored stored in in a RISC-V Vault. These credentials will be known by the CoGNETs Swarm. We need to know that machines or services have no identity, the identity is associated to the persons and owned the machines and/or services. Therefore, the process consists on generating user credentials to be used by machines or services based on the identity of the person (physical or legal) who owns those machines or services.

Next, the K3S requests the scripts and the Helm charts to install the required software for the Device Registry (step 1). The node then downloads these scripts and the helm charts into the K3S Node (Step 2). Finally, Helm Step deploys the Device registry for the node (Steps 3 and 4).

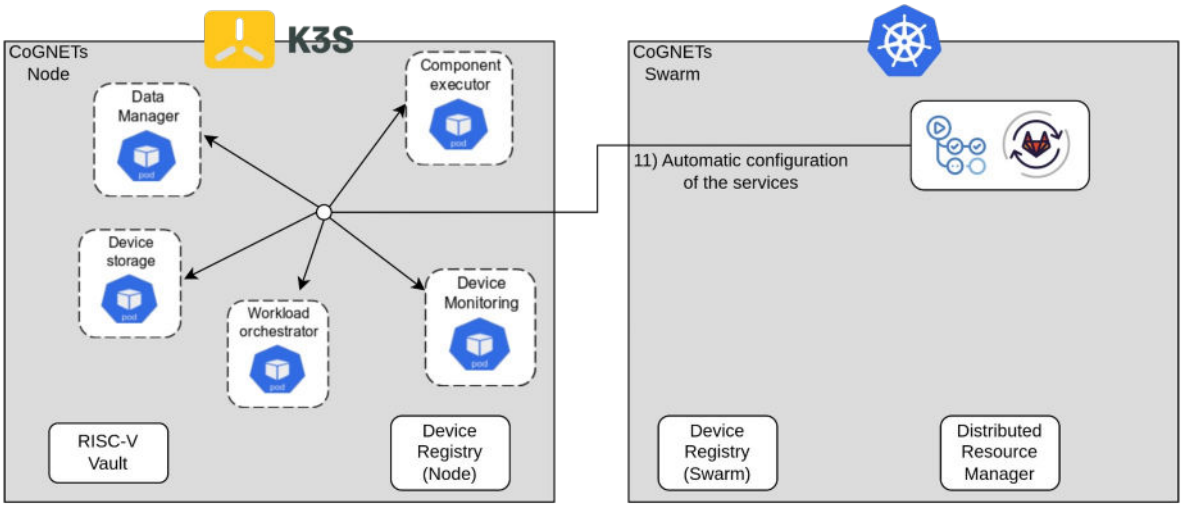
Funded by the European Union
Project funded by

Figure 14: Registration of the Edge node into the CI/CD process



The device registry of the K3S node retrieves the user-generated credentials from the RISC-V Vault (Step 5) and exchanges them with the Device registry in the Swarm node (Step 6), authenticating and authorizing each other. Once the K3S Node is authenticated and authorized in the CoGNETs Swarm cluster, it triggers a new registration process in the Distributed Resource Manager (Step 7), which requests to the CI/CD system to Register the node (Step 8) and send to the K3S Node the corresponding helm charts to be run in the K3S Node (Step 8 –Should be Step 9 in Diagram). These helm charts include all the CoGNETs Node Modules, including Data manager, Component executor, Device storage, Workload orchestrator and Device monitoring (Step 10).

Figure 15: Automatic configuration of the Edge logical building blocks



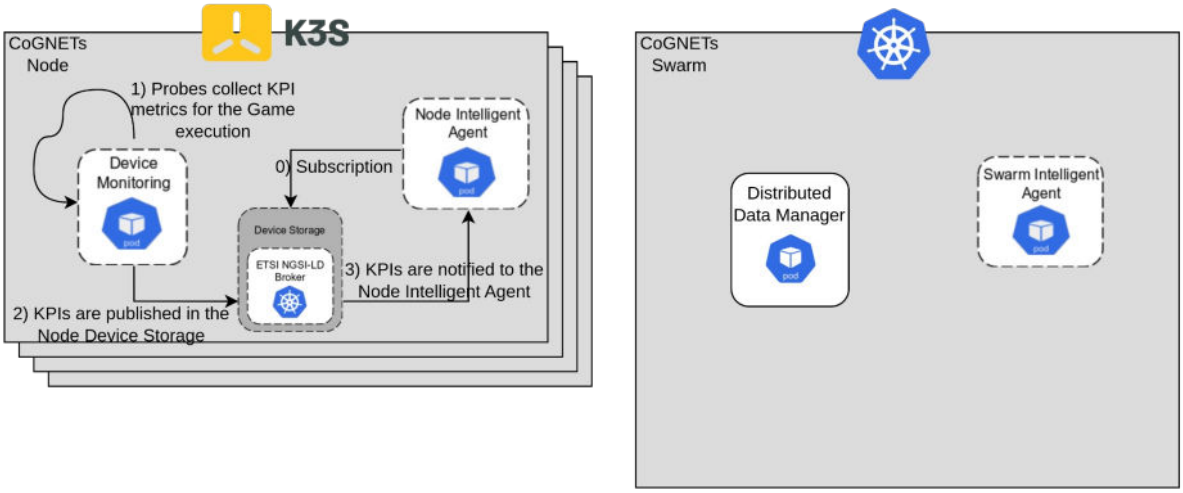
Once the services are deployed, they interact with the CI/CD system in order to trigger the automatic configuration process (Step 11) for each service. The configuration data retrieved from the central CI/CD system seamlessly configures and reconfigures all services. A single change in the central repository is enough to fine-tune each service across every K3S cluster, ensuring consistency to all registered and deployed services.

Funded by the European Union
Project funded by

6.3.2 Game Intelligent Agent System: Nodes Selection

In this point we describe how the nodes collect data in order to make metrics available and allowing CoGNETs to have context from every device. This picture shows the process of acquiring self-assessment in the nodes and how the Device monitoring can reach the expected functionalities.

Figure 16: Collection of KPIs for the evaluation of the Game execution



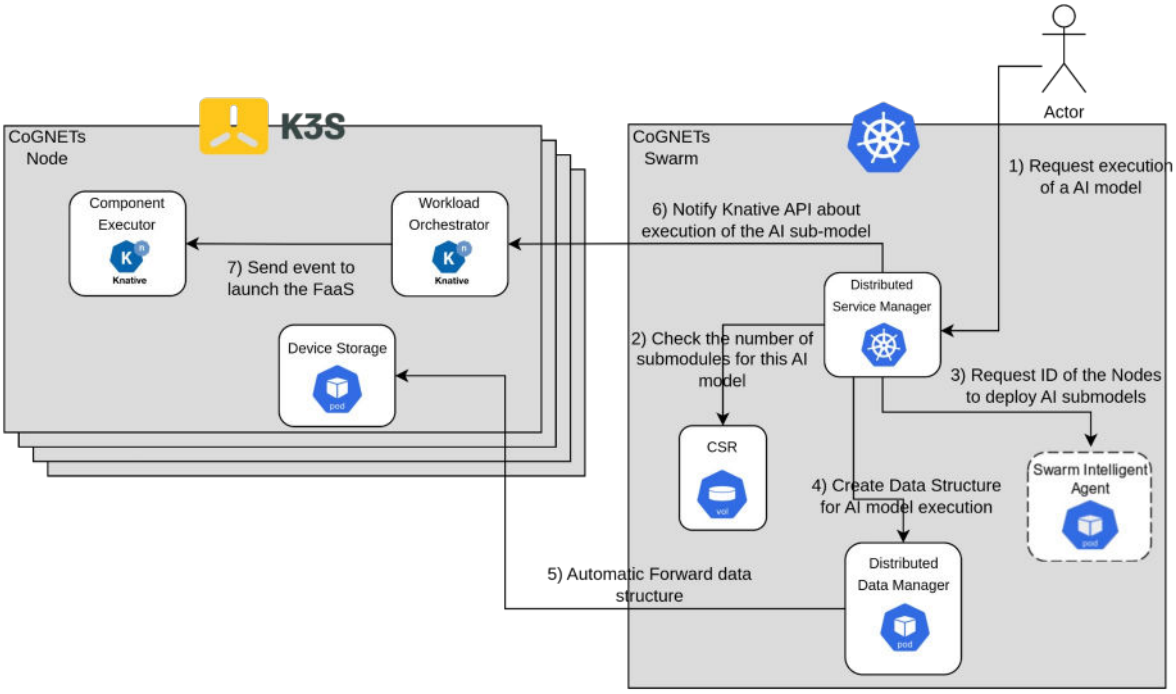
Within the K3S Node, first the Node Intelligent Agent subscribes to the Device Storage (Step 0). The Device Monitoring process collects metrics based on the selected KPIs for the Game processes execution (Step 1). These collected metrics and KPIs are then published to the Node's Device Storage (Step 2). The Node Intelligent Agent (running the Games) uses these KPIs for their execution process (Step 3).

6.3.3 AI modules execution

The following figures represent how the Workload Orchestrator, the Node Manager, the Node Data Manager, among other components, can achieve their functional requirements and we'll show the relationship between all of them and the sequences in their communications in order to achieve the desired functionalities.

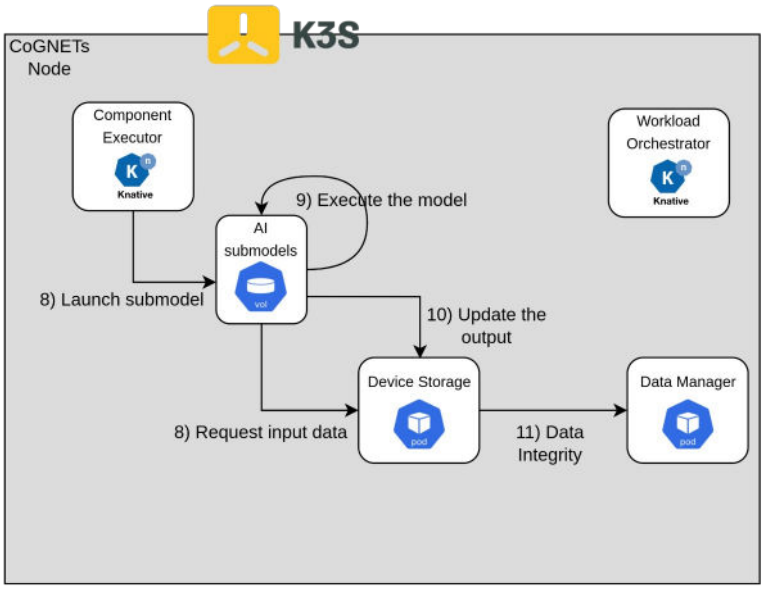
Funded by the European Union
Project funded by

Figure 17: Selection of an AI Module to be executed in the CoGNETs platform



When an actor requests the execution of an AI model (Step 1), the Distributed Service Manager checks the number of submodules required for this AI model in the CSR (Step 2). It then requests the ID of the nodes where the AI models will be deployed from the Swarm Intelligent Agent (Step 3) and creates the data structure for the AI model execution in the Distributed Data Manager (Step 4). The Distributed data Manager forwards the data structure to the Device Storage in the K3S nodes (Step 5). The Distributed Service Manager notifies the Workload orchestrator in the K3S nodes about the execution of the IA sub-model (Step 6). Within the K3S nodes, the Knative Workload orchestration runs the corresponding Function as a Service (FaaS) in the Component executors (Step 7).

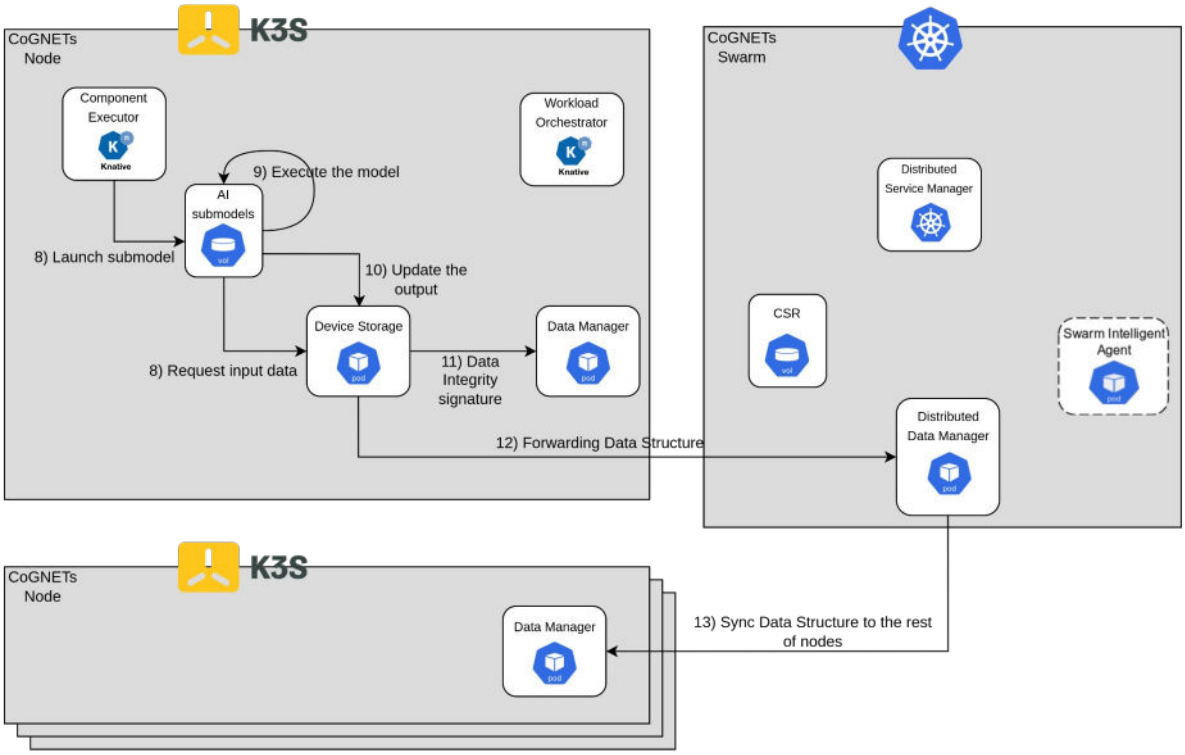
Figure 18: Execution of AI Modules on the Edge nodes



Funded by the European Union
Project funded by

Now, the component executor in the selected K3S node determines what to run, selecting amongst all available AI submodels, and launches the corresponding submodel (Step 8). The submodel is then executed (Step 9), retrieving input data from the device storage and updating the Device Storage with the output from the model execution (Step 10). Checksums and validity hashes are calculated in the Device Storage to ensure data integrity, and these integrity models are stored in the Data Manager (Step 11).

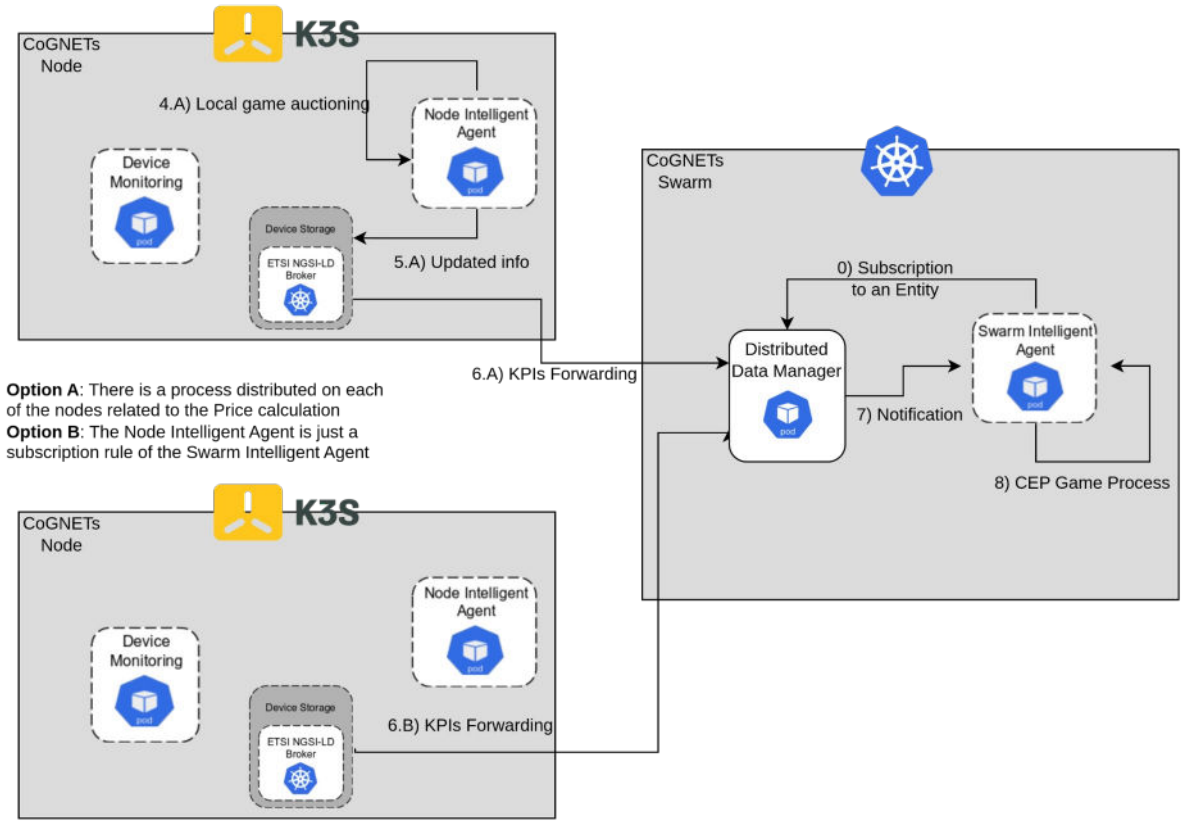
Figure 19: Synchronization of the execution of AI modules



The device Storage forwards the Data Structure to the Distributed Data manager (Step 12) which then re-distributes this data structure to the remaining nodes, synchronizing them all (Step 13).

Funded by the European Union
Project funded by

Figure 20: Price calculation and optimization of the Game process



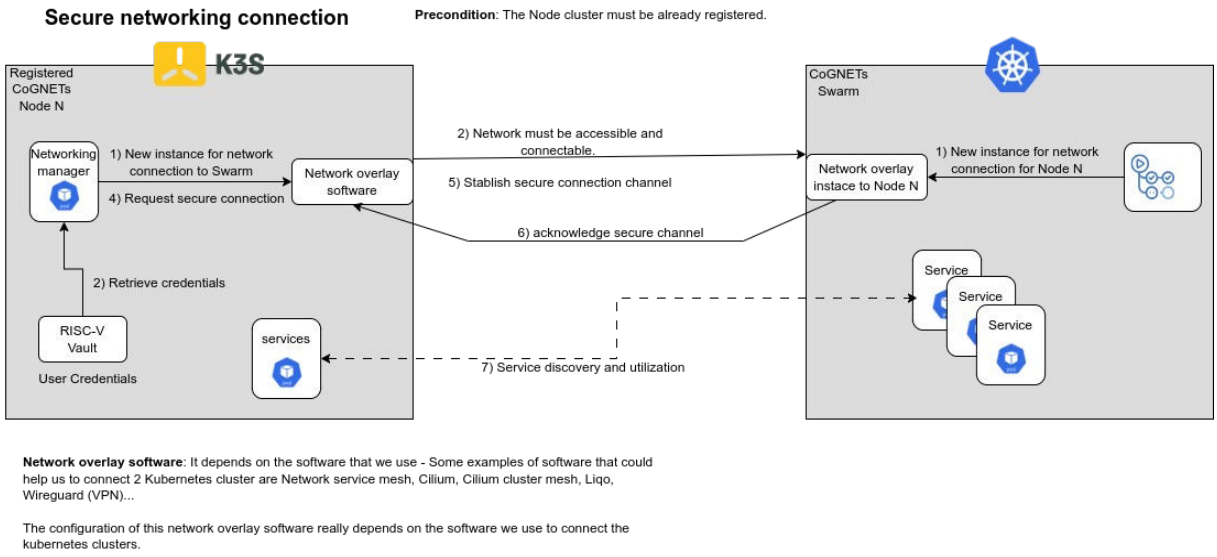
There are two possible approaches for implementing the game process. In Option A, there is a local process to the K3S nodes which calculates the price and makes a local auctioning which is forwarded to the CoGNETs Swarm cluster. In Option B, the calculations are centralized in the CoGNETs Swarm cluster according to the KPIs provided by each of the K8S nodes. The implementation of option A or B will depend on later decisions made by the partners working in WP3.

6.3.4 Secure networking connection

CoGNETs will be able to join many different and distributed K3S nodes. As part of the registration process, we'll have to trigger a mechanism which allows seamless networking between all the different nodes and the CoGNETs Swarm. This picture shows how network is configured accordingly to the registrations in the Node Manager.

Funded by the European Union
Project funded by

Figure 21: Security configuration of the CoGNETs network

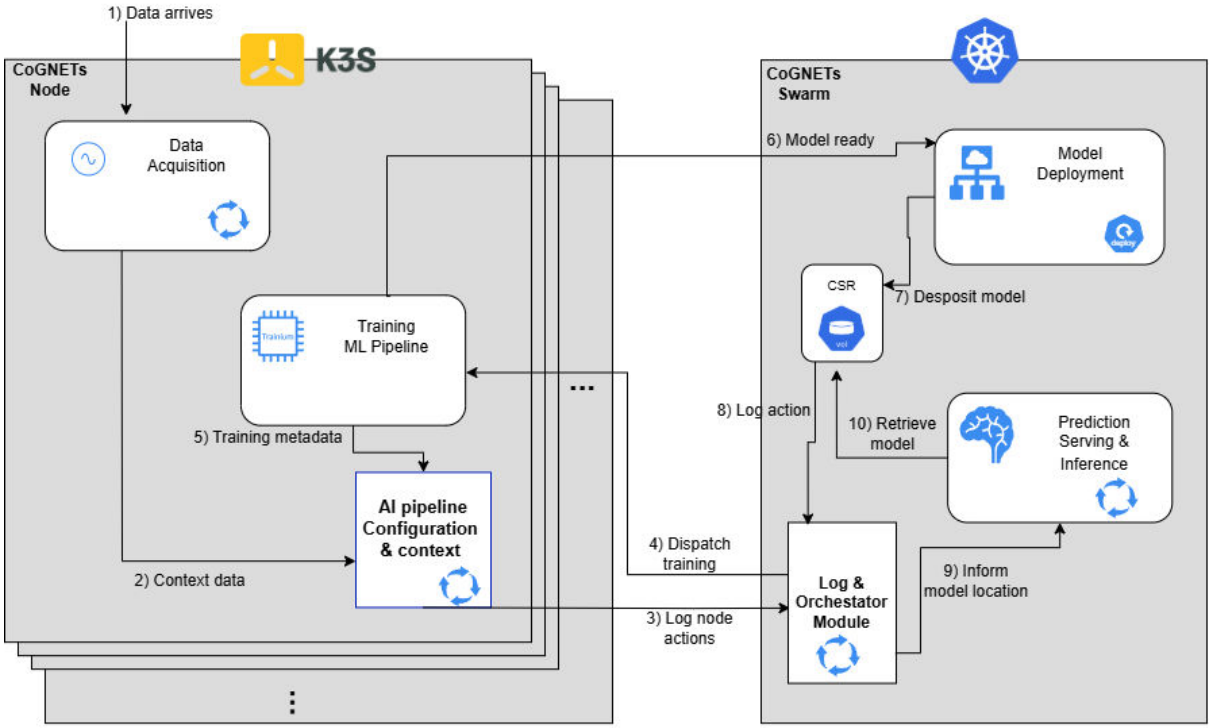


After registering the K3S cluster, a secure connection is established between the new K3S cluster and the CoGNETs Swarm node. Both the CoGNETs Swarm and the K3S cluster start the chosen network overlay software (Step 1) to create an overlay network between the two different nodes, being mandatory to have network connectivity between both clusters (Step 2). There are several software candidates to implement this networking communication between nodes (e.g. cilium, cilium mesh, wireguard, network service mesh, Liqo) and the proper way to establish that communication depends on the chosen solution. However, in order to establish the secure connection, on one side the K3S networking manager must retrieve the credentials validated in the connection process (Step 3) and request a secure connection (Step 4). Using the appropriate credentials and certificates, the communication is established and acknowledged (Steps 5 and 6). Once ready, this connection enables the discovery and utilization of remote services (Step 7).

Funded by the European Union
Project funded by

6.3.5 DevOps and MLOps activities

Figure 22: MLOps flows



The process begins when new data arrives at the node (Step 1). The Data Acquisition module ingests and preprocesses this data—performing tasks such as cleansing, normalization, and feature extraction—while collecting any relevant metadata (Step 2). This information is then logged by the Log & Orchestrator module (Step 3). Next, the Training ML Pipeline (Step 4) manages activities such as hyper-parameter tuning, iterative model training, and validation. The training data is saved in the AI Pipeline Configuration & Context (Step 5), ensuring adherence to correct parameters, resource constraints, and versioning policies. Once training is complete, the process announces that the model is ready to be deployed (Step 6). The Model Deployment module, which handles containerization, health checks, and model versioning, finalizes the deployment and places the model in the CSR (Step 7). This action is recorded by the Log Module (Step 8). Finally, the Prediction Serving & Inference module retrieves model information (Step 9) and the newly deployed model for inference (Step 10), enabling real-time or batch prediction serving and closing the loop of this MLOps pipeline.

6.4 TECHNOLOGIES AND TOOLS TO ARCHITECT COGNETS

In this section we introduce the tools that are already available and will be integrated into the CoGNETs platform in order to facilitate the creation of the Edge-Cloud continuum platform. The following subsections introduce those components that will be specifically configured afterwards in the different work packages to facilitate the functionality of the CoGNETS logical building blocks identified in this document.

Funded by the European Union
Project funded by

6.4.1 FIWARE IoT Agents

FIWARE IoT Agents address a common set of challenges in the IoT landscape, particularly the diverse data representation formats and heterogeneous communication protocols. IoT Agents provide a solution by enabling a group of devices to send their contextual information to and be managed by an ETSI NGSI-LD Broker using their native protocols. Furthermore, IoT Agents must effectively handle security aspects, including authentication and authorization processes, to establish secure communication between IoT devices, robots, and the ETSI NGSI-LD Broker. This security mechanism is based on the standardized management of JSON Web Tokens (JWT) in the header.

The IoT Agent components function as practical interfaces with IoT devices, robots, and third-party systems, facilitating the retrieval of valuable contextual information (a.k.a. entities in ETSI NGSI-LD API data model) and enabling the triggering of actions in response to updates. The primary purpose of the IoT Agent is to serve as a gateway that translates payloads and transport protocols into a JSON-LD format compliant with the ETSI NGSI-LD API. This translation allows for effective querying or subscription to changes occurring in the real world, which is particularly beneficial for monitoring various parameters represented in the attributes of NGSI-LD entities at the ETSI NGSI-LD Broker level.

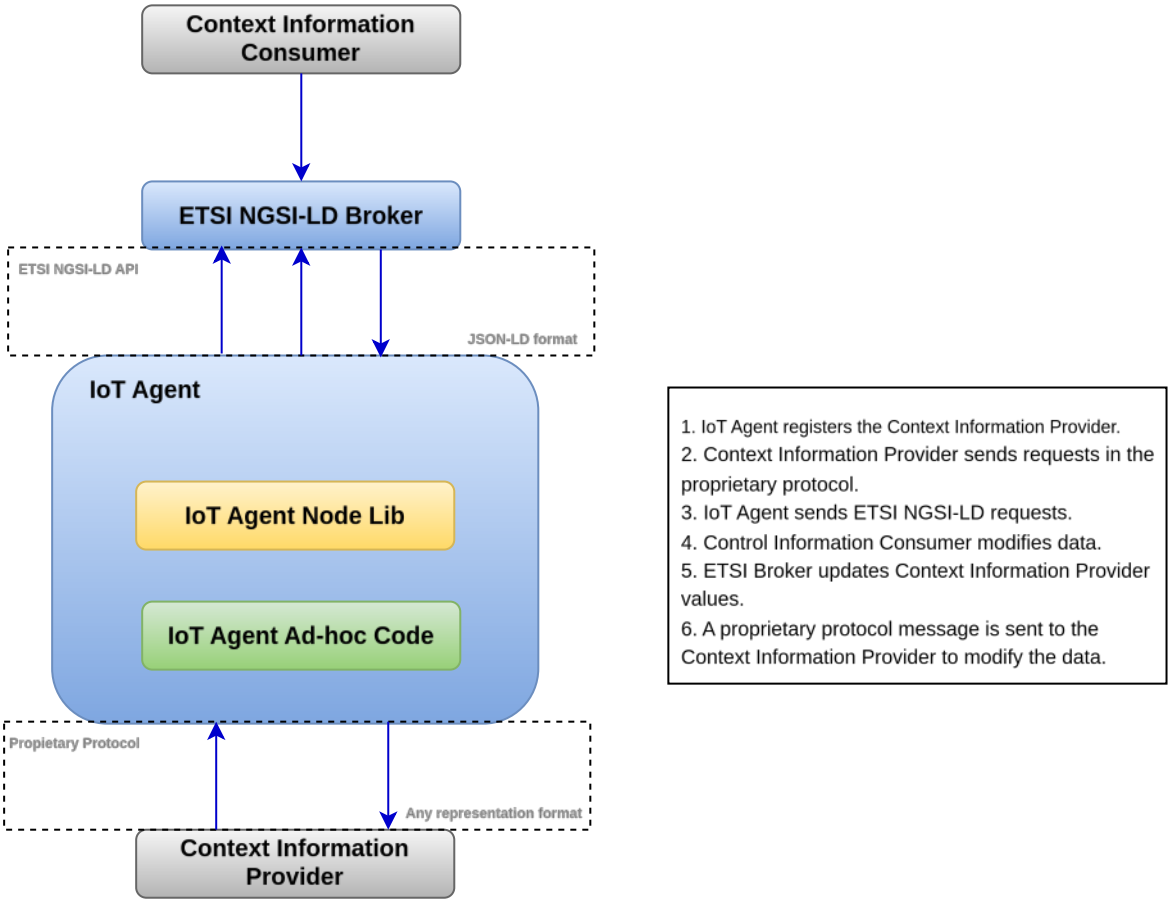
Consequently, IoT Agents serve as intermediaries that manage the complexity and heterogeneity of data integration and protocol transformation. They ensure that contextual information from diverse sources is represented and managed in a standardized manner, making it accessible through the Swarm Node using the ETSI NGSI-LD.

The design of IoT Agents has been developed to facilitate the future creation of new components that utilize different protocols and data format representations. This is accomplished through the encapsulation of all ETSI NGSI-LD API operations into a dedicated library, known as the IoT Agent Node Lib. The primary purpose of this library is to provide a common framework for provisioning IoT Agents, enabling each individual IoT Agent to access standardized mapping data for devices and to offer a range of utility functions for Northbound communications, typically involving connectivity with ETSI NGSI-LD Brokers. As a result, each IoT Agent represents the implementation of tailored code to handle the proprietary communication protocols and data format representations of the devices and robots.

Funded by the European Union

Project funded by

Figure 23: Architecture of an IoT Agent



As illustrated in the figure, the IoT Agent is divided into northbound and southbound communications:

- **Southbound communications:** The IoT Agent Ad-hoc Code monitors changes in context information entities and triggers specific callbacks to the IoT Agent Node Lib for processing the information obtained from devices and robots.
- **Northbound communications:** The IoT Agent Node Lib provides an interface that accepts structured input data in accordance with the defined data model for the captured context information entities, facilitating the transmission of this data to the appropriate ETSI NGSI-LD Broker.

Currently, the IoT Agent manages JSON, UL, and XML data representation formats, as well as communication protocols such as HTTP, MQTT, LWM2M, LoRaWAN, SigFox, and OPC-UA. It utilizes a facade pattern to simplify the handling of this complexity. In the context of CoGNETs, we will primarily use MQTT, and potentially OPC-UA if needed for PUC1.

Additionally, the IoT Agent offers a straightforward HTTP REST API that provides common functionalities for accessing, provisioning, and shutting down context information providers, as well as configuring a group of devices with similar properties. The following table lists all the operations exposed by the API, which can be utilized to integrate the agent with other components of the CoGNETs.

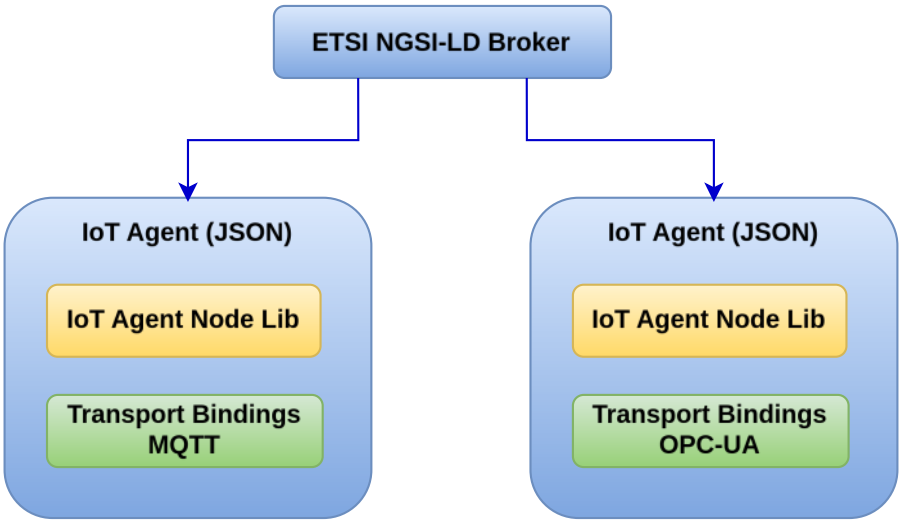
Funded by the European Union
Project funded by

Table 6: List of IoT Agent operations.

Endpoint	Operation	Description
/iot/about	Service Health	Check the IoT Agent Service Health status for information on the IoT Agent Node Lib version, the port used, the base root, and the version of the IoT Agent.
/iot/services	Service Group Management	Provide operations to manage the group management of IoT Agents. Group management refers to a set of IoT Agents that share the same properties, including API Key, ETSI NGSI-LD Broker URL, entity type, resource used in the base URL, and the list of attributes.
/iot/devices	Device or actuator management	Provides operations related to the management of a specific device or actuator.

At this stage of the project, utilizing the IoT Agent with JSON payload format and support for MQTT and OPC-UA transport protocols appears to be highly relevant for the CoGNETs project. The focus of the project will be on provisioning and supporting the corresponding IoT Agents to be adopted for each of the PUCs as well as the implementation of the corresponding building blocks to collect the status of the Edge node and the synchronization of the execution of the AI modules.

Figure 24: Composition of the different IoT Agents



Funded by the European Union
Project funded by

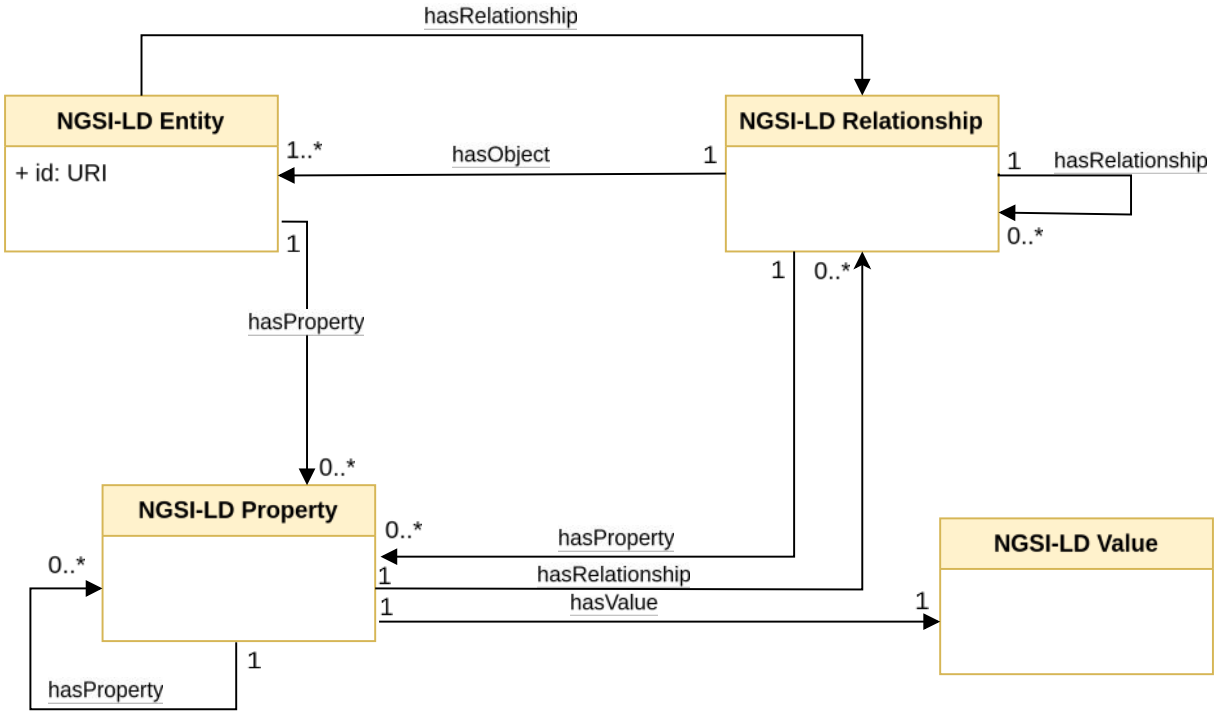
6.4.2 ETSI NGSI-LD Broker (FIWARE Orion-ld)

The ETSI NGSI-LD Broker (hereafter referred to as the Broker) is a decentralized storage system responsible for maintaining the state of context information among various elements or components within each domain. This component enables the provision and consumption of context information. Additionally, the Broker allows subscriptions to context information, facilitating notifications about any changes to the data.

Context information consists of a series of attributes associated with entities, reflecting the status and behaviour of the real world. This concept is also referred to as "Digital Twins." Furthermore, the Broker has the capacity to exchange this contextual information through the implementation of the ETSI NGSI-LD API. To understand this API, it is essential to define some of its principles. In ETSI NGSI-LD API, the world comprises a set of entities, which are composed of:

- An **entity identifier** that uniquely distinguishes each entity.
- An **entity type**, which describes the kind of information provided by an entity and is associated with a corresponding defined data model that outlines this information.
- **Properties** that represent the information expected to be found in these entities. Each property has values and may include other sub-attributes, which are pieces of metadata that describe the attribute. The definitions of these properties are established in the corresponding **data model**.
- **Relationships** that represent the connections between different entities.
- **Values** defined for each property.

Figure 25: NGSI-LD information model as UML

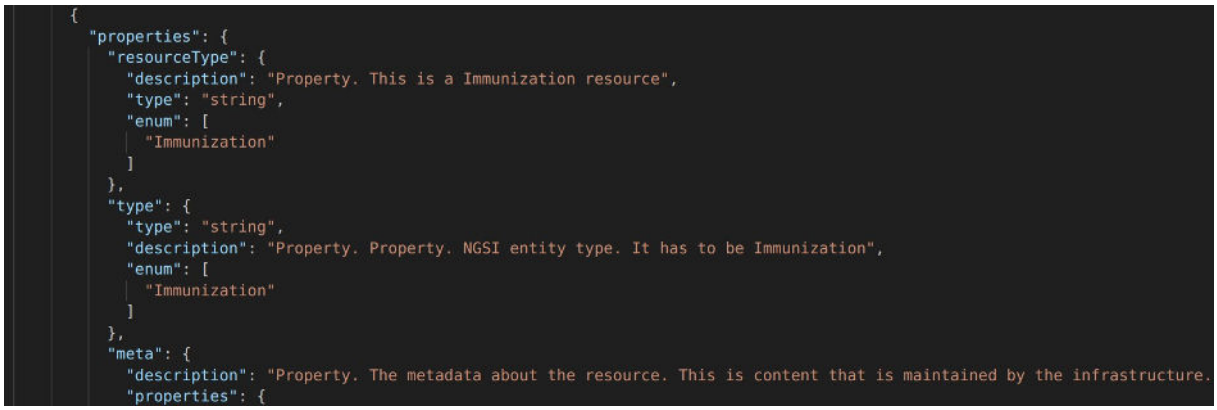


Funded by the European Union
Project funded by

As a result, a diverse array of objects can be represented as entities, including drones, robots, sensors, and others, contingent upon the availability of a suitable data model to describe each entity. The ETSI NGSI-LD API provides a standardized framework to support the modelling of these entities, utilizing the JSON-LD data format along with JSON Schema to define the structure of this data.

Moreover, JSON-LD's @context is employed to expand the properties defined for an entity. This @context contains a URI or a collection of URIs that offer the appropriate semantic definitions for the properties, as well as the expected value types. Each property is also accompanied by a description and a URL to the model definition. This separation of semantic definitions from entity properties facilitates improved interoperability and data management.

Figure 26: Example of definition of properties in JSON Schema

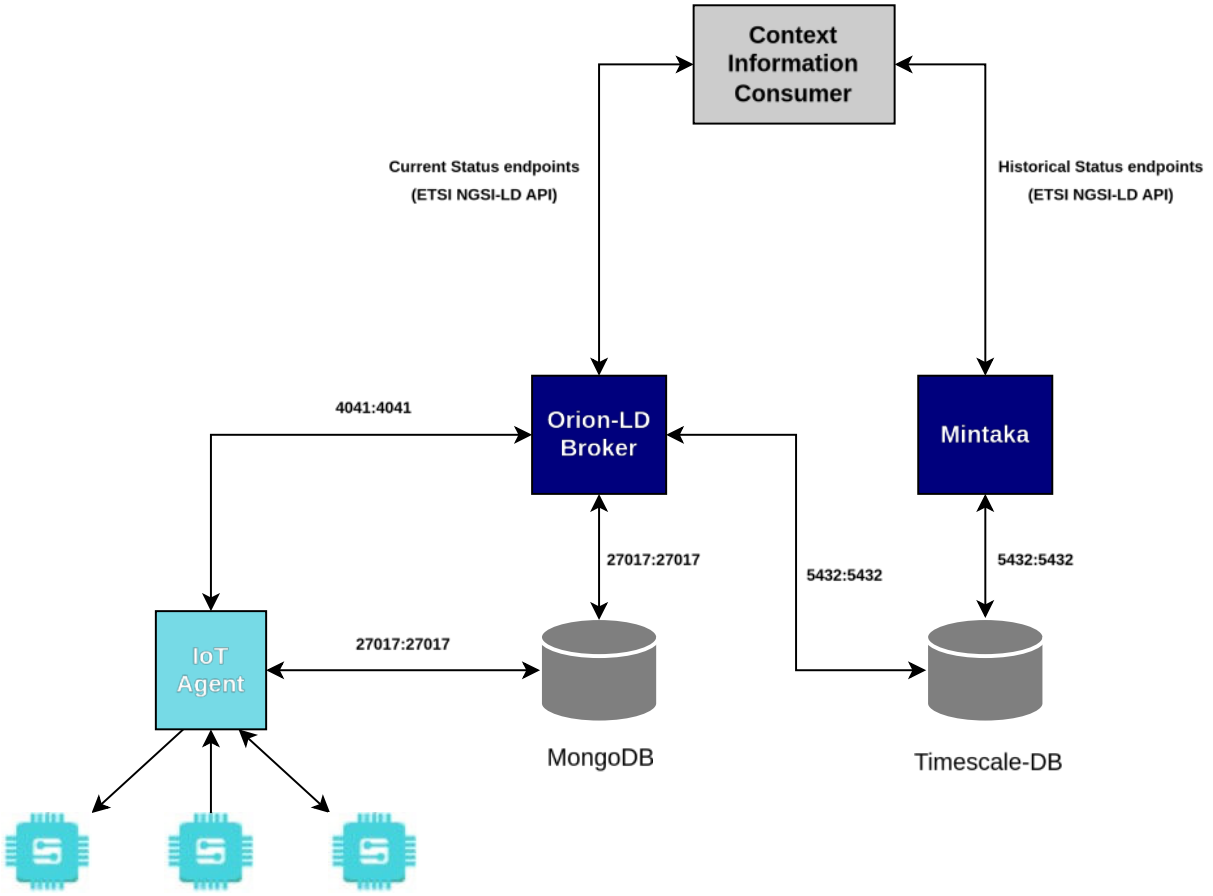


Furthermore, a Broker enables the temporal representation of entities using sub-properties such as createdAt, modifiedAt, deletedAt, and/or observedAt. The ETSI NGSI-LD API outlines a set of operations for managing the temporal evolution of entities and their attributes, as well as for consuming this data. The Broker employs a historical database (e.g., TimescaleDB in FIWARE Orion-ld implementation) to store and access the historical information of entities. Temporal data consumption (GET operations) is facilitated by Mintaka, while data provision operations (POST, PATCH, DELETE) are handled by a separate Orion-ld instance, effectively distinguishing between read and write operations for temporal data.

Funded by the European Union

Project funded by

Figure 27: ETSI NGSI-LD Broker and IoT Agent in the CoGNETs architecture

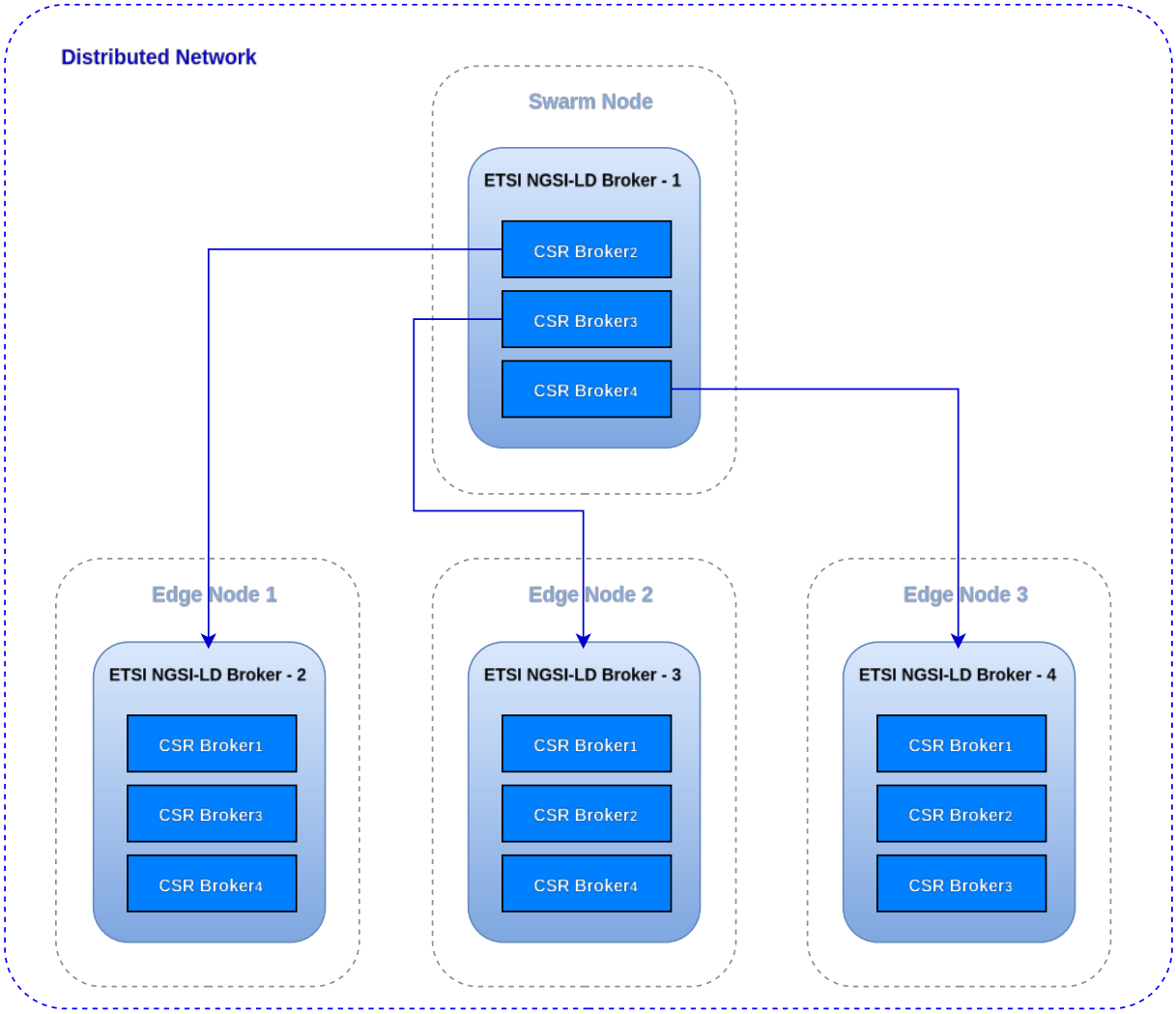


Additionally, a critical feature of the Broker is its capacity to access the state of the continuum in a decentralized manner. This allows the Swarm Context to monitor the continuum's status and adjust configurations within the Edge domain, regardless of the ecosystem's condition. This distributed state concept can be viewed as a decentralized storage system that preserves state information among various components of the CoGNETs architecture. Brokers are capable of synchronizing this contextual information through defined mechanisms in the ETSI NGSI-LD API, referred to as Distributed Operations. As a result, the Broker will maintain multiple entities and their attributes across a distributed network.

The implementation of Distributed Operations, known as Context Source Registration (CSR) operations in the ETSI NGSI-LD API, is currently underway in the development of Orion-ld. A CSR is an operation that informs a Broker where non-local entities can be located. As a result, a local query to a Broker retrieves not only local entities but also remote entities through a distributed request to the Brokers associated with the matching CSR, with those remote entities appended to the final response. Consequently, any Broker can query others within the distributed network and receive the same response. For this system to function effectively, each Broker must have at least one CSR for every other Broker connected to the Edge nodes in this distributed network.

Funded by the European Union
Project funded by

Figure 28: Distributed Operations of ETSI NGSI-LD Brokers



The ETSI NGSI-LD API outlines various methods for configuring Context Source Registrations (CSRs) to manage the distributed state, but we focus on the following two approaches:

- **Inclusive CSRs**: In this approach, each Broker replicates the entire state. This method allows for faster queries and helps resolve potential communication issues among the edge nodes.
- **Exclusive CSRs**: In this approach, each Broker retains only its own data. During queries, each Broker forwards requests to other Brokers. While this method enables faster update operations, it results in slower query responses.

Finally, a Broker provides a straightforward HTTP REST API that facilitates common functionalities for accessing, provisioning, and subscribing to operations over entities. The following table summarizes all the operations that will be exposed by the Logical Building Blocks utilizing the Broker as an implementation solution for the CoGNETs platform.

Funded by the European Union
Project funded by

Table 7: List of ETSI NGSI-LD API endpoints

Endpoint	Operation	Description
/ngsi-ld/v1/entities	Context Information Provision	Operations for Managing Entities and Attributes
/ngsi-ld/v1/entities /ngsi-ld/v1/entityOperations	Context Information Consumption	Operations for Consuming Entities and Checking Available Entity Types and Attributes
/ngsi-ld/v1/subscriptions	Context Information Subscription	Operations for Subscribing to Entities, Receiving Notifications, and Managing Subscriptions
/ngsi-ld/v1/temporal/entities	Temporal Context Information Provision	Operations for Managing the Temporal Evolution of Entities and Attributes
/ngsi-ld/v1/temporal/entities /ngsi-ld/v1/temporal/entityOperations	Temporal Context Information Consumption	Operations for Consuming the Temporal Evolution of Entities
/ngsi-ld/v1/csourceRegistrations	Context Source Registration	Operations for Registering Context Sources and Managing Context Source Registrations (CSRs)
/ngsi-ld/v1/csourceRegistrations	Context Source Discovery	Operations for retrieving and discovering CSRs

6.4.3 Streamhandler

INTRA's Streamhandler is a platform that enables interconnecting, storing, processing and visualizing real time data. Aside from message handling, the platform can be used as an MLOps orchestrator facilitating the triggering of AI models as well as their deployments. This effectively renders Streamhandler a full-blown Big Data solution with AI orchestration capabilities.

The Streamhandler platform is a high-performance distributed platform for handling real-time data based on various data streaming technologies like Apache Kafka. It can efficiently handle massive amounts of messages and data into processing pipelines, for both real-time and batch processing. Streamhandler is a scalable solution that offers efficiency, robustness and performance to address challenging data handling and messaging requirements. Its underlying technologies can support any type of data-intensive services from cloud to edge, also facilitating a security-and-privacy by design approach. By identifying and designing the Streamhandler's Connector's specifications, the user is able to provide trigger integration activities and manage streaming data from multiple sources. The key capabilities and features offered by the platform include:

1) Real-time monitoring and event-processing

2) Interoperability with all modern data storage technologies

Funded by the European Union
Project funded by

3) Distributed messaging

4) High fault-tolerance and support of automatic recovery

5) High scalability

6) Security (encryption, authentication, authorization).

Streamhandler is capable of scaling out and accommodating various data streams from different data domains and can support all major data-centric programming languages including Python, Java, R and Scala.

The Streamhandler Platform includes the following components:

1) Integrated Connectors

2) Streaming Core Platform

3) Schema Registry

4) Security Management

5) Platform Admin and Monitoring Dashboard.

Streamhandler, developed by INTRA, will be the key enabler of the CoGNETs lab-based testbed. The CoGNETs' lab-based testbed is a multi-faceted environment designed to facilitate the development, experimentation, and demonstration of the advanced AI and cognitive computing technologies developed throughout the project. In the context of CoGNETs, the Streamhandler platform will function as the orchestrator of the CoGNETs' MLOps and as the intermediary between user requests and service integration triggering. The platform will be validated and refined through the project's use cases as it will have to be adjusted based on the ongoing project's requirements leading to a major enhancement of its operations and integrated tools, further increasing its capabilities and potential uses.

Funded by the European Union

Project funded by

7 ARCHITECTURE REQUIREMENTS

The objective of this section is to provide a consolidated view of the CoGNETs requirements and a formal identity to facilitate the tracking of them during the project's lifespan. For this purpose, we assess the intend position of the project within distributed IoT infrastructures deployed into Edge-Cloud Continuum environment and taking into account the interconnection technologies involved in the PUCs without forgetting the security and AI aspects. During the execution of the corresponding requirements collection were identified five categories and eighteen logical building blocks and three physical subgroups. They are the following:

- **Application Layer** includes the User Interface (UI), Cognitive AI Service Repository (CSR), and the DevSecOps platform (DOP).
- **Middleware Layer – Swarm Context** includes all the core services of the platform to be executed in the Swarm cloud: Swarm AI Game Agent (SGA), Distributed Resource Manager (DRM), Distributed Service Manager (DSM), Distributed Workload Manager (DWM), Distributed Data Manager (DDM).
- **Middleware Layer – Node Context** includes all the services to be executed on the edge nodes: Node AI Game Agent (NGA), Device Monitoring (NDMo), Device Registration (NDR), Device Storage (NDS), Data Manager (NDM), Workload Orchestrator (NWO), Component Executor (NCE).
- **Middleware Layer – Security** includes all the requirements to manage the security operation of the platform. They are divided into Hardware-level Security (S-HW), Software-level Security (S-SL), Application-level Security (S-APL), AI-level Security (S-AI).
- **Physical Layer** includes the requirements/restrictions related to the hardware (HW), operating system (OS), and/or connectivity types (CT) that could be requested by the PUCs.

Additionally, each of the logical building blocks and physical subcategories are divided into four type of requirements:

- **Functional Requirements (FNC)** which describes what the system should do (e.g., features, functionalities).
- **Non-Functional Requirements (NFN)** which defines how the system performs its functions (e.g., performance, security, usability).
- **Business Requirements (BUS)** which shows the high-level needs of the organization, focusing on goals and objectives, related to the corresponding logical building block or physical subgroup or requirements.
- **Technical Requirements (BTC)** which details specific technical specifications necessary to implement the business requirements.

Besides, each of the requirements is defined with the purpose to clearly identify the subject and condition of them where the subject answers to who should request the requirement (e.g., a user, the dashboard, etc.) or to 'Who/What' this requirement refers to and the condition provides the attribute or attributes that allow a specific requirement to be formulated (i.e.,

Funded by the European Union
Project funded by

it makes a requirement for a specific subject dependent to pre-conditions rather than be generic for all potential cases). In addition, each requirements have identified the corresponding action, object, and constraint and/or value, where:

- The **action** shows 'in what way/how' this requirement is materialized.
- The **object** answers to 'what is the requirements for the subject'.
- The **constraints** provide the restrictions for which the action will apply to the object.
- The **value** answers to how much the requirement is (either measurable or comparable to something else).

It helps us to identify possible KPIs in the future implementation of the CoGNETS logical building blocks and follow up on compliance with the identified requirements.

Moreover, CoGNETs has adopted the MoSCoW technique to prioritize the creation of requirements. This method is used in project management and requirements gathering with the intention to prioritise and categorize the requirements based on their importance and urgency. MoSCoW method classifies requirements in four different premises:

- **Must have (M)**: requirements that are essential to the success of the CoGNETs project or the core functionality of the product. They are not negotiable and cannot be compromised.
- **Should have (S)**: requirements are important but not critical, therefore can be prioritized lower that must-have requirements.
- **Could have (C)**: requirements that are nice to have if resources permit it, therefore they are not mandatory requirements. These requirements are desirable but not essential for the project success or core functionality.
- **Won't have (W)**: requirements that has been dismissed or have low priority and can be excluded intentionally from the current project. Agreed not to include in the current project scope.

This method was applied separately to all requirements for each of the identified logical building blocks. The result is a comprehensive set of functional and non-functional requirements, together with business and business technical requirements needs that shape the platform's design and implementation specifications of the CoGNETs platform.

7.1 APPLICATION LAYER

7.1.1 Dashboard (UI)

The dashboard plays a central role in managing and monitoring system operations. To ensure secure and controlled access, it is essential to implement a robust authentication system. This section presents the requirements for Dashboard (UI). This component will provide access to user to request the execution of a CoGNETs AI module in a PUC. It will translate the request to the Distributed Service Manager (DSM) to launch the process to execute the CoGNETs AI. Lastly, it will provide a UI interface to visualize the status of the execution of the CoGNETs AI modules.

Funded by the European Union
Project funded by

Beyond security, another key aspect is notification management, which allows users configure their notification preferences directly from the dashboard. This will allow them to receive customized alerts regarding task completions and system anomalies. Notification settings will be highly flexible, enabling the system operator to define which events should trigger an alert and with what level of priority. Through integration with the Distributed Service Manager (DSM) and the Node Manager - Data Manager (NDM), the dashboard will ensure efficient and timely monitoring of operations, improving system responsiveness and optimizing resource management.

7.1.1.1 Functional Requirements (FNC)

Table 8: UI.FNC requirements

Req. Id	Requirement Description	
UI.FNC.001	The dashboard must permit the system operator near real-time tracking of active AI modules across the IoT-Edge-Cloud continuum to assure smooth performance and solve problems proactively. **Action**: Allow near real-time tracking. **Object**: Active AI modules. **Constraint**: Provide status updates with the minimum latency possible enabling near real-time updates. **Value**: Near real-time responsiveness.	
	Affected components	Dashboard (UI) Distributed Workload Manager (DWM) Node Manager - Device Monitoring (NDMo)
	Contributing Partner	MEDITECH, HMU
	Comment	Dashboard should view both visual (graphical) and numerical status updates for every module.
	Classification	Must Have (M)
	Related topic	IoT-Edge-Cloud swarm continuum architectures
UI.FNC.002	The user should be able to setup notification preferences using the dashboard to retrieve alerts for task completions and system anomalies. **Action**: set notification preferences. **Object**: Alerts and notifications. **Constraint**: enable configuration for email, SMS, or in-app notifications.	
	Affected components	Dashboard (UI) Distributed Service Manager (DSM) Node Manager - Data Manager (NDM)
	Contributing Partner	MEDITECH
	Comment	Notification configurations should be customizable for each activity or anomaly kind.
	Classification	Should Have (S)
UI.FNC.003	Proper access control mechanisms should guarantee that different roles and different users are only capable to access only what is appropriate. **Action**: Enforce secure authentication and authorization. **Object**: Access control-based user access.	

Funded by the European Union
Project funded by

Req. Id	Requirement Description	
	Constraint: Utilize IAM	
	Affected components	Dashboard (UI)
	Contributing Partner	UBITECH
	Comment	Keycloak can be used.
	Classification	Must Have (M)
UI.FNC.004	The Dashboard should include Explainable AI functionality by explaining autonomous swarm decisions. **Action**: Provide explanations for decisions taken autonomously from the swarm. **Object**: Pre-trained ML models. **Constraint**: Access to pre-trained ML models and test data is required. **Value**: Ensures interpretability and reliability of decisions.	
	Affected components	Dashboard (UI)
	Contributing Partner	K3Y
	Comment	-
	Classification	Must Have (M)

7.1.1.2 Non-Functional Requirements (NFN)

Table 9: UI.NFN requirements

Req. Id	Requirement Description	
UI.NFN.001	The dashboard must provide the minimum response time possible for all user actions, providing a reliable user experience. **Action**: Reliable responsiveness. **Object**: User actions on the dashboard. **Constraint/Value**: Maximum response time of 3 seconds.	
	Affected components	Dashboard (UI) Distributed Service Manager (DSM) Node Manager - Data Manager (NDM)
	Contributing Partner	MEDITECH, HMU
	Comment	Test response times with various user load scenarios.
	Classification	Must Have (M)
	Related topic	Scalability and adaptability mechanisms
UI.NFN.002	The dashboard must prop a high availability rate of 99.9%, maintaining continuous operation in time sensitive scenarios. **Action**: Provide high availability. **Object**: Dashboard uptime. **Constraint/Value**: Support uptime with a maximum downtime of 8.76 hours per year.	
	Affected components	Dashboard (UI)

Funded by the European Union
Project funded by

Req. Id	Requirement Description	
		Distributed Service Manager (DSM) Node Manager - Data Manager (NDM)
	Contributing Partner	MEDITECH
	Comment	Leverage extra servers and automated failover mechanisms.
	Classification	Must Have (M)
	Related topic	Scalability and adaptability mechanisms, data manageability

7.1.1.3 Business Requirements (BUS)

Table 10: UI.BUS requirements

Req. Id	Requirement Description	
UI.BUS.001	The dashboard should provide the business stakeholder summarized reports of system performance and operation execution to support efficient strategic decision-making. **Action**: Provide summarized performance reports. **Object**: System performance and operation execution data. **Constraint/Value**: Reports must be exportable in TXT, CSV, PDF, and DOCX formats.	
	Affected components	Dashboard (UI) Distributed Data Manager (DDM)
	Contributing Partner	MEDITECH, HMU
	Comment	Reports should be provided for daily, weekly, and monthly summaries.
	Classification	Should Have (S)
	Related topic	Scalability and adaptability mechanisms
UI.BUS.002	Support HMS integration: The dashboard must integrate with existing Hospital Management System (HMS) to simplify the clinical workflows for the healthcare organization. **Action**: Support HMS integration. **Object**: Hospital management systems. **Constraint/Value**: Utilise standard APIs like HL7 or FHIR for integration seamlessly.	
	Affected components	Dashboard (UI) Cognitive AI Service Repository (CSR)
	Contributing Partner	MEDITECH
	Comment	Assure compliance with healthcare data standards.
	Classification	Must Have (M)
	Related topic	Data manageability, cognitive computing & programming models
UI.BUS.003	The dashboard must visualize all important KPIs of a vehicle's Thermal Manage-	

Funded by the European Union
Project funded by

	ment System (TMS) which have impact to overall powertrain performance and passengers' comfort. Can be utilized for final validation. **Action**: Support TMS integration. **Object**: Thermal Management Systems. **Constraint/Value**: None identified	
	Affected components	Dashboard (UI) Cognitive AI Service Repository (CSR)
	Contributing Partner	AVL
	Comment	Guarantee compliance with European regulations and healthcare data standards.
	Classification	Must Have (M)

7.1.1.4 Business Business Technical Requirements (BTC)

Table 11: UI.BTC requirements

Req. Id	Requirement Description	
UI.BTC.001	The dashboard must utilise a secure and robust authentication system, guaranteeing that just authorized users can access its features. **Action**: Enforce secure authentication. **Object**: Dashboard user access. **Constraint/Value**: Must support multi-factor authentication (MFA).	
	Affected components	Dashboard (UI) DevOps Platform (DOP)
	Contributing Partner	MEDITECH
	Comment	Support SSO (Single Sign-On) for enterprise users.
	Classification	Must Have (M)
	Related topic	Swarm-wise distributed security paradigms
UI.BTC.002	The user should be able to setup notification preferences using the dashboard to retrieve alerts for task completions and system anomalies. **Action**: Support encrypted communication. **Object**: UI and back-end communication. **Constraint/Value**: Utilise TLS 1.3 or higher for encryption.	
	Affected components	Dashboard (UI) Distributed Data Manager (DDM) Middleware Layer
	Contributing Partner	MEDITECH
	Comment	Maintain continuous vulnerability measurements to guarantee encryption protocols stay secure.
	Classification	Must Have (M)

Funded by the European Union
Project funded by

7.1.2 Cognitive AI Service Repository (CSR)

The **Cognitive AI Service Repository (CSR)** is responsible for the articulation, storage, and management of **CoGNETs AI service algorithms**. It maintains a metadata description of these algorithms, enabling their organization and retrieval. CSR connects to the **Distributed Service Manager**, supporting both **CNN/DNN models** and **Q-Learning implementations**. To standardize representation, a **Data Model** must be defined, mapping CNN/DNN and Q-Learning structures to a **Directed Decision Acyclic Graph (DDAG)**.

7.1.2.1 Functional Requirements (FNC)

Table 12: CSR.FNC requirements

Req. Id	Requirement Description	
CSR.FNC.001	CSR must allow the developer to store and retrieve AI service algorithms with metadata, such that the repository can aid fast and precise searches for CoGNETs AI modules. **Action**: Store and retrieve algorithms **Object**: CoGNETs AI service algorithms and metadata **Constraint/Value**: Ensure the retrieval process completes within 4 seconds for datasets up to 2,000 entries.	
	Affected components	Cognitive AI Service Repository (CSR) Distributed Service Manager
	Contributing Partner	MEDITECH, ULANCS
	Comment	Essential for facilitating efficient repository tasks and assuring smooth service deployment.
	Classification	Must Have (M)
	Related topic	Cognitive computing & programming models
CSR.FNC.002	An CSR manager / developer / AI model developer should be able to backtrace individual AI models in order: (i) to be able to view the AI models evolution (ii) to be able to compare / evaluate efficiency of the evolution of individual AI models (iii) to be able to switch between AI models swiftly. **Action**: Being able to assess updates / evolution of a given AI model **Object**: CoGNETs AI service algorithms and metadata **Constraint**: The information about AI models should be stored in a well defined and clear structure. **Value**: The functionality should enable an easy and swift overview of current and past AI models along with their specifications.	
	Affected components	Cognitive AI Service Repository (CSR) Dashboard (UI)
	Contributing Partner	Beyond
	Comment	-
	Classification	Should Have (S)
	Related topic	Scalability and adaptability mechanisms

Funded by the European Union
Project funded by

CSR.FNC.003	An AI model developer/user must input the NN-structure, or a split version of the NN-structure, that is intended to be deployed in the different hierarchical nodes for training/inference. **Action**: Provide a way to input NN-structures to be deployed in the different tiers of the hierarchical topology of the network. **Object**: CSR infrastructure. **Constraint/Value**: Conformation to a predefined format to ensure compatibility with the CSR infrastructure.	
	Affected components	Distributed Service Manager (DSM)
	Contributing Partner	VTT
	Comment	In an initial development stage it is proposed to split manually the NN and input to the CSR the different split parts.
	Classification	Must Have (M)
CSR.FNC.004	CSR could include explainability metadata to enhance understanding of ML model predictions. **Action**: Store explainability metadata for ML models. tiers of the hierarchical topology of the network. **Object**: Explanations of pre-trained ML models. **Constraint**: Metadata must be structured for efficient retrieval. **Value**: Enhances interpretability and transparency of model predictions.	
	Affected components	Cognitive AI Service Repository (CSR) Dashboard (UI)
	Contributing Partner	K3Y
	Comment	-
	Classification	Could Have (C)
	Related topic	Federated Learning mechanisms

7.1.2.2 Non-Functional Requirements (NFN)

Table 13: CSR.NFN requirements

Req. Id	Requirement Description	
CSR.NFN.001	The Cognitive AI Service Repository must support concurrent requests from up to 200 users without performance degradation. **Action**: Support high-concurrency requests **Object**: CSR infrastructure **Constraint/Value:** Ensure response time remains under 3 seconds for 90% of requests	
	Affected components	Cognitive AI Service Repository (CSR) Distributed Resource Manager (DRM) Distributed Service Manager (DSM)
	Contributing Partner	MEDITECH

Funded by the European Union
Project funded by

	Comment	Essential to assure the system can scale to handle multiple developers or end-users accessing AI services simultaneously.
	Classification	Must Have (M)
	Related topic	Scalability and adaptability mechanisms
CSR.NFN.002	The CSR should be able to power the same service with multiple AI models in order for service to apply several AI model at the same time within the single service. In such way, updates are introduced in a more controlled way (perhaps this demand should be listed on the service specification list, however, by placing it here we are addressing the functionality more consistently). **Action**: Being able to gradually introduce new / updated AI model into service **Object**: CoGNETs AI service algorithms and metadata **Constraint/Value**: the service should be able to handle the feature	
	Affected components	Cognitive AI Service Repository (CSR) Distributed Service Manager (DSM)
	Contributing Partner	Beyond, ULANCS
	Comment	-
	Classification	Could Have (C)
	Related topic	Scalability and adaptability mechanisms
CSR.NFN.003	The AI model developers should be aware of multiple AI models. So when an updated AI model is shipped, CSR should retain all the historically available AI model versions. Individual AI model should be called through group Id and a tag Id, so when the AI model is updated, one needs to update the group Id only (as tag remains the same). **Action**: the ability to select predefined AI model to be served when required **Object**: Being able to easily access any (current or historical) AI model **Constraint**: The information about AI models should be stored in a well defined and clear structure. **Value**: The selection of the AI model, that will be served in relation to specific service, should be done easily and swiftly by the end user.	
	Affected components	Cognitive AI Service Repository (CSR) Dashboard (UI)
	Contributing Partner	Beyond
	Comment	-
	Classification	Should Have (S)
	Related topic	Scalability and adaptability mechanisms

7.1.2.3 Business Requirements (BUS)

Table 14: CSR.BUS requirement

Req. Id	Requirement Description
CSR.BUS.001	When serving AI models, the CSR should be containing several available (not his-

Funded by the European Union
Project funded by

	torical, but up-to-date) versions of a given individual AI model (imagine a group of AI models not a single AI models ready for a service). Several version would enable business a pricing diversity and targeting various customer segments. **Action**: enable business to grow pricing strategies / address various segments **Object**: CoGNETs AI service algorithms and metadata **Constraint**: the CSR should enable AI models grouping and tagging. All future AI updates should be aware of and follow the initial group / tag system.	
	Affected components	Cognitive AI Service Repository (CSR) Dashboard (UI) Distributed Service Manager (DSM)
	Contributing Partner	BEYOND
	Comment	-
	Classification	Should Have (S)
	Related topic	Scalability and adaptability mechanisms

7.1.2.4 Business Technical Requirements (BTC)

Table 15: CSR.BTC requirement

Req. Id	Requirement Description	
CSR.BTC.001	The Cognitive AI Service Repository should integrate with version control systems like Git and CI/CD tools to ensure automated updates and synchronization of AI algorithms. **Action**: Enable integration **Object**: Git, Jenkins **Constraint/Value:** Ensure synchronization occurs within 5 minutes of a new commit or algorithm update.	
	Affected components	Cognitive AI Service Repository (CSR) DevOps Platform (DOP)
	Contributing Partner	MEDITECH, ULANCS
	Comment	Facilitates continuous improvement and updates of AI services to comply with the evolving pilot use case needs.
	Classification	Should Have (S)
	Related topic	Federated Learning mechanisms

7.1.3 DevSecOps platform (DOP)

The CoGNETs' DevSecOps platform is a solution designed to deploy and manage Middleware components by automating and optimizing development processes. It embeds security practices throughout the development lifecycle, supporting efficient software updates through automated CI/CD pipelines, while minimizing development time and operational costs. The

Funded by the European Union
Project funded by

platform integrates with container and orchestration tools such as Docker and Jenkins, includes secure authentication, code quality checks and generates detailed logs all accessible from its central monitoring tool. This facilitates the efficient deployment of CoGNETs services across distributed infrastructure, ensuring secure and efficient operations.

7.1.3.1 Functional Requirements (FNC)

Table 16: DOP.FNC requirements

Req. Id	Requirement Description	
DOP.FNC.001	As a platform administrator, I want to automate the deployment of Middleware components across IoT, Edge, and Cloud layers, so that the system can support seamless scaling and dynamic resource allocation. **Action**: Automate component deployment. **Object**: Middleware components **Constraint/Value:** Ensure compatibility across heterogeneous devices/components	
	Affected components	DevOps Platform (DOP)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
DOP.FNC.002	As a developer, I want to enable automated CI/CD pipelines for CoGNETs services, so that software updates and patches can be deployed efficiently across the swarm. **Action**: Design and implement CI/CD Pipelines. **Object**: All deployed components are required to be automatically configured using and upgraded by the CI/CD pipelines. **Constraint/Value:** Upgrade the pipeline keeping integrity with a low interruption of services ≤150ms	
	Affected components	All software components deployed in K3S nodes and CoGNETs Swarm.
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)

7.1.3.2 Non-Functional Requirements (NFN)

Table 17: DOP.NFN requirements

Req. Id	Requirement Description
DOP.NFN.001	The DevOps platform should support concurrent builds and deployments for at least 50 nodes simultaneously, ensuring no more than a 5% performance degradation. **Action**: Implement a scalable DevOps platform configuration. **Object**: CoGNETs builds and deployments.

Funded by the European Union
Project funded by

	Constraint: Maximum 5% performance degradation during concurrent deployments. **Value:** Support for large-scale swarms.	
	Affected components	DevOps Platform (DOP)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Could Have (C)
DOP.NFN.002	The platform must maintain an availability of 99.9% to ensure continuous operation across critical IoT-to-Cloud applications. **Action**: Ensure high-availability infrastructure setup. **Object**: Platform infrastructure components. **Constraint:** Maximum allowable downtime of 8.76 hours/year (or 43.8 minutes/month). **Value:** High availability to support mission-critical IoT-Cloud applications without significant service disruptions.	
	Affected components	DevOps Platform (DOP)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Should Have (S)

7.1.3.3 Business Requirements (BUS)

Table 18: DOP.BUS requirements

Req. Id	Requirement Description	
DOP.BUS.001	As a business stakeholder, I want the DevOps platform to minimize deployment time and operational costs, so that resources are utilized efficiently without compromising service quality. **Action**: Minimize time and cost **Object**: Deployment and operation processes. **Value:** Cost-effectiveness and operational efficiency.	
	Affected components	DevOps Platform (DOP)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
DOP.BUS.002	As a project manager, I want the platform to generate detailed logs and reports of deployment activities, so that compliance with organizational and legal standards can be ensured. **Action**: Generate deployment logs **Object**: Logs and reports **Constraint:** Reports must align with EU data privacy regulations.	
	Affected components	DevOps Platform (DOP)

Funded by the European Union
Project funded by

	Contributing Partner	FIWARE
	Comment	-
	Classification	Could Have (C)

7.1.3.4 Business Technical Requirements (BTC)

Table 19: DOP.BTC requirements

Req. Id	Requirement Description	
DOP.BTC.001	The DevOps platform should integrate with existing tools like Docker, and Jenkins, enabling containerized and orchestrated deployments of CoGNETs AI services. **Action**: Support integration **Object**: Docker, Jenkins **Constraint:** Ensure seamless compatibility with tools	
	Affected components	DevOps Platform (DOP)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
DOP.BTC.002	The platform must include a secure authentication mechanism for developers and administrators to access deployment pipelines and logs. **Action**: Provide secure authentication. **Object**: Deployment pipelines and logs. **Constraint**: Use of JWT and DID.	
	Affected components	DevOps Platform (DOP)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Could Have (C)
DOP.BTC.003	The platform's logs should be searchable in order to easily track system behaviour as well as quickly find potential issues. **Action**: Provide Full Text Search Capability. **Object**: ELK Stack. **Constraint**: Data retention policies & Data anonymization.	
	Affected component	DevOps Platform (DOP)
	Contributing Partner	INTRA
	Comment	-
	Classification	Could Have (C)

Funded by the European Union
Project funded by

7.2 MIDDLEWARE LAYER – SWARM CONTEXT

7.2.1 Swarm AI Game Agent (SGA)

Swarm AI Game Agent (SGA) simulates group behaviours inspired by nature, where individual agents follow simple rules that result in complex, coordinated actions. Each agent interacts with its environment and others, creating emergent behaviour. Swarm AI is used in games for enemy AI, NPC behaviour, pathfinding, and large-scale bot control. It's efficient, scalable, and adaptable, offering dynamic challenges for players. Popular techniques include Boids for flocking behaviour, Ant Colony Optimization for pathfinding, and Particle Swarm Optimization for collaborative tasks.

7.2.1.1 Functional Requirements (FNC)

Table 20: SGA.FNC requirements

Req. Id	Requirement Description	
SGA.FNC.001	As a swarm AI game agent, I want to collect real-time information about resources and workload status from each swarm node, so that bidding optimization can be performed with accuracy. **Action**: Collect research information **Object**: Real-time node information (i.e., resource usage, workload) **Constraint/Value**: Ensure accuracy of collected data with a latency less than 100ms per node	
	Affected components	Swarm AI Game Agent (SGA) Node Manager – Device Monitoring (NDMo)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
SGA.FNC.002	As a swarm AI game agent, I want to interact with the DSM to update the execution of CoGNETs AI services, so that bidding decisions are synchronized across the swarm. **Action**: Interact with the Distributed Service Manager **Object**: Distributed Service Manager **Constraint/Value**: Ensure synchronization latency remains below 150ms	
	Affected components	Swarm AI Game Agent (SGA) Distributed Service Manager (DSM)
	Contributing Partner	CERTH, ULANCS
	Comment	-
	Classification	Must Have (M)
SGA.FNC.003	As a swarm AI game agent, I want to analyse bidding results and provide actionable network optimization strategies, so that each node can autonomously participate in dynamic swarm activities. **Action**: Analyse bidding results and provide optimization strategies	

Funded by the European Union
Project funded by

	Object: Bidding results and actionable network optimization strategies **Constraint/Value**: Ensure node decisions are update within 100ms of game result availability	
	Affected components	Swarm AI Game Agent (SGA) Node AI Game Agent (NGA)
	Contributing Partner	CERTH, ULANCS, UAVIGN
	Comment	-
	Classification	Should Have (S)
SGA.FNC.004	As a swarm AI game agent, I want to validate that a swarm has at least 3 participating nodes before initiating bidding, so that bidding processes are meaningful and efficient. **Action**: Validate number of participating nodes in the swarm **Object**: Number of participating nodes in the swarm **Constraint/Value**: Ensure that the swarm contains at least 3 nodes	
	Affected components	Swarm AI Game Agent (SGA) Node AI Game Agent (NGA)
	Contributing Partner	CERTH, UAVIGN
	Comment	-
	Classification	Must Have (M)
SGA.FNC.005	The Swarm AI Game Agent (SGA), when receiving input from the Distributed Service Manager (DSM), must process game results to deter-mine optimal device actions in dynamic swarms. **Action**: Interpret game results (Action) to generate network function updates **Object**: Network function updates **Constraint/Value**: Maintain computational latency below 50ms and energy usage within a 10% tolerance of predefined node limits	
	Affected components	Swarm AI Game Agent (SGA)
	Contributing Partner	MEDITECH
	Comment	Assure scalability of the SGA for processing up to 1000 nodes in a swarm without notable degradation in performance.
	Classification	Must Have (M)
SGA.FNC.006	The Swarm AI Game Agent (SGA), when receiving input from the Distributed Service Manager (DSM), must receive the prices associated to bidding for resources. **Action**: Ensure the SGA receives pricing information for resource bidding. **Object**: Pricing data provided by the DSM. **Constraint/Value**: Predefined structure of the pricing information within DSM's communication payload.	
	Affected components	Swarm AI Game Agent (SGA)
	Contributing Partner	UAVIGN

Funded by the European Union
Project funded by

	Comment	Ensure the prices lead to efficient game optimization
	Classification	Must Have (M)

7.2.1.2 Non-Functional Requirements (NFN)

Table 21: SGA.NFN requirements

Req. Id	Requirement Description
SGA.NFN.001	As a swarm AI game agent, I want the data collection process to have a maximum delay of 120ms per node, so that bidding decisions remain timely and accurate. **Action**: Ensure the collection process is immediate. **Object**: Data collection latency. **Constraint/Value**: Latency must be less than 120ms per node.
	Affected components: Swarm AI Game Agent (SGA) Node Manager - Device Monitoring (NDMo)
	Contributing Partner: CERTH
	Comment: -
	Classification: Must Have (M)
	Related Topics: Scalability and adaptability mechanisms
SGA.NFN.002	As a swarm AI game agent, I want to scale data collection and bidding optimization to support swarms with up to 200 nodes, so that the system remains efficient in larger deployments. **Action**: Scale data collection and bidding optimization. **Object**: Data collection and bidding optimization. **Constraint**: Ensure scalability up to 200 nodes. **Value**: Efficient system capability with minimum performance degradation.
	Affected components: Swarm AI Game Agent (SGA)
	Contributing Partner: CERTH, ULANCS, UAVIGN
	Comment: -
	Classification: Could Have (C)
	Related Topics: Game optimization strategies
SGA.NFN.003	The Swarm AI Game Agent (SGA) under the Middleware Layer (Swarm Context) must ensure consistent and low-latency communication with all participating nodes in the swarm network. **Action:** Establish an optimized data exchange mechanism **Object:** Game state updates and decision dissemination **Constraint/Value:** Latency ≤ 50ms under normal operating conditions, scalability up to 100 nodes without significant performance degradation. **Value:** Ensure seamless real-time communication, maintaining decision consistency across all nodes while preventing network bottlenecks.
	Affected components: Middleware Layer (Swarm Context)

Funded by the European Union
Project funded by

		Swarm AI Game Agent (SGA) Distributed Service Manager (DSM) Distributed Data Manager (DDM)
	Contributing Partner	MEDITECH
	Comment	This requirement assures that the SGA maintains optimal performance while handling dynamic swarm compositions and supports real-time game decision-making operation.
	Classification	Must Have (M)

7.2.1.3 Business Requirements (BUS)

Table 22: SGA.BUS requirements

Req. Id	Requirement Description	
SGA.BUS.001	As a business stakeholder, I want the SGA to improve network efficiency through game-based (bidding) resource allocation, so that operational costs are minimized while maintaining performance. **Action**: Optimize network efficiency through game-based allocation. **Object**: Network efficiency. **Constraint**: Ensure cost reductions without sacrificing performance. **Value**: Enhanced performance with minimal cost.	
	Affected components	Swarm AI Game Agent (SGA) Distributed Resource Manager (DRM)
	Contributing Partner	CERTH, ULANCS, UAVIGN
	Comment	-
	Classification	Must Have (M)
	Related Topics	Scalability and adaptability mechanisms
SGA.BUS.002	As a system administrator or stakeholder, I want SGAs to dynamically evaluate and resolve resource conflicts in the IoT-to-Cloud swarm, so that the system achieves optimal resource utilization. **Action**: Dynamically evaluate and resolve resource conflicts. **Object**: Resource allocation across IEC swarm nodes. **Constraint/Value**: System-wide efficiency.	
	Affected components	Swarm AI Game Agent (SGA) Distributed Workload Manager (DWM) Distributed Data Manager (DDM) Distributed Resource Manager (DRM) Distributed Service Manager (DSM) (swarm level) Node AI Game Agent (NGA) Node Manager - Device Monitoring (NDMo) Node Manager - Workload Orchestrator (NWO) Node Manager - Data Manager (NDM) (node level) Software-level Security (S-SL)

Funded by the European Union
Project funded by

		AI-level Security (S-AI) (security)
	Contributing Partner	CERTH
	Comment	-
	Classification	Should Have (S)
	Related Topics	Game optimization strategies IEC swarm continuum architectures

7.2.1.4 Business Technical Requirements (BTC)

Table 23: SGA.BTC requirement

Req. Id	Requirement Description	
SGA.BTC.001	As a swarm AI game agent, I want to integrate game theory-based models (e.g., auction-based algorithms), so that resource allocation decisions are optimized. **Action**: Integrate game theory-based models **Object**: Game theory-based algorithms **Constraint/Value**: Optimized and automated resource allocation	
	Affected components	Swarm AI Game Agent (SGA) Node AI Game Agent (NGA)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)

7.2.2 Distributed Resource Manager (DRM)

The requirements for the Distributed Resource Manager (DRM) are presented in this section. It will be responsible for creating the Trust List of Nodes. It will provide Decentralized Id Management and Identity Secrets Management of a new node and coordinate the initialization of the Nodes to deploy the requested services including Device Monitoring, Device Registration, Device Storage, Data Manager, Workload Orchestrator and Component Executor.

7.2.2.1 Functional Requirements (FNC)

Table 24: DRM.FNC requirements

Req. Id	Requirement Description
DRM.FNC.001	As a DRM, I want to generate and keep track for all the IoT-to-Cloud Swarms and Node Contexts available at every moment. This information must be distributed among all the nodes. **Action**: Keep track of IoT-to-Cloud Swarm and node contexts. **Object**: N/A. **Constraint/Value**: Keep information about Swarms and Node Contexts infra-

Funded by the European Union
Project funded by

	structures consistent.	
	Affected components	Distributed Resource Manager (DRM) Distributed Data Manager (DDM)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
DRM.FNC.002	As a DRM, I want build in some distributed storage with the information that will be offered by Node Contexts of each device. This information will be kept up to date via consensus algorithms. **Action**: Keep distributed information consistent. **Object**: Node-Context information. **Constraint/Value**: Provide discovery mechanisms.	
	Affected components	Distributed Resource Manager (DRM) Distributed Data Manager (DDM)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
DRM.FNC.003	The Distributed Resource Manager (DRM) shall manage and maintain a Trust List of Nodes for the swarm environment. DRM must dynamically update the Trust List of Nodes by validating new nodes through decentralized identity mechanisms, ensuring they comply with predefined security and performance metrics. Updates to the Trust List must occur within 10 seconds of node validation to maintain real-time system integrity. **Action**: Dynamically update the Trust List of Nodes by validating new nodes. **Object**: Trust List of Nodes. **Constraint**: Validation must be performed using decentralized identity mechanisms, ensuring compliance with predefined security and performance metrics. **Value**: Updates to the Trust List must occur within 10 seconds of node validation to maintain real-time system integrity.	
	Affected components	Distributed Resource Manager (DRM) Node Manager - Device Registration (NDR)
	Contributing Partner	MEDITECH
	Comment	This requirement guarantees smooth nodes integration with the swarm environment while providing strong security and performance.
	Classification	Must Have (M)
	Related topic	IoT-Edge-Cloud swarm continuum architectures, Swarm-wise distributed security paradigms

Funded by the European Union
Project funded by

7.2.2.2 Non-Functional Requirements (NFN)

Table 25: DRM.NFN requirements

Req. Id	Requirement Description	
DRM.NFN.001	As Distributed Resource Manager user, I want to operate securely across nodes, ensuring confidentiality. **Action:** Ensure privacy. **Object:** Distributed Resource Manager (DRM). **Constraint/Value:** Optimized nodes to ensure security.	
	Affected components	Distributed Resource Manager (DRM) Distributed Data Manager (DDM)
	Contributing Partner	CERTH
	Comment	Scalability and adaptability mechanisms.
	Classification	Must Have (M)
DRM.NFN.002	**Security**: The Distributed Resource Manager (DRM) must ensure secure and efficient initialization of nodes under fluctuating network conditions. **Action**: Process node initialization and identity management leveraging decentralized protocols **Object**: Node initialization and identity management. **Constraint/Value**: Must ensure synchronization with the network plane with a response time of ≤ 250ms for up to 500 nodes.	
	Affected components	Distributed Resource Manager (DRM) Node Manager - Device Monitoring (NDMo) Node Manager - Device Registration (NDR)
	Contributing Partner	MEDITECH
	Comment	This requirement emphasizes security and scalability to support real-time processes in distributed sittings.
	Classification	Should Have (S)
	Related topic	IoT-Edge-Cloud swarm continuum architectures, Swarm-wise distributed security paradigms, Data manageability.

7.2.2.3 Business Requirements (BUS)

Table 26: DRM.BUS requirements

Req. Id	Requirement Description
DRM.BUS.001	As a DRM, I need that all information available to the DRM is synchronized among all the nodes in a close to zero time. **Action:** Process and synchronize information as fast as possible, in a close to zero time. **Object:** Node synchronization in the Network plane. **Constraint/Value:** Synchronization must be close to zero time ≤ 250ms

Funded by the European Union
Project funded by

	Affected components	Distributed Resource Manager (DRM)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Should Have (S)
DRM.BUS.002	As a DRM, I need to ensure high availability. **Action:** Availability must be close to "always". **Object:** Node synchronization in the Network plane. **Constraint/Value:** Availability of the DRM must be 100%	
	Affected components	Distributed Resource Manager (DRM)
	Contributing Partner	FIWARE/MEDITECH
	Comment	DRM is an important piece of software for CoGNETs that must be up and running always to ensure the proper functioning of CoGNETs.
	Classification	Must have (M)
DRM.BUS.003	As a DRM, I need to scale so I can manage resource in many nodes. **Action:** Scale properly according to the needs of the nodes connected to the CoGNETs Swarm. **Object:** Node synchronization in the Network plane. **Constraint/Value:** Scale to at least 500 nodes keeping performance.	
	Affected components	Distributed Resource Manager (DRM)
	Contributing Partner	FIWARE/MEDITECH
	Comment	Scalability shouldn't diminish the availability and the performance requirements.
	Classification	Should have (S)
DRM.BUS.004	As a DRM, I expect security in my use. **Action:** Protect the DRM with security components. **Object:** Security components and DRM. **Constraint/Value:** Set a set of security policies in DRM utilization to provide security to the DRM. 0% of unauthenticated/unauthorized requests can be accepted.	
	Affected components	Distributed Resource Manager (DRM) Security components
	Contributing Partner	FIWARE/MEDITECH
	Comment	A number of security implementation should be put in place.
	Classification	Must have (M)
DRM.BUS.005	As a DRM, I a good and intuitive console to be configured and managed. **Action:** Design and build an intuitive console to manage the DRM. **Object:** DRM management and configuration. **Constraint/Value:** Have a good and intuitive console to configure and manage the DRM.	
	Affected components	Distributed Resource Manager (DRM)

Funded by the European Union
Project funded by

		Dashboard (UI)
	Contributing Partner	FIWARE/MEDITECH
	Comment	The console must have security and the DRM must be protected against undesired access through the console.
	Classification	Should have (S)

7.2.2.4 Business Technical Requirements (BTC)

Table 27: DRM.BTC requirement

Req. Id	Requirement Description	
DRM.BTC.001	As a Distributed Resource Manager user, I want the DRM to be scalable. **Action**: Optimize nodes. **Object**: Distributed Resource Manager. **Constraint/Value**: To handle increasing numbers of nodes and services effectively.	
	Affected components	Distributed Resource Manager (DRM) Distributed Data Manager (DDM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)

7.2.3 Distributed Service Manager (DSM)

This section outlines the requirements for the Distributed Service Manager (DSM) which is targeted to constitute the Swarm/Node Contexts functionalities. It is also responsible of the synchronization of the activities Workload and Data through the DDAG with the nodes.

7.2.3.1 Functional Requirements (FNC)

Table 28: DSM.FNC requirements

Req. Id	Requirement Description	
DSM.FNC.001	As DSM, I want to get the results of the integrated Game Agents and fit the services to be run in the most suitable resources available. **Action**: Fit Service to Node. **Object**: Service manager, IoT-to-Cloud devices. **Constraint/Value:** Efficient allocation of services on resources.	
	Affected components	Distributed Service Manager (DSM) Swarm AI Game Agent (SGA) Distributed Workload Manager (DWM) Node Manager - Workload Orchestrator (NWO)

Funded by the European Union
Project funded by

	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
DSM.FNC.002	As DSM, I want to be able to manage the lifecycle execution and termination by interacting with the Distributed workload orchestrator. **Action**: Manage service execution. **Object**: Service manager, IoT-to-Cloud devices. **Constraint/Value:** Efficient management of resources in the Swarm.	
	Affected components	Distributed Service Manager (DSM) Distributed Workload Manager (DWM) Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
DSM.FNC.003	As DSM, I want to take self-management actions in case the expected minimum QoS isn't achieved or in case a failure is detected in an executing resource at the management of a node. **Action**: Manage service execution. **Object**: Service manager, IoT-to-Cloud devices. **Constraint/Value:** Efficient management of resources in the Swarm.	
	Affected components	Distributed Service Manager (DSM) Distributed Workload Manager (DWM) Node Manager - Workload Orchestrator (NWO) Distributed Resource Manager (DRM)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
DSM.FNC.004	Distributed Service Manager (DSM) must ensure synchronization between Node Managers when a new CoGNETs AI service is requested. The DSM will communicate with Node Managers via the DDAG to synchronize workload and data updates. Constraint: Synchronization must occur within 100ms to maintain real-time service performance. **Action:** Communicate with Node Managers via the DDAG to synchronize workload and data updates. **Object:** Workload and data updates. **Constraint:** Synchronization must occur within 100ms. **Value:** Maintain real-time service performance.	
	Affected components	Node Manager - Device Monitoring (NDMo) Node Manager - Device Registration (NDR) Node Manager - Device Storage (NDS) Node Manager - Workload Orchestrator (NWO) Node Manager - Data Manager (NDM)
	Contributing Partner	MEDITECH
	Comment	Ensure the synchronization mechanism accounts for

Funded by the European Union
Project funded by

		heterogeneous hardware and varying network conditions.
	Classification	Must Have (M)

7.2.3.2 Non-Functional Requirements (NFN)

Table 29: DSM.NFN requirement

Req. Id	Requirement Description	
DSM.NFN.001	Distributed Service Manager (DSM) must provide reliable communication between swarm nodes under variable network conditions. The DSM will utilize fault-tolerant communication protocols to ensure data consistency in distributed environments. **Action:** Utilize fault-tolerant communication protocols. **Object:** Data consistency in distributed environments. **Constraint/Value:** Ensure reliability in variable network conditions. The reliability should exceed 99.9% uptime in normal operations.	
	Affected components	Middleware Layer (Swarm Context) Node Manager - Data Manager (NDM) Distributed Data Manager (DDM) Distributed Resource Manager (DRM)
	Contributing Partner	MEDITECH
	Comment	Consider using advanced protocols like gRPC or MQTT to enhance fault tolerance and reduce latency.
	Classification	Should Have (S)

7.2.3.3 Business Requirements (BUS)

Table 30: DSM.BUS requirement

Req. Id	Requirement Description	
DSM.BUS.001	As a system user, I want devices in the IoT-to-Cloud swarm to autonomously collaborate in real time using intelligent mechanisms, so that diverse AI tasks can be handled efficiently based on current resource availability and task requirements. **Action:** Enable autonomous collaboration. **Object:** IoT-to-Cloud devices. **Constraint:** Real-time collaboration using intelligent mechanisms. **Value:** Efficient AI task-handling based on resource availability.	
	Affected components	Distributed Service Manager (DSM) Distributed Resource Manager (DRM) Distributed Workload Manager (DWM) Node Manager – Workload Orchestrator (NWO)
	Contributing Partner	CERTH

Funded by the European Union
Project funded by

	Comment	-
	Classification	Must Have (M)

7.2.3.4 Business Technical Requirements (BTC)

Table 31: DSM.BTC requirement

Req. Id	Requirement Description	
DSM.BTC.001	The possibility to coordinate multiple swarms for a common goal, such as co-ordinated training in a Federated/Split Learning way, may be required. **Action:** Allow multiple and coordinated swarms per service. **Object:** **F**lexibility to combine into a service multiple swarms (note that in the case of FL or split learning, different swarms will converge to a common node in the core-cloud tier). **Constraint/Value:** Dynamic coordination of multiple swarms converging at a common node, enabling collaborative model training while preserving swarm autonomy.	
	Affected components	Distributed Service Manager (DSM)
	Contributing Partner	VTT, AXON
	Comment	-
	Classification	Should Have (S)

7.2.4 Distributed Workload Manager (DWM)

The Distributed Workload Manager component is responsible for assigning AI modules to specific nodes within the CoGNETs network, based on the outcomes provided by the Swarm Game Intelligent Agent, to ensure optimal distribution of AI workloads across nodes. This component allows for optimized resource allocation and system performance.

7.2.4.1 Functional Requirements (FNC)

Table 32: DWM.FNC requirements

Req. Id	Requirement Description	
DWM.FNC.001	As a workload orchestrator, I want to assign CoGNETs AI modules to the appro-priate nodes based on their available resources and capabilities, so that the AI workloads are executed efficiently and with minimum delay. **Action:** Assign AI modules to the appropriate nodes. **Object:** CoGNETs AI modules. **Constraint:** Assign the module to the node with the highest available resources, considering computational power, energy and security constraints.	
	Affected components	Distributed Workload Manager (DWM)
	Contributing Partner	CERTH

Funded by the European Union
Project funded by

	Comment	-
	Classification	Must Have (M)
DWM.FNC.002	As a workload orchestrator, I want to schedule the execution of CoGNETs AI modules across nodes based on priority, available resources and time sensitivity, so that the AI modules are executed in time and efficiently. **Action:** Schedule the execution of the AI modules. **Object:** CoGNETs AI modules. **Constraint:** Ensure that the AI module is scheduled according to priority levels, resource availability and deadlines.	
	Affected components	Distributed Workload Manager (DWM) Distributed Service Manager (DSM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Should Have (S)
DWM.FNC.003	The Distributed Workload Manager (DWM) must assign AI modules to swarm nodes dynamically based on real-time resource data. **Action:** Assign AI modules to swarm nodes dynamically. **Object:** AI modules and swarm nodes. **Constraint/Value:** Workload distribution must not exceed node computational capacity. **Value:** Real-time resource data	
	Affected components	Node Manager - Workload Orchestrator (NWO) Distributed Workload Manager (DWM) Swarm AI Game Agent (SGA)
	Contributing Partner	MEDITECH
	Comment	Essential for resource management and balancing over swarm nodes efficiently.
	Classification	Must Have (M)
DWM.FNC.004	The DWM must be able to translate the requested execution of tasks to Kubernetes PODs, manage the lifecycle of tasks and services executed by each node by relying on existing baseline to deploy services in Node contexts enabled by Kubernetes. **Action**: Allocate resources as PODs in Kubernetes clusters and manage their lifecycle based on SGA calculations **Object**: Deployment and lifecycle management of Kubernetes PODs for executing tasks and services **Constraint**: Align with the execution plan defined by the SGA and comply with existing Kubernetes-based service deployment baselines **Value**: Ensure efficient resource utilization and service execution within Kubernetes-managed node contexts	
	Affected components	Distributed Workload Manager (DWM) Swam AI Game Agent (SGA)
	Contributing Partner	FIWARE
	Comment	-

Funded by the European Union
Project funded by

	Classification	Must Have (M)

7.2.4.2 Non-Functional Requirements (NFN)

Table 33: DWM.NFN requirements

Req. Id	Requirement Description	
DWM.NFN.001	As a workload orchestrator, I want the system to assign workloads to nodes with a latency of less than 100ms, so that the system remains responsive and efficient. **Action:** Ensure efficient workload assignment **Object:** Latency for workload assignment **Constraint:** System responsiveness **Value:** Latency below 100ms	
	Affected components	Distributed Workload Manager (DWM) Distributed Service Manager (DSM)
	Contributing Partner	CERTH
	Comment	Ensures that the platform remains responsive to changes in workload demand.
	Classification	Must Have (M)
DWM.NFN.002	As a workload orchestrator, I want to ensure that workloads are distributed across nodes with varying resource capabilities, so that the entire swarm operates efficiently without overloading any single node. **Action:** Distribute workloads across swarm nodes **Object:** AI workloads **Constraint:** No single node is overloaded **Value:** Evenly distributed workloads based on each node's capabilities	
	Affected components	Distributed Workload Manager (DWM) Node Manager – Workload Orchestrator (NWO) Node Manager - Device Monitoring (NDMo)
	Contributing Partner	CERTH
	Comment	Ensures that the platform remains responsive to changes in workload demand.
	Classification	Must Have (M)
DWM.NFN.003	The Distributed Workload Manager (DWM) must process workload allocation requests within a determined time limit. **Action**: Assure workload allocation latency. **Object**: Workload allocation latency. **Constraint**: Must not exceed 50 milliseconds. **Value**: Normal operating settings.	
	Affected components	Distributed Workload Manager (DWM)
	Contributing Partner	MEDITECH
	Comment	Ensures that the platform remains responsive to

Funded by the European Union
Project funded by

		changes in workload demand.
	Classification	Should Have (S)

7.2.4.3 Business Requirements (BUS)

Table 34: DWM.BUS requirements

Req. Id	Requirement Description	
DWM.BUS.001	As a business stakeholder, I want to minimize the operational costs of workload management by optimizing the workload assignment and execution process across the nodes, so that the overall cost of resource us-age is reduced. **Action**: Minimize operational costs **Object**: Workload assignment and execution processes **Constraint/Value**: Operational cost minimization, associated with resource usage and management	
	Affected components	Distributed Workload Manager (DWM) Node Manager – Workload Orchestrator (NWO)
	Contributing Partner	CERTH
	Comment	-
	Classification	Should Have (S)
DWM.BUS.002	As a business stakeholder, I want to improve the performance of the swarm by ensuring that AI workloads are executed as efficiently as possible, with minimum delays and maximum throughput, so as to meet business goals. **Action**: Improve swarm performance **Object**: AI workload performance **Constraint/Value**: Ensure minimum delays and maximum throughput	
	Affected components	Distributed Workload Manager (DWM) Distributed Service Manager (DSM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)

7.2.4.4 Business Technical Requirements (BTC)

Table 35: DWM.BTC requirement

Req. Id	Requirement Description
DWM.BTC.001	As a workload orchestrator, I want the system to support dynamic workload alloc-ation based on changing resource availability in the swarm, so as to ensure work-load management flexibility.

Funded by the European Union
Project funded by

	Action: When resources aren't available, the workload in the K3S nodes must be redistributed automatically to the available K3S nodes, so the processes can finish. **Object:** Data processing in the K3S nodes. **Constraint/Value:** The failure with data processing must be detected quickly in order to redistribute the workload to other nodes as soon as possible. Ideally in less than a second.	
	Affected components	Distributed Workload Manager (DWM) Distributed Resource Manager (DRM)
	Contributing Partner	CERTH
	Comment	This topic is related to scalability and adaptability mechanisms
	Classification	Must have (M)

7.2.5 Distributed Data Manager (DDM)

In this section, the requirements for the Distributed Data Manager (DDM) are presented. This component will provide an abstraction layer, enabling services to utilize data sources defined by the storage services of each Node Context. It will efficiently manage data replication and movement, ensuring transparency, data integrity and optimization to enhance service execution performance.

7.2.5.1 Functional Requirements (FNC)

Table 36: DDM.FNC requirements

Req. Id	Requirement Description	
DDM.FNC.001	As an operator / developer, I want data to be seamless distributed amongst all the Node Context, allowing every node context to know everything it needs to know from all other Node Contexts. **Action:** Integration of node context. **Object:** Logical backbone, Federated multi-context broker. **Constraint:** Real-time processing. **Value:** Synchronization and orchestration of the swarm.	
	Affected components	Distributed Data Manager (DDM) Node Manager - Data Manager (NDM) Distributed Data Manager (DDM) Distributed Workload Manager (DWM)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
DDM.FNC.002	The Distributed Data Manager (DDM) must synchronize output data generated by AI modules over nodes in the Swarm Context. Synchronize and update the results of AI module execution across the Swarm Context (Constraint: Ensure data synchronization latency does not exceed 100ms during normal operation).	

Funded by the European Union
Project funded by

	Action: Synchronize and update results. **Object:** Results of AI module execution across the Swarm Context. **Constraint/Value:** Ensure data synchronization latency does not exceed 100ms during normal operation. **Value**: 100ms latency	
	Affected components	Distributed Data Manager (DDM) Node Manager - Data Manager (NDM)
	Contributing Partner	MEDITECH
	Comment	This requirement ensures accurate, real-time updates to the distributed data while maintaining low latency critical for the platform's operation.
	Classification	Must Have (M)

7.2.5.2 Non-Functional Requirements (NFN)

Table 21: DDM.NFN.001 requirement

Req. Id	Requirement Description	
DDM.NFN.001	The Distributed Data Manager (DDM) must support high availability and fault tolerance for Swarm Context operations. Ensure the system achieves 99.95% uptime. **Action:** Ensure the system achieves 99.95% uptime. **Object:** 99.95% uptime. **Constraint/Value:** Operate under distributed network conditions with node failures of up to 10%	
	Affected components	Distributed Data Manager (DDM) Node Manager - Device Monitoring (NDMo)
	Contributing Partner	MEDITECH
	Comment	High availability ensures uninterrupted distributed synchronization even under challenging conditions, aligning with performance expectations.
	Classification	Must Have (M)

7.2.5.3 Business Requirements (BUS)

Table 37: DDM.BUS requirements

Req. Id	Requirement Description
DDM.BUS.001	As a developer, I want seamless data orchestration across IoT, Edge and Cloud layers, so that immediate decisions can be made for time-sensitive use cases. **Action:** Enable seamless integration.

Funded by the European Union
Project funded by

	Object: Data across IoT, Edge, Cloud layers. **Constraint:** Real-time processing. **Value**: Immediate decisions.	
	Affected components	Distributed Data Manager (DDM) Distributed Workload Manager (DWM) Node Manager - Data Manager (NDM) Distributed Service Manager (DSM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
DDM.BUS.002	As a distributed data manager, I want to enable real-time data management and processing across distributed environments, so that mission-critical applications can operate efficiently. **Action:** Implement real-time data management and processing **Object:** Distributed data environments **Constraint/Value**: Ensure low-latency access and high efficiency in data processing	
	Affected components	Distributed Data Manager (DDM)
	Contributing Partner	HMU, CERTH
	Comment	This requirement ensures seamless data management across distributed environments, supporting real-time operations for mission-critical applications.
	Classification	Must Have (M)
DDM.BUS.003	As a distributed data manager, I want to provide high availability and fault tolerance, so that mission-critical applications remain operational under all conditions. **Action:** Implement redundancy and fault-tolerant mechanisms **Object:** Data availability and system resilience **Constraint:** Ensure continuous operation with minimal downtime	
	Affected components	Distributed Data Manager (DDM) Node Manager - Device Storage (NDS)
	Contributing Partner	HMU, CERTH
	Comment	This requirement guarantees uninterrupted service and system resilience, preventing data loss and ensuring operational continuity.
	Classification	Must Have (M)
DDM.BUS.004	As a distributed data manager, I want to ensure scalable storage and retrieval for large datasets, so that system performance is not degraded as data volume increases. **Action:** Implement scalable storage and efficient retrieval mechanisms **Object:** Large-scale datasets **Constraint**: Maintain performance stability under high data loads	
	Affected components	Distributed Data Manager (DDM) Node Manager - Device Storage (NDS)
	Contributing Partner	HMU, CERTH

Funded by the European Union
Project funded by

	Comment	This requirement ensures that the system can handle increasing data volumes efficiently, preventing bottlenecks and performance degradation.
	Classification	Must Have (M)
DDM.BUS.005	As a distributed data manager, I want to facilitate data sharing and collaboration across departments and geographies, so that teams can work efficiently with synchronized data. **Action:** Enable secure and efficient data sharing **Object:** Distributed and collaborative data access **Constraint**: Ensure data consistency and accessibility across locations	
	Affected components	Distributed Data Manager (DDM) AI Service Model
	Contributing Partner	HMU, CERTH
	Comment	This requirement enhances cross-functional collaboration by ensuring secure and synchronized data availability across multiple locations..
	Classification	Must Have (M)

7.2.5.4 Business Technical Requirements (BTC)

Table 38: DDM.BTC requirement

Req. Id	Requirement Description
DDM.BTC.001	The Distributed Data Manager (DDM), should provide a solution to allow the flow of information in the swarm (data plane) in both directions. If two swarms converge into a common node (as part of a service like FL or split learning), a node should be able to multicast data plane information to nodes of different swarms that belong to the same service instance. **Action:** Enable data plane information between nodes of a swarm and nodes of different swarms that belong to the same service instance. **Object:** Flexibility to send data plane information to nodes of different swarms that belong to the same service instance. **Constraint/Value:** Real-time processing.
	Affected components	Distributed Data Manager (DDM)
	Contributing Partner	VTT
	Comment	If two chains of nodes starting at different end-device nodes, and converging to a common node in the core-cloud tier, are considered as two swarms that belong to a common service instance: the node at the core cloud must receive and send data to two or more nodes (depending on how many swarms converge to the core-cloud node). In this context, the core-cloud node communicates data to nodes in another tier, by multi-casting the data to the containers handling the data processing

Funded by the European Union
Project funded by

		in the corresponding nodes.
	Classification	Should Have (S)

7.3 MIDDLEWARE LAYER – NODE CONTEXT

7.3.1 Node AI Game Agent (NGA)

CoGNETs users will have to face changing conditions of the network, so the communication has to be ensured among different areas of the project. In particular network changes should not affect the communication among the NGA and the DSM.

7.3.1.1 Functional Requirements (FNC)

Table 39: NGA.FNC requirements

Req. Id	Requirement Description	
NGA.FNC.001	As a node AI game agent, I want to collect real-time data about resources, workload and security status from the node, so that I can participate in bidding operations effectively. **Action**: Collect resource-based information from the node **Object**: Real-time node data (e.g., resource usage, workload, security metrics) **Constraint/Value**: Effective participation to swarm bidding operations	
	Affected components	Node AI Game Agent (NGA) Node Manager – Device Monitoring (NDMo)
	Contributing Partner	CERTH, FIWARE
	Comment	This requirement ensures that the AI game agent can make informed decisions in bidding operations by continuously monitoring node conditions. Real-time data collection on resources, workload, and security enhances the agent's adaptability and responsiveness within the swarm continuum.
	Classification	Must Have (M)
NGA.FNC.002	As a node AI game agent, I want to report updates regarding the node's status and decisions to the DSM, so that they can be synchronized with the overall swarm. **Action**: Report updates to the DSM **Object**: Node status and decisions **Constraint/Value**: Effective node synchronization with the swarm	
	Affected components	Node AI Game Agent (NGA) Distributed Service Manager (DSM)
	Contributing Partner	CERTH, FIWARE
	Comment	Should also be taken into account this information also is paramount for the DWM to schedule tasks in the K8S cluster.

Funded by the European Union
Project funded by

	Classification	Must Have (M)
NGA.FNC.003	As a node AI game agent, I want to provide real-time data and context to support game optimization strategies, so that the node may contribute to effective resource allocation in the swarm. **Action**: Provide real-time node data and context to the swarm **Object**: Real-time data and contextual information **Constraint/Value**: Data accuracy and relevance for game optimization decisions	
	Affected components	Node AI Game Agent (NGA) Swarm AI Game Agent (SGA)
	Contributing Partner	CERTH
	Comment	This requirement enables the AI game agent to enhance swarm efficiency by sharing real-time data and contextual insights. By contributing accurate and relevant information, the agent supports dynamic resource allocation and game optimization strategies within the swarm continuum.
	Classification	Must Have (M)
NGA.FNC.004	The Node AI Game Agent (NGA) must gather resource and security information from the node context when demanded. **Action**: Collect resource-based information from the node. **Object**: Real-time node data (e.g., resource usage, workload, security metrics). **Constraint/Value**: Effective participation to swarm bidding operations.	
	Affected components	Node AI Game Agent (NGA) Node Manager – Device Monitoring (NDMo)
	Contributing Partner	CERTH, FIWARE
	Comment	This requirement ensures that the Node AI Game Agent can retrieve critical resource and security data on demand, enabling adaptive decision-making. Access to real-time node context enhances responsiveness and effectiveness in swarm bidding operations.
	Classification	Must Have (M)

7.3.1.2 Non-Functional Requirements (NFN)

Table 40: NGA.NFN requirements

Req. Id	Requirement Description	
NGA.NFN.001	As a node AI game agent, I want to process node data in real time, so that bidding decisions can reflect current conditions accurately. **Action:** Process real-time node data **Object:** Real-time node data **Constraint/Value:** Data must be processed within 100ms of collection	
	Affected components	Node AI Game Agent (NGA)

Funded by the European Union
Project funded by

		Node Manager - Device Monitoring (NDMo)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
NGA.NFN.002	As a node AI game agent, I want to handle increasing amounts of node data as workloads grow, so that system performance is not compromised **Action:** Handle increasing amounts of node data **Object:** Data handling capabilities **Constraint/Value:** Ensure no more than 10% performance degradation when handling high data loads	
	Affected components	Node AI Game Agent (NGA) Distributed Workload Manager (DWM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
NGA.NFN.003	The Node AI Game Agent (NGA) must guarantee consistent communication with the Distributed Service Manager (DSM) under changing network conditions. Implement a fault-tolerant messaging mechanism to maintain synchronization with the Distributed Service Manager (DSM), ensuring a maximum latency of 90ms and packet loss not exceeding 1% for critical tasks. **Action:** Implement a fault-tolerant messaging mechanism **Object:** Maintain synchronization with the Distributed Service Manager (DSM) **Constraint/Value:** maximum latency of 90ms and packet loss not exceeding 1% for critical tasks.	
	Affected components	Node AI Game Agent (NGA) Distributed Service Manager (DSM)
	Contributing Partner	MEDITECH
	Comment	This requirement guarantees reliability in communication and operational efficiency of the Node AI Game Agent in dynamic edge-cloud environments.
	Classification	Must Have (M)

7.3.1.3 Business Requirements (BUS)

Table 41: NGA.BUS requirement

Req. Id	Requirement Description
NGA.BUS.001	As a business stakeholder, I want the NGA to optimize the node's resource usage through game-based strategies, so that operational costs are minimized. **Action:** Optimize node resource usage **Object:** Node resource usage **Value:** Minimum operational costs with high node performance

Funded by the European Union
Project funded by

	Affected components	Node AI Game Agent (NGA)
	Contributing Partner	CERTH
	Comment	This requirement ensures that the Node AI Game Agent applies game-based optimization techniques to enhance resource efficiency. By minimizing operational costs while maintaining high performance, the system supports sustainable and cost-effective node management.
	Classification	Must Have (M)

7.3.1.4 Business Technical Requirements (BTC)

Table 30. NGA.BTC requirement

Req. Id	Requirement Description	
NGA.BTC.001	As a node AI game agent, I want to integrate game theory-based models (e.g., auction-based algorithms), so that resource allocation decisions are optimized. **Action**: Integrate game theory-based models **Object**: Game theory-based algorithms **Constraint/Value**: Optimized and automated resource allocation	
	Affected components	Node AI Game Agent (NGA) Swarm AI Game Agent (SGA)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)

7.3.2 Node Manager - Device Monitoring (NDMo)

The Device Monitoring, associated with the Game Intelligent Agent, is responsible for collecting and analysing requested metrics from the Node Context. It ensures performance monitoring and operational efficiency within the swarm network.

7.3.2.1 Functional Requirements (FNC)

Table 42: NDMo.FNC requirements

Req. Id	Requirement Description
NDMo.FNC.001	As a node operator, I want devices within the CoGNETs ecosystem to self-assess their capabilities using embedded intelligence, so that they can autonomously determine their roles in the continuum and ensure efficient resource allocation. **Action:** Enable self-assessment and role determination **Object:** CoGNETs devices **Constraint:** Embedded

Funded by the European Union
Project funded by

	Value: Efficient resource allocation	
	Affected components	Node Manager - Device Monitoring (NDMo) Node AI Game Agent (NGA)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
	Related Topic	Swarm-wise distributed security paradigms
NDMo.FNC.002	As a node operator I need to know the state and availability of each device at any time, and publish that information to be used by other components. **Action:** Self-awareness of devices and monitoring **Object:** CoGNETs devices **Value:** Efficient resource allocation and monitoring.	
	Affected components	Node Manager - Device Monitoring (NDMo) Node Manager - Device Registration (NDR) Node AI Game Agent (NGA) Distributed Resource Manager (DRM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
	Related Topic	Swarm-wise distributed security paradigms
NDMo.FNC.003	Near Real Time Monitoring: Node Manager - Device Monitoring (NDMo) must monitor node metrics continuously under normal operating conditions **Action**: Collect real-time metrics from each node. **Object**: CPU usage, memory, energy consumption, network performance. **Constraint**: Ensure data accuracy within ±2% of deviation from actual values. **Value**: Real-time metrics from each node.	
	Affected components	Distributed Resource Manager (DRM) Device Monitoring (NDMo) Distributed Data Manager (DDM)
	Contributing Partner	MEDITECH
	Comment	Assure scalable monitoring over nodes
	Classification	Must Have (M)
	Related Topic	Data manageability, IoT-Edge-Cloud swarm continuum architectures, Scalability and adaptability mechanisms

Funded by the European Union
Project funded by

7.3.2.2 Non-Functional Requirements (NFN)

Table 43: NDMo.NFN requirement

Req. Id	Requirement Description	
NDMo.NFN.001	Node Manager - Device Monitoring (NDMo) must operate efficiently under high node density scenarios (e.g., 500 nodes) **Action**: Maintain a response time for monitoring data collection. **Object**: Monitoring data collection. **Constraint**: Support up to 500 nodes simultaneously with no more than 5% performance degradation. **Value**: Response time ≤100ms per node.	
	Affected components	Distributed Workload Manager (DWM) Middleware Layer (Node Context) Distributed Service Manager (DSM) Node Manager - Device Monitoring (NDMo)
	Contributing Partner	MEDITECH
	Comment	Ensure performance optimization for large-scale systems
	Classification	Should Have (S)
	Related Topic	Scalability, and adaptability mechanisms

7.3.2.3 Business Requirements (BUS)

Table 44: NDMo.BUS requirements

Req. Id	Requirement Description	
NDMo.BUS.001	As a system component, the NDMo must support up to 100 network devices simultaneously, ensuring scalability while maintaining efficient operation. Performance degradation should not exceed 5% under peak load conditions. **Action:** Ensure scalability for handling multiple devices. **Object:** Network device management. **Constraint:** Support up to 100 nodes simultaneously with minimal performance impact. **Value:** Performance degradation should not exceed 5% under peak load.	
	Affected components	Distributed Workload Manager (DWM) Distributed Service Manager (DSM) Node Manager - Device Monitoring (NDMo)
	Contributing Partner	ULANCS
	Comment	This requirement ensures that the NDMo system remains scalable and can accommodate large-scale deployments without significant performance loss.
	Classification	Could Have (C)

Funded by the European Union
Project funded by

	Related Topic	Scalability and adaptability mechanisms
NDMo.BUS.002	As a system component, the NDMo must collect and process monitoring data in real-time, ensuring a response time of ≤100ms per device for timely decision-making. **Action:** Maintain a response time for monitoring data collection **Object:** Real-time data monitoring **Constraint:** Response time must not exceed 100ms per node **Value**: Timely insights and decision-making	
	Affected components	Distributed Workload Manager (DWM) Distributed Service Manager (DSM) Node Manager - Device Monitoring (NDMo)
	Contributing Partner	ULANCS
	Comment	This requirement ensures real-time efficiency in data collection and processing, enabling accurate and timely system insights.
	Classification	Could Have (C)
NDMo.BUS.003	As a system component, the NDMo must ensure consistent monitoring accuracy and availability, even under high-density deployment scenarios. **Action:** Maintain system reliability and stability **Object:** Monitoring accuracy and system availability **Constraint:** Must function without failure or data loss under high-density conditions **Value**: Ensure continuous monitoring accuracy	
	Affected components	Distributed Workload Manager (DWM) Distributed Service Manager (DSM) Node Manager - Device Monitoring (NDMo)
	Contributing Partner	ULANCS
	Comment	This requirement ensures that the NDMo remains stable and reliable, even when deployed in high-density network environments.
	Classification	Could Have (C)
NDMo.BUS.004	As a system component, the NDMo must leverage efficient data collection, network optimization, and load balancing techniques to maintain responsiveness and prevent system overload. **Action:** Optimize network resource usage and data handling **Object:** Data collection, network optimization and load balancing **Constraint:** Ensure optimized resource allocation for uninterrupted system performance **Value**: Maintain high responsiveness and prevent bottlenecks	
	Affected components	Distributed Workload Manager (DWM) Distributed Service Manager (DSM) Node Manager - Device Monitoring (NDMo)

Funded by the European Union
Project funded by

	Contributing Partner	ULANCS
	Comment	This requirement ensures that the system can dynamically adjust resource allocation and data handling techniques to maintain efficiency.
	Classification	Could Have (C)

7.3.2.4 Business Technical Requirements (BTC)

Table 45: NDMo.BTC requirement

Req. Id	Requirement Description	
NDMo.BTC.001	As a node operator, I want to integrate utility-based ranking algorithms for energy, computing and security optimization, so that real-time performance updates are provided to the DSM. **Action:** Integrate utility-based ranking **Object:** Utility-based ranking algorithms **Constraint:** Real-time updates must be synchronized with the DSM	
	Affected components	Node Manager - Device Monitoring (NDMo) Distributed Service Manager (DSM) Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	CERTH, ULANCS
	Comment	-
	Classification	Should Have (S)
	Related Topic	Scalability and adaptability mechanisms

7.3.3 Node Manager - Device Registration (NDR)

This section outlines the requirements for the Node Manager - Device Registration (NDR), which is responsible for securely onboarding nodes into the CoGNETS Swarm-Node network. This includes handling the DevOps operations required to initialize and integrate new nodes into the network's architecture.. It is also responsible for the automatic configuration of the distributed network (network plane), ensuring secure and smooth connectivity between nodes.

7.3.3.1 Functional Requirements (FNC)

Table 46: NDR.FNC requirements

Req. Id	Requirement Description
NDR.FNC.001	When initializing the CoGNETs swarm: Register and configure devices to automate secure DevOps procedures.

Funded by the European Union
Project funded by

	Action: Register and configure devices into the swarm. **Object:** Nodes and devices. **Constraint/Value:** Must align with Secure DevOps automation practices.	
	Affected components	Node Manager - Device Registration (NDR) Distributed Service Manager (DSM) Distributed Resource Manager (DRM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
NDR.FNC.002	When a new node is added to the CoGNETs network: Automatically register the node by securely generating a decentralized identity and storing it in the distributed network. **Action**: Automatically register the node. **Object**: Nodes and devices. **Constraint/Value**: Must comply with IoT-Edge-Cloud standards for identity management, and securely generate and store the decentralized identity in the distributed network.	
	Affected components	Distributed Resource Manager (DRM) Node Manager - Data Manager (NDM)
	Contributing Partner	MEDITECH
	Comment	During node initialization, assure reliable identity management.
	Classification	Must Have (M)
NDR.FNC.003	When registering a new node or device: Ensure that only trusted devices are assigned identities and integrated into the swarm. **Action:** Verify device trust and register it securely in the swarm **Object:** Devices and nodes **Constraint/Value:** Trust must be established through cryptographic verification and secure onboarding protocols.	
	Affected components	Distributed Resource Manager (DRM) Node Manager - Data Manager (NDM)
	Contributing Partner	UBITECH
	Comment	-
	Classification	Must Have (M)

Funded by the European Union
Project funded by

7.3.3.2 Non-Functional Requirements (NFN)

Table 47: NDR.NFN requirements

Req. Id	Requirement Description	
NDR.NFN.001	The Node Manager - Device Monitoring (NDMo) must operate efficiently under dynamic conditions. **Action:** Ensure real-time collection and processing of metrics from the Node Context. **Object:** Device monitoring. **Constraint/Value:** Ensure real-time collection and processing of metrics from the Node Context.	
	Affected components	Distributed Resource Manager (DRM) Node Manager - Data Manager (NDM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
NDR.NFN.002	When performing identity registration: Ensure that the registration process completes within a maximum latency of 500ms and maintains a 99.99% success rate for high availability. **Action**: Execute identity registration and verify completion within performance limits. **Object**: Device identity registration process **Constraint/Value**: Maximum latency of 500ms, 99.99% success rate, tested under normal and high network loads.	
	Affected components	Distributed Resource Manager (DRM) Node Manager - Data Manager (NDM)
	Contributing Partner	MEDITECH
	Comment	Ensure performance optimization over various IoT-Edge-Cloud constrains.
	Classification	Should Have (S)

7.3.3.3 Business Requirements (BUS)

Table 48: NDR.BUS requirement

Req. Id	Requirement Description
NDR.BUS.001	When registering a new node or device: Automate node registration and monitoring processes, so that operational costs and deployment time are minimized. **Action**: Automate node registration and monitoring **Object**: Node registration and monitoring processes **Constraint/Value**: Minimize Deployment time and operational costs through automation

Funded by the European Union
Project funded by

	Affected components	Node Manager - Device Registration (NDR) Node Manager - Data Monitoring (NDMo) Distributed Service Manager (DSM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)

7.3.3.4 Business Technical Requirements (BTC)

Table 49: NDR.BTC requirements

Req. Id	Requirement Description	
NDR.BTC.001	As a network device registration component, I want every device to have a unique identification and secure authentication mechanisms (e.g., certificates), so that device security and integrity are ensured. **Action**: Implement unique device identification and authentication mechanisms **Object**: Device authentication and security **Constraint**: Authentication must be secure and tamper-resistant **Value:** Ensures trusted device access and prevents unauthorized use	
	Affected components	Node Manager - Device Registration (NDR)
	Contributing Partner	UAVIGN
	Comment	This requirement ensures that devices are uniquely identifiable and securely authenticated to prevent unauthorized access.
	Classification	Must Have (M)
NDR.BTC.002	As a network device registration component, I want to support automated and manual device registration, track device status (e.g., AUTH/DEAUTH), and handle re-registration, so that device lifecycle management is maintained. **Action**: Implement registration and lifecycle management mechanisms **Object:** Device registration and status tracking **Constraint:** Must support both automated and manual registration processes **Value:** Ensures efficient tracking, updates, and re-registration	
	Affected components	Node Manager - Device Registration (NDR)
	Contributing Partner	UAVIGN
	Comment	This requirement ensures that device registration is efficiently managed throughout its lifecycle, improving system reliability.
	Classification	Must Have (M)
NDR.BTC.003	As a network device registration component, I want to adopt role-based permissions, encryption for data security, and audit logging, so that device and user access is controlled and secure. **Action**: Implement RBAC, encryption and logging	

Funded by the European Union
Project funded by

	Object: Device access permissions and security policies **Constraint:** Ensure data confidentiality, integrity, and traceability **Value:** Improves security compliance and access control	
	Affected components	Node Manager - Device Registration (NDR)
	Contributing Partner	UAVIGN
	Comment	This requirement strengthens security by enforcing access control, encrypting sensitive data, and logging critical actions.
	Classification	Must Have (M)
NDR.BTC.004	As a network device registration component, I want to support high-volume device registrations and fault tolerance mechanisms, so that the system remains scalable and reliable under heavy load. **Action**: Enable scalable registration handling and fault tolerance **Object:** Large-scale device registration and operational stability **Constraint:** Ensure minimal performance degradation under high loads **Value:** Improves system reliability and scalability	
	Affected components	Node Manager - Device Registration (NDR) Distributed Workload Manager (DWM)
	Contributing Partner	UAVIGN
	Comment	This requirement ensures the system can handle a growing number of devices while maintaining performance and fault tolerance.
	Classification	Must Have (M)
NDR.BTC.005	As a network device registration component, I want to ensure compliance with existing regulations, so that the system meets legal and industry standards. **Action**: Implement regulatory compliance mechanisms **Object:** Regulatory adherence and policy enforcement **Constraint:** Must align with industry security and privacy standards **Value:** Ensures legal compliance and system trustworthiness	
	Affected components	Node Manager - Device Registration (NDR)
	Contributing Partner	UAVIGN
	Comment	This requirement ensures the system can handle a growing number of devices while maintaining performance and fault tolerance.
	Classification	Must Have (M)
NDR.BTC.006	As a network device registration component, I want to integrate authentication mechanisms into a web-based/mobile-friendly UI, providing dashboards for administrators and users, so that system interaction is user-friendly. **Action**: Develop web-based/mobile authentication UI with dashboards **Object:** User-friendly interface for authentication and management **Constraint:** Must support multiple device types and access levels **Value:** Enhances user experience and system usability	
	Affected components	Node Manager - Device Registration (NDR)

Funded by the European Union
Project funded by

	Contributing Partner	UAVIGN
	Comment	This requirement ensures that authentication and device management are accessible through an intuitive and responsive UI.
	Classification	Must Have (M)
NDR.BTC.007	As a network device registration component, I want to support deployment in cloud or on-premises environments, with automated updates, backups, and disaster recovery, so that the system remains secure and operational. **Action**: Enable flexible deployment, updates, and disaster recovery **Object:** Cloud/on-premises infrastructure and system resilience **Constraint:** Must support automated updates and failover mechanisms **Value:** Ensures continuous operation and security	
	Affected components	Node Manager - Device Registration (NDR)
	Contributing Partner	UAVIGN
	Comment	This requirement ensures that the system remains functional, secure, and recoverable in case of failures.
	Classification	Must Have (M)

7.3.4 Node Manager - Device Storage (NDS)

This section will outline the initial defined requirements for the Device Storage (NDS), derived from its interactions with its linked components. This component will be responsible for the storage of the data in the Node, related to the DDAG update and synchronization operations with the Swarm Context. It will allow Entity Data to be available for subsequent processes avoiding retraining of CoGNETs AI models and thus reducing time and energy consumption. NDS will incorporate an NGSI-LD API standard for the management of CoGNETs AI Service model JSON Schemas as well as AI module executions, as well as a Non-Relational Database to allow flexibility in the schemas.

7.3.4.1 Functional Requirements (FNC)

Table 50: NDS.FNC requirements

Req. Id	Requirement Description	
NDS.FNC.001	As a device storage component, I want to store input, intermediate and output data for CoGNETs AI module executions, so that data are available for subsequent processes. **Action**: Store data for subsequent module executions. **Object**: Input, intermediate and output data. **Value**: Data availability when requested.	
	Affected components	Node Manager - Device Storage (NDS) Node Manager - Component Executor (NCE)
	Contributing Partner	CERTH

Funded by the European Union
Project funded by

	Comment	Assure to comply with DDAG data structure for smooth updates.
	Classification	Must Have (M)
NDS.FNC.002	Node Manager - Device Storage (NDS) shall handle data storage requests from Node Contexts when a data update or synchronization task is triggered and ensure the storage of data within the node by using a distributed data synchronization mechanism. **Action:** Ensure the storage of data within the node using a distributed data synchronization mechanism. **Object:** Data storage within the node. **Constraint:** Storage should be updated in real-time. **Value:** Synchronization latency ≤ 90ms.	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM) Distributed Service Manager (DSM)
	Contributing Partner	MEDITECH
	Comment	Assure to comply with DDAG data structure for smooth updates.
	Classification	Should Have (S)
NDS.FNC.003	As a device storage component, I want to support the stamping of data with a digital timestamp when the entity is updated, so that the data reflects its accurate state at the time of the update. **Action**: Stamping data with a timestamp **Object**: Entity data **Constraint/Value**: Timestamp accuracy ≤ 1ms	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM)
	Contributing Partner	HMU
	Comment	Ensures accurate version tracking and data synchronization in the system.
	Classification	Must Have (M)
NDS.FNC.004	As a device storage component, I want to allow near real-time updates of the entity data in the Device Storage, so that the node always has the most current data for synchronization. **Action**: Near real-time updates of entity data **Object**: Entity data **Constraint/Value**: Synchronization latency ≤ 90ms	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM)
	Contributing Partner	HMU
	Comment	Ensures seamless near real-time data availability for distributed operations.
	Classification	Must Have (M)
NDS.FNC.005	As a device storage component, I should provide entity data search queries for	

Funded by the European Union
Project funded by

	filtered data retrieval. **Action**: Filter entity data. **Object**: List of Entity data. **Constraint/Value**: Availability ≥ 99.99%	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM)
	Contributing Partner	HMU
	Comment	Allows filtered data retrieval for various operations.
	Classification	Must Have (M)
NDS.FNC.006	As a device storage component, I must support categorization of my entity data to ensure simplicity and heterogeneity, enabling organized access, scalability, and flexibility. **Action**: Categorization of Entity Data. **Object**: Stored data. **Constraint/Value**: Category_Num > 1	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM)
	Contributing Partner	HMU
	Comment	Allows filtered data retrieval for various operations.
	Classification	Must Have (M)

7.3.4.2 Non-Functional Requirements (NFN)

Table 51 NDS.NFN Requirements

Req. Id	Requirement Description	
NDS.NFN.001	As a device storage component, I want to retrieve stored data with a latency of less than 50ms, so that node processes are not delayed. **Action**: Retrieve stored data. **Object**: Stored node data. **Constraint/Value**: Ensure retrieval latency is less than 50ms.	
	Affected components	Node Manager - Device Storage (NDS)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
NDS.NFN.002	The Node Manager - Device Storage (NDS) system must securely store and manage data generated by the Distributed Data Manager (DDM) in compliance with GDPR and relevant data governance policies, ensuring high availability and integrity of the stored data. The NDS must provide encrypted storage mechanisms (AES-256 encryption) and implement redundancy protocols to achieve a data availability of 99.99%.	

Funded by the European Union
Project funded by

	Action: Provide encrypted storage mechanisms and implement redundancy protocols. **Object:** Data storage within the Node Manager - Device Storage (NDS). **Constraint:** Must use AES-256 encryption and implement redundancy protocols. **Value**: Achieve a data availability of 99.99%.	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM)
	Contributing Partner	MEDITECH
	Comment	This requirement focus on secure and efficient data storage to comply with regulatory compliance while providing operational performance for swarm-powered IoT deployments.
	Classification	Must Have (M)
NDS.NFN.003	Node Manager - Device Storage (NDS) should maintain a high level of data integrity under simultaneous node synchronization activities and guarantee data accuracy and reliability in distributed storage operations. **Action:** Guarantee data accuracy and reliability in distributed storage operations **Object:** Distributed storage operations within the Node Manager - Device Storage (NDS) **Constraint:** Error rate must be ≤ 0.01% **Value**: Data verification mechanism included	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM)
	Contributing Partner	MEDITECH
	Comment	Incorporate periodic integrity monitoring to assure data security and consistency maintenance.
	Classification	Should Have (S)
NDS.NFN.004	As a device storage component, I want to ensure that entity data in the storage is synchronized with other nodes in the Swarm Context, so that all nodes have consistent data. **Action**: Synchronize entity data with Swarm Context **Object**: Entity data **Constraint/Value**: Synchronization latency ≤ 100ms	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM) Swarm Context
	Contributing Partner	HMU
	Comment	Essential for ensuring consistency across all nodes in the Swarm.
	Classification	Must Have (M)
NDS.NFN.005	As a device storage component, I want to maintain high availability of stored data during synchronization operations, so that data is accessible without interruption. **Action**: Ensure data availability during synchronization	

Funded by the European Union
Project funded by

	Object: Stored data **Constraint/Value**: Availability ≥ 99.99%	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM)
	Contributing Partner	HMU
	Comment	Data must remain accessible even during distributed synchronization activities to ensure uninterrupted service.
	Classification	Must Have (M)
NDS.NFN.006	As a device storage component, I want to implement regular backups to ensure data integrity and availability. **Action**: Hold regular data backups **Object**: Stored data **Constraint/Value**: Availability ≥ 99.99%, Integrity ≥ 99.99%	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM)
	Contributing Partner	HMU
	Comment	Device Storage must have enough memory for backups
	Classification	Must Have (M)
NDS.NFN.007	As a device storage component, I want to conduct regular health checks to ensure my health state and be constantly informed for any possible issue. **Action**: Regular health checks of systems and services **Object**: Stored data **Constraint/Value**: Health_State ≥ 99.99%	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM)
	Contributing Partner	HMU
	Comment	This can be done with a scheduler (node-schedule)
	Classification	Must Have (M)

7.3.4.3 Business Requirements (BUS)

Table 52: NDS.BUS Requirement

Req. Id	Requirement Description	
NDS.BUS.001	As a device storage component, I want to provide mechanisms for easy retrieval of stored data upon request, so that the data can be used for subsequent AI module executions. **Action**: Retrieve stored data upon request **Object**: Stored data **Constraint/Value**: Retrieval latency ≤ 50ms	
	Affected components	Node Manager - Device Storage (NDS) AI Service Model

Funded by the European Union
Project funded by

	Contributing Partner	HMU
	Comment	Ensures efficient retrieval for AI service processing, improving overall system performance.
	Classification	Must Have (M)

7.3.4.4 Business Technical Requirements (BTC)

Table 53 NDS.BTC Requirement

Req. Id	Requirement Description	
NDS.BTC.001	As a device storage component, I want to utilize secure and robust data transmission, granting authorization to the components to write and extract entity data and other info. **Action**: Ensure authentication and security with other components **Object**: Entity data **Constraint/Value**: Security ≥ 99.99%	
	Affected components	Node Manager - Device Storage (NDS) Distributed Data Manager (DDM)
	Contributing Partner	HMU
	Comment	This requirement ensures that the device storage component maintains a high level of security and integrity when transmitting and managing entity data. By enforcing authentication and secure data exchange, the system guarantees reliable access control and prevents unauthorized modifications, achieving a security threshold of ≥99.99%.
	Classification	Must Have (M)

7.3.5 Node Manager - Data Manager (NDM)

The Data Manager (NDM) is responsible for controlling the sharing of data collected by the cognitive resource and associated sensors with the rest of the swarm by linking with the Distributed Data Manager outputs at the Swarm level. In addition, this component will track the data provenance in the node to ensure data integrity, ownership and policy enforcement. A signature of the CoGNETS AI result will be obtained to maintain trustworthiness in the communication and provide a stamp in the DDAG.

This section will outline the initial defined requirements for the NDM derived from its interactions with its linked components.

Funded by the European Union
Project funded by

7.3.5.1 Functional Requirements (FNC)

Table 54: NDM.FNC requirements

Req. Id	Requirement Description	
NDM.FNC.001	As a node data manager, I want to track the provenance of all data used and generated by the node, so that their origins and transformations can be verified. **Action**: Track data provenance **Object**: Provenance of input, intermediate and output data **Constraint/Value**: Verification of node data origin and transformation	
	Affected components	Node Manager - Data Manager (NDM) Distributed Data Manager (DDM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
	Related Topics	Data manageability
NDM.FNC.002	As a node data manager, I want to compose and append stamp in the DDAG that includes data provenance and signature information, so that the node's contribution is recorded. **Action**: Compose and append DDAG stamp. **Object**: Provenance and signature stamp in the DDAG. **Constraint/Value**: Record with the node's contribution.	
	Affected components	Node Manager - Data Manager (NDM) DDM
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
	Related Topics	IEC swarm continuum architectures
NDM.FNC.003	The Node Manager - Data Manager (NDM) must ensure data provenance and verification in the Edge-Cloud Continuum. Validate and store the provenance of the data generated by CoGNETs AI modules, applying cryptographic signatures to ensure trustworthy communication and update the DDAG. The provenance verification must occur within 50ms latency. **Action:** Validate and store the provenance of the data. **Object:** Data generated by CoGNETs AI modules. **Constraint:** Provenance verification must occur within 50ms latency. **Value:** Application of cryptographic signatures to ensure trustworthy communication and update the DDAG.	
	Affected components	Distributed Data Manager (DDM) Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	MEDITECH
	Comment	Assures trustworthiness and traceability of the data in distributed setups.
	Classification	Must Have (M)

Funded by the European Union
Project funded by

	Related Topics	

7.3.5.2 Non-Functional Requirements (NFN)

Table 55: NDM.NFN requirement

Req. Id	Requirement Description	
NDM.NFN.001	The Node Manager - Data Manager (NDM) must maintain high performance and security standards under load. **Action**: Ensure the secure handling of data. **Object**: Data (handling/processing) within the Node Manager - Data Manager (NDM). **Constraint**: Downtime must be no more than 0.01% over the course of one month. **Value**: Data verification processes must be compatible with scalability mechanisms to handle up to 1000 concurrent data provenance checks per second.	
	Affected components	Node Manager - Device Storage (NDS) Node Manager - Data Manager (NDM)
	Contributing Partner	MEDITECH
	Comment	Focuses on scalability and reliable performance in distributed setups.
	Classification	Should Have (S)

7.3.5.3 Business Requirements (BUS)

Table 56: NDM.BUS requirement

Req. Id	Requirement Description	
NDM.BUS.001	The Node Manager - Data Manager (NDM) must ensure that data adheres to legal and regulatory requirements concerning data management. **Action**: Ensure compliance with data governance policies. **Object**: Data storage, processing and sharing mechanisms. **Constraint:** Adherence to GDPR, HIPAA and other relevant legal frameworks. **Value:** Guarantee lawful data usage, user privacy protection, and regulatory alignment.	
	Affected components	Distributed Data Manager (DDM) Node Manager - Data Manager (NDM)
	Contributing Partner	FIWARE
	Comment	Legal and ethical aspects
	Classification	Must Have (M)

Funded by the European Union
Project funded by

7.3.5.4 Business Technical Requirements (BTC)

Table 57: NDM.BTC requirement

Req. Id	Requirement Description	
NDM.BTC.001	As a node operator, I want to integrate utility-based ranking algorithms for energy, computing and security optimization, so that real-time performance updates are provided to the DSM. **Action**: Integrate utility-based ranking **Object:** Utility-based ranking algorithms **Constraint:** Real-time updates must be synchronized with the DSM	
	Affected components	Distributed Service Manager (DSM) Node Manager - Data Manager (NDM) Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	CERTH
	Comment	-
	Classification	Should Have (S)

7.3.6 Node Manager - Workload Orchestrator (NWO)

This section lists the initially defined requirements related to Workload Orchestrator (WO), one of the key node services. Specifically, WO is responsible to identify the need of a Node Context to execute a CoGNETs AI module. For this purpose, it takes the corresponding docker image of the AI Module along with the associated input data and calls the Component Executor (another Node service) to launch the execution. After that it collects the output data to update the DDAG.

7.3.6.1 Functional Requirements (FNC)

Table 58: NWO.FNC requirements

Req. Id	Requirement Description	
NWO.FNC.001	As a workload orchestrator, I want to identify when a node context needs to execute a CoGNETs AI module, so that execution begins immediately and resources are efficiently allocated. **Action**: Identify needs for execution **Object**: AI module execution triggers **Value**: Immediate detection of execution needs	
	Affected components	Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
NWO.FNC.002	As a workload orchestrator, I want to retrieve the corresponding docker image of the CoGNETs AI module and the required input data, so that the execution environment is set up correctly.	

Funded by the European Union
Project funded by

	Action: Retrieve and manage docker images and data **Object**: Docker images and input data **Constraint**: Ensure compatibility with node-level execution requirements and resource availability	
	Affected components	Node Manager - Workload Orchestrator (NWO) Cognitive AI Service Repository (CSR)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
NWO.FNC.003	As a workload orchestrator, I want to call the NCE to launch the execution of the CoGNETs AI module, so that the output can be processed and stored. **Action**: Trigger NCE **Object**: NCE for module execution **Constraint**: Execution must begin promptly	
	Affected components	Node Manager - Workload Orchestrator (NWO) Node Manager - Component Executor (NCE)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
NWO.FNC.004	The Node Manager - Workload Orchestrator (NWO) shall dynamically assign tasks when a CoGNETs AI module request is received when resource availability and security criteria are validated by the Node Context. **Action**: The NWO shall identify the appropriate AI module from the Cognitive AI Service Repository (CSR), retrieve the corresponding containerized instance (e.g., Docker image), and execute it on the node. **Object**: AI module from the Cognitive AI Service Repository (CSR). **Constraint/Value:** Successful execution of AI modules with an accuracy of ≥95% adherence to input-output validation parameters.	
	Affected components	Node Manager - Workload Orchestrator (NWO) Node Manager - Component Executor (NCE) Application Layer Cognitive AI Service Repository (CSR)
	Contributing Partner	MEDITECH
	Comment	The requirement assures efficient execution and allocation of distributed AI modules while adhering to system constraints like security, resource availability, and performance. The NWO must communicate smoothly with the CSR and local Node Context for enhanced orchestration of tasks.
	Classification	Should Have (S)
NWO.FNC.005	As a Node Orchestrator I want to collect self-awareness data of the node I orchestrate and publish it so other components can use this information to improve their decisions.	

Funded by the European Union
Project funded by

	Action: The NWO must collect information on how the Nodes are performing in order to know exactly what is happening in the different nodes. This information will be later used to deploy other services or in cases, to redeploy currently running services. **Object**: Node performance metrics (e.g., resource usage, latency, execution status). **Constraint/Value**: Ensure that data collection occurs in real-time (latency <=100ms).	
	Affected components	Middleware Layer (Node context). Application layer Physical layer.
	Contributing Partner	FIWARE
	Comment	-
	Classification	Should Have (S)
NWO.FNC.006	As a workload orchestrator, I want to ensure the proper lifecycle of the containers launched by the NCE. Thus, I must handle the termination of the container execution in the case that the container fails to end by itself (that is, ending triggered from within the container). This is to ensure that the container does not remain lying around (and thus using resources) when the AI module ends the execution. **Action**: Handle properly the lifecycle of the container from the NWO. **Object**: Suitability of the NWO to force the termination of the container execution. **Constraint/Value**: None identified.	
	Affected components	Node Manager - Component Executor (NCE)
	Contributing Partner	VTT
	Comment	-
	Classification	Must Have (M)

7.3.6.2 Non-Functional Requirements (NFN)

Table 59: NWO.NFN requirements

Req. Id	Requirement Description	
NWO.NFN.001	As a workload orchestrator, I want to manage the execution of AI models with a latency of less than 100ms, so that the node maintains real-time responsiveness. **Action:** Ensure real-time execution of AI models **Object:** Real-time execution **Constraint:** Ensure latency remains below 100ms	
	Affected components	Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	CERTH
	Comment	Scalability and adaptability mechanisms
	Classification	Must Have (M)
NWO.NFN.002	As a workload orchestrator, I want to ensure that the AI module execution is compatible with the node's available resources, so that the system avoids	

Funded by the European Union
Project funded by

	overloading. **Action:** Ensure compatible AI execution with the node's available resources **Object:** AI execution based on the node's available resources **Value:** Overload prevention	
	Affected components	Node Manager - Workload Orchestrator (NWO) Node Manager - Device Monitoring (NDMo)
	Contributing Partner	CERTH
	Comment	Data manageability
	Classification	Must Have (M)
NWO.NFN.003	The Node Manager - Workload Orchestrator (NWO) must ensure optimal orchestration of AI module workloads under peak workload scenarios with at least 80% resource utilization. The NWO must dynamically allocate workloads to available nodes while maintaining response times below 50ms for module initiation, ensuring that the system adheres to specified throughput levels and avoids bottlenecks. **Action:** Dynamically allocate workloads to available nodes. **Object:** Workloads for AI module execution. **Constraint:** Maintain response times below 50ms for module initiation and ensure adherence to specified throughput levels while avoiding bottlenecks. **Value:** Efficient workload distribution under peak workload scenarios with at least 80% resource utilization.	
	Affected components	Distributed Service Manager (DSM) Node AI Game Agent (NGA) Cognitive AI Service Repository (CSR)
	Contributing Partner	MEDITECH
	Comment	This requirement ensures efficient workload orchestration and alignment with the system's scalability objectives while maintaining performance under high demand.
	Classification	Must Have (M)

7.3.6.3 Business Requirements (BUS)

Table 60: NWO.BUS requirement

Req. Id	Requirement Description	
NWO.BUS.001	As a business stakeholder, I want the NWO to automate the execution of AI modules within the node, so that operational efficiency is improved. **Action:** Automate AI module execution **Object:** AI module execution processes **Value:** Manual intervention reduction, operational costs minimization	
	Affected components	Node Manager - Workload Orchestrator (NWO)

Funded by the European Union
Project funded by

	Contributing Partner	CERTH
	Comment	-
	Classification	Should Have (S)

7.3.6.4 Business Technical Requirements (BTC)

Table 61: NWO.BTC requirement

Req. Id	Requirement Description	
NWO.BTC.001	As a workload orchestrator, I want to recover the output data from the NCE and update the DDAG, so that the results are accessible for synchronization. **Action:** Recover output data and update DDAG. **Object:** Output data and DDAG with execution results. **Value:** Ensure data are synchronized with the DDAG, promptly after retrieval.	
	Affected components	Node Manager - Workload Orchestrator (NWO) Distributed Data Manager (DDM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)

7.3.7 Node Manager - Component Executor (NCE)

The Node Manager - Component Executor (NCE) is a core component responsible for the execution and lifecycle management of AI modules within the node. It works in close coordination with the Node Manager - Workload Orchestrator (NWO) to ensure efficient resource allocation, timely execution, and proper termination of containerized AI workloads. The NCE plays a critical role in maintaining system performance and resource optimization, particularly in dynamic and resource-constrained environments.

7.3.7.1 Functional Requirements (FNC)

Table 62: NCE.FNC requirements

Req. Id	Requirement Description	
NCE.FNC.001	As a component executor, I want to execute containerized CoGNETs AI workloads and update results to the DDAG so that the AI module execution is seamless. **Action:** Execute and update AI workloads **Object:** Containerized AI workloads and DDAG results **Value:** Seamless execution with result synchronization	
	Affected components	Node Manager – Component Executor (NCE) Distributed Device Manager (DDM) Distributed Service Manager (DSM)
	Contributing Partner	CERTH

Funded by the European Union
Project funded by

	Comment	-
	Classification	Must Have (M)
NCE.FNC.002	As a component executor, I want to process the input data provided by the NWO, so that the AI module is executed with the correct parameters and context. **Action:** Process input data from NWO **Object:** Input data **Value:** Correct model execution	
	Affected components	Node Manager – Component Executor (NCE) Node Manager – Workload Orchestrator (NWO)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
NCE.FNC.003	The Node Manager - Component Executor (NCE) must execute CoGNETs AI modules when triggered by the Node Manager - Workload Orchestrator (NWO) and supplied with the required input data. **Action:** The NCE will load the specific AI module container, execute it using the provided input data, and return the results to the Node Manager - Data Manager (NDM) for synchronization with the DDAG. **Object:** AI module execution process. **Constraint:** Must adhere to defined memory and CPU resource constraints (<80% of allocated resources per node). **Value:** Efficient execution of AI modules while ensuring resource optimization and avoiding bottlenecks.	
	Affected components	Node Manager - Component Executor (NCE) Node Manager - Workload Orchestrator (NWO) Node Manager - Data Manager (NDM)
	Contributing Partner	MEDITECH
	Comment	Ensures efficient execution of CoGNETs AI modules with defined constraints to avoid resource bottlenecks.
	Classification	Must Have (M)
NCE.FNC.004	The Node Manager - Component Executor (NCE) must ensure reliable and timely execution of CoGNETs AI modules under varying workloads. **Action:** Achieve execution latency and manage concurrent AI module executions. **Object:** AI module execution process. **Constraint:** Execution latency of ≤100ms for AI module initialization and ≤500ms for AI module execution, with a failure rate of <0.5%. **Value:** Support up to 10 concurrent AI module executions per node without performance degradation.	
	Affected components	Node Manager - Component Executor (NCE) Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	MEDITECH
	Comment	Defines performance benchmarks to ensure system re-

Funded by the European Union
Project funded by

		liability under high workload scenarios.
	Classification	Must Have (M)
NCE.FNC.005	As a component executor, I must handle properly the termination of the container execution from the container, as part of its lifecycle, to ensure that the container does not remain around (and thus using resources) when the AI module ends the execution. **Action**: Handle properly the lifecycle of the container from inside of the container. **Object**: Proper creation of the image to handle the termination of the container. **Constraint/Value**: None identified.	
	Affected components	Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	VTT
	Comment	-
	Classification	Must Have (M)

7.3.7.2 Non-Functional Requirements (NFN)

Table 63: NCE.NFN requirements

Req. Id	Requirement Description	
NCE.NFN.001	As a component executor, I want to ensure that the AI module execution is performed in real time, so that the node maintains operational responsiveness. **Action:** Ensure real-time AI module execution **Object:** AI modules **Constraint:** Ensure execution latency remains below 100ms for each module	
	Affected components	Node Manager - Component Executor (NCE)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
NCE.NFN.002	As a component executor, I want to optimize the use of computational and memory resources, so that node operations remain efficient even under high workloads. **Action: Optimize computational and memory resource usage** **Object:** Computational and memory resources usage **Constraint:** Resource usage must not exceed 80% of the node's capacity during peak workloads	
	Affected components	Node Manager - Component Executor (NCE) Node Manager – Device Monitoring (NDMo)
	Contributing Partner	CERTH
	Comment	-
	Classification	Should Have (S)

Funded by the European Union
Project funded by

NCE.NFN.003	The Node Manager - Component Executor (NCE) must ensure reliable and timely execution of CoGNETs AI modules under varying workloads. The NCE will achieve execution latency of ≤100ms for AI module initialization and ≤500ms for AI module execution, with a failure rate of <0.5%. The NCE must handle up to 10 concurrent AI module executions per node without performance degradation. **Action:** Achieve execution latency and manage concurrent AI module executions. **Object:** AI module execution process. **Constraint:** Execution latency of ≤100ms for AI module initialization and ≤500ms for AI module execution, with a failure rate of <0.5%. **Value:** Support up to 10 concurrent AI module executions per node without performance degradation.	
	Affected components	Node Manager - Component Executor (NCE) Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	MEDITECH
	Comment	Defines performance benchmarks to ensure system reliability under high workload scenarios.
	Classification	Must Have (M)

7.3.7.3 Business Requirements (BUS)

Table 64: NCE.BUS requirement

Req. Id	Requirement Description	
NCE.BUS.001	As a business stakeholder, I want the NCE to automate the execution of AI modules, so that the operational costs are reduced. **Action:** Execution Automation **Object:** Execution of AI modules **Value:** Ensured automation as input processing, execution and output delivery	
	Affected components	Node Manager - Component Executor (NCE)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)

7.3.7.4 Business Technical Requirements (BTC)

Table 65: NCE.BTC requirement

Req. Id	Requirement Description
NCE.BTC.001	As a component executor, I want to manage the execution of multiple AI workloads concurrently, so that the node can efficiently process multiple tasks. **Action:** Manage concurrent workloads execution

Funded by the European Union
Project funded by

	Object: Concurrent execution of multiple AI workloads **Constraint/Value:** Ensure workloads are executed concurrently without exceeding resource thresholds	
	Affected components	Node Manager - Component Executor (NCE) Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	CERTH
	Comment	Cognitive computing and programming models.
	Classification	Could Have (C)

7.4 MIDDLEWARE LAYER – SECURITY

7.4.1 Hardware-level Security (S-HW)

The Hardware-Level Security (S-HW) platform is designed to ensure robust protection against unauthorized access and maintain data integrity and confidentiality across the system. It employs mechanisms like Physically Unclonable Functions (PUFs) and supports RISC-V architecture to secure nodes in the Edge-Cloud environment. By leveraging AI acceleration and separation of software execution, the platform safeguards critical data even in the event of a complete software breach. It integrates security measures at every layer of the **CoGNETs** architecture, ensuring the robustness of underlying components against attacks. This comprehensive approach protects the infrastructure and sensitive data, ensuring secure and reliable operations.

7.4.1.1 Functional Requirements (FNC)

Table 66: S-HW.FNC requirements

Req. Id	Requirement Description	
S-HW.FNC.001	The system must handle hardware-level security mechanisms like Physically Unclonable Functions (PUFs) when initializing a node in the Edge-Cloud environment. **Action**: Provide the generation and storage of unique cryptographic keys. **Object**: Unique cryptographic keys for every device utilizing PUFs. **Constraint**: Guarantee secure identity and authentication of nodes during deployment. **Value**: Latency threshold of 15ms for key generation.	
	Affected components	Node Manager - Device Registration (NDR) Hardware-level Security (S-HW)
	Contributing Partner	MEDITECH
	Comment	Essential for the secure integration of nodes into the network, particularly in adversarial environments.
	Classification	Must Have (M)
S-HW.FNC.002	As a system security engineer, I want to use Physically Un-clonable Functions (PUFs) and hardware mechanisms. **Action:** By leveraging remote attestation to monitor code execution integrity in	

Funded by the European Union
Project funded by

	real-time. **Object**: Every layer of the CoGNETs architecture **Constraint**: Must use RISC-V and AI **Value**: To strengthen system trustworthiness	
	Affected components	Hardware-level Security (S-HW) Software-level Security (S-SL) Application-level Security (S-APL) AI-level Security (S-AI)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
S-HW.FNC.003	As a system security architect, I am interested in the Security of the underlying components and the robustness of the hardware against attacks, specifically the RISC-V architecture, so that critical software and data remain secure. **Object**: Hardware layer **Action**: We will analyse the robustness of the underlying RISC-V hardware against attacks, e.g. glitching. **Constraint**: Must use RISC-V **Value**: To strengthen system trustworthiness	
	Affected components	Hardware-level Security (S-HW) Software-level Security (S-SL) Application-level Security (S-APL) AI-level Security (S-AI)
	Contributing Partner	BEYOND
	Comment	-
	Classification	Should Have (S)
S-HW.FNC.004	As a system user, I want that my identity is protected and only executed with authorized code and never share my identity to the network. **Object**: Protect (confidentiality primary, other derives from it due to operation of PUF) of identity trust anchors and other long term secrets. **Action**: Long term secrets protected by PUF derived KEK or are directly derived from PUF. **Constraint**: Requires usage of physical PUF device **Value**: High level of assurance that identity is no tampered with. Execution of authorized code (only).	
	Affected components	Hardware-level Security (S-HW) Software-level Security (S-SL) Application-level Security (S-APL) AI-level Security (S-AI)
	Contributing Partner	BEYOND
	Comment	-
	Classification	Should Have (S)
S-HW.FNC.005	As a system user, I want that the system that store my identity can be resilience against the execution of unauthorized code.	

Funded by the European Union
Project funded by

	Object: Execution of authorized code (only). **Action**: Provide RISC-V based secure execution environment that increases resilience against execution of unauthorized code (e.g. malicious code, code not authorized by appropriate – defined by system policy – stake holders, etc.) leveraging control flow integrity technologies (e.g. attestation, etc. **Constraint**: Must use RISC-V processor hardware security primitives **Value**: Limit consequence of breaches and increase assurance that code is correctly executed.	
	Affected components	Hardware-level Security (S-HW) Software-level Security (S-SL) Application-level Security (S-APL) AI-level Security (S-AI)
	Contributing Partner	BEYOND
	Comment	-
	Classification	Should Have (S)

7.4.1.2 Non-Functional Requirements (NFN)

Table 67: S-HW.NFN requirement

Req. Id	Requirement Description	
S-HW.NFN.001	Hardware-level security should ensure protection against unauthorized access, even in the event of a complete software breach. **Action**: Guarantee hardware-enforced isolation of secrets and cryptographic operations. **Object**: Secrets and cryptographic operations. **Constraint/Value**: Protection against unauthorized access	
	Affected components	Hardware-level Security (S-HW) Distributed Resource Manager (DRM)
	Contributing Partner	MEDITECH
	Comment	Essential for providing trustworthiness in resource-constrained sittings and offering secure multi-tenant processes.
	Classification	Should Have (S)

7.4.1.3 Business Requirements (BUS)

Table 68: S-HW.BUS requirement

Req. Id	Requirement Description
S-HW.BUS.001	As a system security developer, I want data integrity and confidentiality to be protected at every layer of the CoGNETs architecture using RISC-V and AI acceleration, so that critical data remain secure across the system.

Funded by the European Union
Project funded by

	Action: Protect data integrity and confidentiality **Object**: Every layer of the CoGNETs architecture **Constraint**: Must use RISC-V and AI **Value**: Enhanced system-wide security	
	Affected components	Hardware-level Security (S-HW) Software-level Security (S-SL) Application-level Security (S-APL) AI-level Security (S-AI)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)

7.4.1.4 Business Technical Requirements (BTC)

Table 69: S-HW.BTC requirement

Req. Id	Requirement Description
S-HW.BTC.001	As a system architect, I want the hardware-level security mechanisms to support RISC-V and separation of software execution. **Action**: To prevent the exfiltration of identity secrets, even in the case of a full breach of all software layers of the Distributed Identity (DID) system. **Object**: Every layer of the CoGNETs architecture. **Constraint**: Must use RISC-V and AI. **Value**: So that the system ensures compliance with high-security standards while preventing physical and software-based attacks.
	Affected components: Hardware-level Security (S-HW) Software-level Security (S-SL) Application-level Security (S-APL) AI-level Security (S-AI)
	Contributing Partner: CERTH
	Comment: -
	Classification: Must Have (M)
S-HW.BTC.002	As a system security architect, I am interested in the Security of the underlying components and the robustness of the hardware against attacks, specifically the RISC-V architecture, so that critical software and data remain secure. **Action**: Analyse the robustness of the underlying RISC-V hardware against attacks, e.g. glitching. **Object**: Hardware layer **Constraint**: Must use RISC-V **Value**: To strengthen system trustworthiness
	Affected components: Hardware-level Security (S-HW) Software-level Security (S-SL) Application-level Security (S-APL) AI-level Security (S-AI)

Funded by the European Union
Project funded by

	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)

7.4.2 Software-level Security (S-SL)

Software-level security will be underpinned by low-overhead DID mechanisms as extensions of the DID Management for node identity and authentication, translated to the IoT-to-Cloud environment.

7.4.2.1 Functional Requirements (FNC)

Table 70: S-SL.FNC requirements

Req. Id	Requirement Description	
S-SL.FNC.001	The security system shall integrate low-overhead Decentralized Identifier (DID) mechanisms to support node identity and authentication in the IoT-to-Cloud environment. **Action**: Implement low-overhead Decentralized Identifier (DID) mechanisms. **Object**: Node identity and authentication. **Value**: Support node identity and authentication in the IoT-to-Cloud environment.	
	Affected components	Software-level Security (S-SL)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
	Related topic	Swarm-wise distributed security paradigms
S-SL.FNC.002	The security system shall allow Swarm Nodes to authenticate themselves using Self-Issued ID Tokens signed with keys under the Swarm Node's control. **Action**: Adopt Self-Issued OpenID Provider v2 (SIOPv2). **Object**: Node identity and authentication. **Value**: Make the swarm node become the issuer of identity information.	
	Affected components	Software-level Security (S-SL)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
	Related topic	Swarm-wise distributed security paradigms

Funded by the European Union
Project funded by

7.4.2.2 Non-Functional Requirements (NFN)

Table 71: S-SL.NFN requirements

Req. Id	Requirement Description	
S-SL.NFN.001	The security system shall ensure that the authorization code flow for presenting Verifiable Credentials (VCs) is secure and efficient, adhering to the OpenID Connect for Verifiable Presentations (OIDC4VP) standards. **Action**: Adopt OpenID Connect for Verifiable Presentations (OIDC4VP) standards. **Object**: Verifiable credentials presentation. **Value**: Secure and efficient authorization code flow for presenting Verifiable Credentials.	
	Affected components	Software-level Security (S-SL)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
	Related topic	Swarm-wise distributed security paradigms
S-SL.NFN.002	The security system shall comply with the future eIDAS2 regulation and the EU Digital ID (EUDI) Wallet Architecture & Reference Framework (ARF) to ensure security and interoperability. **Action**: Ensure compliance with eIDAS2 regulation and the EU Digital ID (EUDI) Wallet Architecture & Reference Framework (ARF). **Object**: Authentication mechanism **Constraint/Value**: Secure and interoperable authentication of nodes.	
	Affected components	Software-level Security (S-SL)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
	Related topic	Swarm-wise distributed security paradigms

7.4.2.3 Business Requirements (BUS)

Table 72: S-SL.BUS requirements

Req. Id	Requirement Description
S-SL.BUS.001	The security system must align with the goals of the i4Trust project to enhance trust and security in decentralized identity management for IoT applications. **Action**: Translation of the i4Trust DID Management proposal to the IoT-to-Cloud. **Object**: Node identity and authentication. **Constraint/Value**: Ensure alignment with GAIA-X and the European Blockchain

Funded by the European Union
Project funded by

	Services Infrastructure (EBSI).	
	Affected components	Software-level Security (S-SL)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
	Related topic	Swarm-wise distributed security paradigms
S-SL.BUS.002	The security system should facilitate the integration of EU Qualified Trust Service Providers (TSPs) to ensure compliance with European regulations and standards for digital identity. **Action**: Facilitate the integration of EU Qualified Trust Service Providers. **Object**: Node identity and authentication. **Constraint/Value**: Ensure compliance with European regulations and standards for digital identity.	
	Affected components	Software-level Security (S-SL)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
	Related topic	Swarm-wise distributed security paradigms

7.4.2.4 Business Technical Requirements (BTC)

Table 73: S-SL.BTC requirements

Req. Id	Requirement Description	
S-SL.BTC.001	The security system shall utilize W3C Verifiable Credentials (VCs) and Verifiable Presentations (VPs) to package and present identity information securely. **Action**: Adopt W3C Verifiable Credentials and Verifiable Presentations. **Object**: Node identity and authentication. **Constraint/Value**: Allow the package of VCs so that the authorship of the data is verifiable.	
	Affected components	Software-level Security (S-SL)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
	Related topic	Swarm-wise distributed security paradigms
S-SL.BTC.002	The security system must leverage Concise Binary Object Representation (CBOR) and ISO18013 for VCs to optimize performance in environments with low communication speed. **Action**: Leverage Concise Binary Object Representation (CBOR) and ISO18013 for	

Funded by the European Union
Project funded by

	VCs. **Object**: Authentication mechanism. **Constraint:** Comply with the EUDI Wallet ARF guidelines. **Value**: Optimize performance in low communication speed environments, such as in IoT-to-Cloud.	
	Affected components	Software-level Security (S-SL)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
	Related topic	Swarm-wise distributed security paradigms

7.4.3 Application-level Security (S-APL)

Application-level Security requirements are focusing on the needs to provide protection of the deployed applications and the underlying swarm infrastructure. The requirements covered by the Application-level Security (S-APL) component (presented in section 4) cover security policies management, monitoring and identity management.

7.4.3.1 Functional Requirements (FNC)

Table 74: S-APL.FNC requirements

Req. Id	Requirement Description	
S-APL.FNC.001	As a security manager, I want the application layer to be dependable and trustworthy, i.e., well security tested. **Action**: Analyse and evaluate the security state of the system and its components in all operational phases (using threat modelling methodologies such as attack trees, STRIDE, and MITRE, and identifying vulnerabilities at the interface between Physical Unclonable Functions (PUFs) and the DID layer). **Object**: Application Layer and communication **Value**: Improved system security - avoiding vulnerabilities	
	Affected components	Middleware Layer (Security Management) Application-level Security (S-APL)
	Contributing Partner	MEDITECH, SBA
	Comment	This requirement ensures that the system proactively addresses potential security vulnerabilities in the Application Layer, focusing on identity layer and risk evaluation.
	Classification	Must Have (M)
S-APL.FNC.002	As a security administrator, I want the system to monitor API calls between components in real time, so that unauthorized access attempts can be detected and prevented. **Action**: Monitor API calls	

Funded by the European Union
Project funded by

	Object: Communication between components **Constraint**: Real-time detection of unauthorized calls within 5 milliseconds	
	Affected components	Middleware Layer (Security Management) Application-level Security (S-APL)
	Contributing Partner	FIWARE, UBI
	Comment	This requirement ensures that the system proactively addresses potential security vulnerabilities in the Application Layer, focusing on identity layer and risk evaluation.
	Classification	Could Have (C)
S-APL.FNC.003	As an application-layer security feature, I want to enforce role-based access control (RBAC) for all system interactions, so that only authorized users can perform sensitive actions **Action:** Enforce role-based access control **Object:** System interactions **Constraint:** Permissions updated within 1 second of changes	
	Affected components	Middleware Layer (Security Management) Application-level Security (S-APL)
	Contributing Partner	FIWARE, UBI
	Comment	This requirement ensures that the system proactively addresses potential security vulnerabilities in the Application Layer, focusing on identity layer and risk evaluation.
	Classification	Could Have (C)

7.4.3.2 Non-Functional Requirements (NFN)

Table 75: S-APL.NFN requirements

Req. Id	Requirement Description	
S-APL.NFN.001	As a security manager, I want the application layer to be dependable and trustworthy, i.e. well security tested when monitoring the identity layer of swarm nodes. **Action**: Analyse and evaluate the security state of the system and its components in all operational phases (using threat modelling methodologies such as attack trees, STRIDE, and MITRE, and identifying vulnerabilities at the interface between PUFs and the DID layer). **Object**: Application Layer and communication. **Constraint/Value**: Improved system security - avoiding vulnerabilities.	
	Affected components	Application Layer Security Manager (S-APL) Physical Unclonable Functions (PUFs) Swarm Nodes Identity Layer Distributed Identity (DID) Layer

Funded by the European Union
Project funded by

		Threat Modelling Mechanisms (e.g., STRIDE, MITRE, Attack Trees)
	Contributing Partner	MEDITECH, SBA
	Comment	This requirement ensures that the system proactively addresses potential security vulnerabilities in the Application Layer, focusing on identity layer and risk evaluation.
	Classification	Must Have (M)
S-APL.NFN.002	The Middleware Layer (Security Management) must analyse system and component-level security across all project phases, with a specific focus on monitoring and anomaly detection at the Application layer. **Action**: Ensure a maximum system response time for anomaly detection and alert generation. **Object**: Anomaly detection and alert generation. **Constraint**: Must be achieved under peak load conditions (simultaneous processing of data from up to 100 nodes). **Value**: 50ms maximum system response time.	
	Affected components	Application-level Security (S-APL)
	Contributing Partner	MEDITECH
	Comment	This requirement is essential to maintaining system integrity and tackling potential threats instantaneously during operation.
	Classification	Must Have (M)
S-APL.NFN.003	The Middleware Layer (Security Management) must monitor and analyse security states at the application level under the condition that distributed AI modules are executing in the Edge-Cloud Continuum. **Action**: Assure complete threat modelling and anomaly detection mechanisms using approaches like STRIDE and MITRE. **Object**: Threat modelling and anomaly detection mechanisms. **Constraint**: Must achieve detection accuracy ≥ 95% for vulnerabilities related to the Physical Unclonable Functions (PUFs) interface and the DID layer. **Value**: Maximum latency of 200ms for response operations.	
	Affected components	Application-level Security (S-APL)
	Contributing Partner	MEDITECH
	Comment	This requirement guarantees reliable monitoring and anomaly detection at the application layer, providing security without affecting on latency or accuracy during distributed AI tasks.
	Classification	Must Have (M)

Funded by the European Union
Project funded by

7.4.3.3 Business Requirements (BUS)

Table 76: S-APL.BUS requirement

Req. Id	Requirement Description	
S-APL.BUS.001	As a business stakeholder, I want the application-layer security to provide automated compliance reporting, so that the system adheres to GDPR and other relevant regulations. **Action:** Automate compliance reporting. **Object:** GDPR and regulatory standards. **Constraint/Value:** Reports generated monthly on-demand.	
	Affected components	Hardware-level Security (S-HW) Software-level Security (S-SL) Application-level Security (S-APL) AI-level Security (S-AI)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)

7.4.3.4 Business Technical Requirements (BTC)

Table 77: S-APL.BTC requirement

Req. Id	Requirement Description	
S-APL.BTC.001	The security system should integrate with external identity providers like OAuth 2.0 and OpenID Connect for seamless authentication **Action:** Integrate with identity providers. **Object:** OAuth 2.0 and OpenID Connect. **Constraint:** Ensure compatibility with existing standards.	
	Affected components	Hardware-level Security (S-HW) Software-level Security (S-SL) Application-level Security (S-APL) AI-level Security (S-AI)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Must Have (M)
S-APL.BTC.002	The application-layer security must implement anomaly detection using machine learning to identify unusual patterns in data access **Action:** Implement anomaly detection **Object:** Data access patterns **Constraint:** Anomalies identified within 2 seconds of detection	
	Affected components	Hardware-level Security (S-HW)

Funded by the European Union
Project funded by

		Software-level Security (S-SL) Application-level Security (S-APL) AI-level Security (S-AI)
	Contributing Partner	FIWARE
	Comment	-
	Classification	Could Have (C)

7.4.4 AI-level Security (S-AI)

AI-level security refers to the advanced measures and protocols that leverage artificial intelligence to protect systems, networks, and data from cyber threats. By utilizing machine learning algorithms, AI can analyse vast amounts of data in real time, identifying patterns and anomalies that may indicate a security breach or malicious activity. This proactive approach enhances threat detection and response times, allowing organizations to mitigate risks more effectively than traditional security measures. Additionally, AI can adapt to evolving threats, continuously improving its defences based on new information and trends, making it a vital component in modern cybersecurity strategies.

7.4.4.1 Functional Requirements (FNC)

Table 78: S-AI.FNC requirements

Req. Id	Requirement Description	
S-AI.FNC.001	As an adversarial shield component, I want to continuously detect adversarial threats in the FL system, in real time, so that the learning processes remain secure and unaffected. **Action**: Detect adversarial threats continuously and in real time **Object**: Adversarial threats in data and model updates **Constraint/Value**: Accurate and immediate threat detection	
	Affected components	AI-level Security (S-AI) Node AI Game Agent (NGA)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
S-AI.FNC.002	As an adversarial shield component, I want to validate individual node contributions during federated learning, so that adversarial inputs do not affect the global model. **Action**: Validate and filter individual node contributions **Object**: Node contributions to the learning process **Constraint/Value**: Only validated contributions are integrated into the global model	
	Affected components	AI-level Security (S-AI) Distributed Service Manager (DSM)

Funded by the European Union
Project funded by

	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)
S-AI.FNC.003	As an adversarial shield component, I want to adapt mitigation strategies to specific learning settings, so that threats are effectively addressed for each use case scenario. **Action:** Adapt to specific learning settings **Object:** Mitigation strategies **Constraint/Value:** Tailored mitigation strategies for each CoGNETs use case scenario	
	Affected components	AI-level Security (S-AI) Distributed Data Manager (DDM)
	Contributing Partner	CERTH
	Comment	-
	Classification	Could Have (C)
S-AI.FNC.004	The split learning mechanism must be robust against attacks from malicious nodes. **Action:** Protect security and privacy of the learning mechanisms from malicious nodes. **Object:** Split learning function and environment. **Constraint/Value**: Protection of the system, nodes and data.	
	Affected components	AI-level Security (S-AI)
	Contributing Partner	SBA
	Comment	-
	Classification	Should Have (S)
S-AI.FNC.005	The split learning mechanism must be robust against external attacks. **Action**: Protect security and privacy of the learning mechanisms from malicious nodes **Object**: Split learning function and environment **Constraint/Value**: Protection of the system, nodes and data	
	Affected components	AI-level Security (S-AI)
	Contributing Partner	SBA
	Comment	-
	Classification	Should Have (S)

Funded by the European Union
Project funded by

7.4.4.2 Non-Functional Requirements (NFN)

Table 79: S-AI.NFN requirement

Req. Id	Requirement Description	
S-AI.NFN.001	As an adversarial shield component, I want the threat detection mechanism to have a latency of less than 20 seconds, so that FL processes remain responsive. **Action:** Establish a threat detection mechanism to ensure responsiveness in the processes. **Object:** Processes and threat detection mechanism. **Constraint/Value:** The threat detection mechanism latency bellow 20s.	
	Affected components	AI-level Security (S-AI)
	Contributing Partner	CERTH
	Comment	Related to swarm-wise distributed security paradigms.
	Classification	Must have (M)
S-AI.NFN.002	As an adversarial shield component, I want to ensure model integrity under adversarial attacks, so that accuracy degradation remains below 10%. **Action:** Stablish a threat detection mechanism to protect and ensure model integrity under attacks. **Object:** Model integrity and threat detection mechanisms. **Constraint/Value:** Accuracy degradation must remain below 10%.	
	Affected components	AI-level Security (S-AI)
	Contributing Partner	CERTH
	Comment	Related to federated learning mechanisms.
	Classification	Should have (S)

7.4.4.3 Business Requirements (BUS)

Table 80: S-AI.BUS requirements

Req. Id	Requirement Description	
S-AI.BUS.001	As a stakeholder, I want to ensure that adversarial threats are mitigated in AI swarm learning, so that operational risks are mitigated. **Action**: Mitigate operational risks. **Object**: Adversarial threats and risks in swarm AI learning processes. **Constraint/Value**: Efficient risks minimization.	
	Affected components	AI-Level Security (S-AI)
	Contributing Partner	CERTH
	Comment	Related to Swarm-wise distributed security paradigms.
	Classification	Should Have (S)
S-AI.BUS.002	As a security administrator, I want the system to continuously monitor AI model	

Funded by the European Union
Project funded by

	inputs and outputs for potential attacks, so that I can ensure the integrity of the model and accurate predictions. **Action**: Detect malicious attempts to manipulate the AI model during both training and inference stages **Object**: Model inputs and outputs **Value**: Efficient malicious attempt detection	
	Affected components	AI-level Security (S-AI) Distributed Data Manager (DDM) Distributed Service Manager (DSM)
	Contributing Partner	CERTH
	Comment	Related to Swarm-wise distributed security paradigms
	Classification	Should Have (S)
S-AI.BUS.003	As a security developer, I want the detection system to be seamlessly integrated with other security layers, so that I can provide a unified and robust defence system throughout the entire CoGNETs ecosystem. **Action**: Integrate seamlessly with all security components (level-wise) to provide a unified defence system **Object**: Security layers (hardware level, system level, application level) **Value**: Efficient threat detection and mitigation	
	Affected components	AI-level Security (S-AI) Hardware-level Security (S-HW) Software-level Security (S-SL) Application-level Security (S-APL) Distributed Resource Manager (DRM)
	Contributing Partner	CERTH
	Comment	Related to Swarm-wise distributed security paradigms
	Classification	Should Have (S)

7.4.4.4 Business Technical Requirements (BTC)

Table 81: S-AI.BTC requirement

Req. Id	Requirement Description	
S-AI.BTC.001	As an adversarial shield component, I want to support risk analysis frameworks, so that threat identification remains consistent across swarm environments. **Action**: Support risk analysis frameworks **Object**: Risk analysis frameworks **Constraint/Value**: Consistent threat identification across swarm environments	
	Affected components	AI-Level Security (S-AI)
	Contributing Partner	CERTH
	Comment	-

Funded by the European Union
Project funded by

	Classification	Must Have (M)

7.5 PHYSICAL LAYER

7.5.1 Operating System (OS)

This section lists the initially defined requirements related to Operating System (OS) of the physical layer i.e. edge devices within the CoGNETs swarm. As CoGNETs actions are to be executed on the swarm, the OS needs to be able to support CoGNETs (e.g. being able to support containerized applications) and at the same time take into account several constraints as, for example, the CPU availability, latency, task scheduling, or resource allocation.

7.5.1.1 Functional Requirements (FNC)

Table 82: OS.FNC requirement

Req. Id	Requirement Description	
OS.FNC.001	The Operating System (OS) in edge devices within the Physical Layer of the CoGNETs platform must be compatible with the Edge-Cloud Continuum architecture and support containerized environments for the execution of CoGNETs AI modules. It should enhance lightweight virtualization technologies like Docker or Kubernetes runtime, assuring compliance with the hardware limitations of edge nodes and supporting dynamic deployment of DNN and RNN components. **Action:** Support containerized environments and enhance lightweight virtualization technologies. **Object:** Execution of CoGNETs AI modules. **Constraint:** Compliance with hardware limitations of edge nodes. **Value:** Dynamic deployment of DNN and RNN components.	
	Affected components	Node Manager - Device Monitoring (NDMo) Node Manager - Component Executor (NCE) Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	MEDITECH
	Comment	Assure that the chosen OS supports real-time processing and complies to security best practices for execution in a distributed system.
	Classification	Must Have (M)

7.5.1.2 Non-Functional Requirements (NFN)

Table 83: OS.NFN requirement

Req. Id	Requirement Description
OS.NFN.001	The Operating System (OS) of all physical devices within the Edge-Cloud continuum under conditions of constrained hardware resources and distributed environment must guarantee resource allocation and task scheduling effectively to support low-latency execution of CoGNETs AI modules, not exceeding a maximum processing

Funded by the European Union
Project funded by

	latency of 40ms per task and maintaining a CPU usage below 80% during peak operation. **Action:** Guarantee resource allocation and task scheduling effectively **Object:** Low-latency execution of CoGNETs AI modules **Constraint:** Maximum processing latency of 40ms per task and CPU usage below 80% during peak operation **Value**: Optimized execution performance under constrained hardware conditions
Affected components	Hardware-level Security (S-HW) Node Manager - Workload Orchestrator (NWO) Distributed Workload Manager (DWM)
Contributing Partner	MEDITECH
Comment	This requirement guarantees the OS optimizes performance under distributed operations and reduces latency while maintaining stable hardware and system-level.
Classification	Must Have (M)

7.5.1.3 Business Requirements (BUS)

Table 84: OS.BUS requirement

Req. Id	Requirement Description
OS.BUS.001	As a business stakeholder, I want the Operating System (OS) to optimize CPU, memory, and storage to handle constrained hardware conditions of each Edge device, so high latency issues in Edge computing environments to be mitigated. **Action:** Optimize CPU, memory, and storage of each Edge device **Object:** Mitigate high latency issues in edge computing environments **Constraint:** OS should be compatible with Devices that run different hardware architectures **Value:** Handle constrained hardware conditions
Affected components	Node Manager - Device Monitoring (NDMo) Node Manager - Device Storage (NDS) Distributed Resource Manager (DRM)
Contributing Partner	SIEMENS
Comment	-
Classification	Must Have (M)

7.5.1.4 Business Technical Requirements (BTC)

Table 85: OS.BTC requirement

Req. Id	Requirement Description
OS.BTC.001	As the Operating System (OS), I want to achieve intelligent task scheduling to maximize performance and minimize latency.

Funded by the European Union
Project funded by

	Action: Intelligent task scheduling. **Object:** Maximize performance and minimize latency. **Constraint:** OS should prioritize low-latency, lightweight scheduling techniques due to Edge devices low hardware constraints.
Affected components	Distributed Resource Manager (DRM) Distributed Service Manager (DSM) Distributed Workload Manager (DWM)
Contributing Partner	AXON
Comment	-
Classification	Should Have (S)

7.5.2 Hardware Platform (HW)

The Hardware Platform (HW), as part of the Physical Layer, will facilitate the deployment of CoGNETs Middleware.

7.5.2.1 Functional Requirements (FNC)

Table 86: HW.FNC requirement

Req. Id	Requirement Description	
HW.FNC.001	The hardware platform (HW) in the Physical Layer should support high-performance execution environments under distributed IoT-Edge-Cloud Continuum operations and assure execution of CoGNETs AI modules with a processing latency of less than 15ms when processing data streams from at least 15 nodes concurrently. **Action:** Assure that the hardware supports local caching of important data and modular redundancy in processing. **Object:** Operational continuity. **Constraint/Value:** Minimal performance degradation. Less than 4% latency increase and less than 2% task completion loss.	
	Affected components	Hardware Platform (HW) Distributed Resource Manager (DRM) Node Manager - Component Executor (NCE) Node Manager - Workload Orchestrator (NWO)
	Contributing Partner	MEDITECH
	Comment	The hardware requirements must align with the computational needs of the DNN and RNN modules while ensuring low power usage and effective cooling methods.
	Classification	Must Have (M)

Funded by the European Union
Project funded by

7.5.2.2 Non-Functional Requirements (NFN)

Table 87: HW.NFN requirement

Req. Id	Requirement Description	
HW.NFN.001	The Hardware Platform (HW) must support the execution of CoGNETs AI modules with conditions of Edge-Cloud Continuum connectivity may encounter intermittent disconnectivity. **Action:** Assure that the hardware supports local caching of important data and modular redundancy in processing to maintain operational continuity with minimal performance degradation (<4% latency increase and <2% task completion loss). **Object:** Operational continuity. **Constraint:** Minimal performance degradation. **Value:** Less than 4% latency increase and less than 2% task completion loss.	
	Affected components	Node Manager - Device Monitoring (NDMo) Node Manager - Device Storage (NDS) Distributed Resource Manager (DRM) Hardware-level Security (S-HW)
	Contributing Partner	MEDITECH
	Comment	This requirement is important to assure the reliability and resilience of the platform in real-world deployment use cases in unpredictable network states.
	Classification	Must Have (M)

7.5.2.3 Business Requirements (BUS)

Table 88: HW.BUS requirement

Req. Id	Requirement Description	
HW.BUS.001	As a business stakeholder, I want the hardware platform to support cost-effective scalability, so that future expansion does not require complete infrastructure overhauls. **Action:** Ensure modular and extensible hardware design **Object:** Hardware platform **Constraint:** Increase in processing capacity through modular upgrades **Value:** Reduced infrastructure replacement costs	
	Affected components	Hardware Platform (HW)
	Contributing Partner	CERTH
	Comment	-
	Classification	Should Have (S)

Funded by the European Union
Project funded by

7.5.2.4 Business Technical Requirements (BTC)

Table 89: HW.BTC requirement

Req. Id	Requirement Description	
HW.BTC.001	As a system architect, I want the hardware platform to support real-time monitoring of performance metrics, so that operational efficiency can be proactively optimized. **Action:** Implement a real-time performance monitoring system **Object:** Diagnostics **Constraint:** Provide alerts of detecting performance degradation **Value:** Reduction in system downtime	
	Affected components	Hardware Platform (HW)
	Contributing Partner	CERTH
	Comment	-
	Classification	Could Have (C)

7.5.3 Connectivity Interfaces (CT)

Connectivity interfaces ensure flexible communication across the IoT-Edge-Cloud continuum by supporting various connectivity interfaces from Ethernet to wireless communications, allowing all sub-components of the CoGNETs architecture to communicate seamlessly across the whole computing continuum.

7.5.3.1 Functional Requirements (FNC)

Table 90: CT.FNC requirement

Req. Id	Requirement Description	
CT.FNC.001	The system should support multiple connectivity interfaces (e.g., Wi-Fi, 5G, and LP-WAN) to ensure seamless integration into diverse IoT infrastructures. Enable dynamic selection and prioritization of connectivity interfaces based on device capability, network availability, and application requirements. This functionality should minimize latency to less than 20ms for time-critical applications. **Action:** Enable dynamic selection and prioritization of connectivity interfaces **Object:** Connectivity interfaces based on device capability, network availability, and application requirements **Constraint:** Functionality should minimize latency **Value:** Latency should be less than 20ms for time-critical applications	
	Affected components	Physical Layer - Connectivity Interfaces (CT)
	Contributing Partner	MEDITECH, SBA
	Comment	Critical for ensuring reliable communication between nodes and the Edge-Cloud Continuum.
	Classification	Must Have (M)

Funded by the European Union
Project funded by

7.5.3.2 Non-Functional Requirements (NFN)

Table 91: CT.NFN requirements

Req. Id	Requirement Description	
CT.NFN.001	Connectivity Interfaces (CT) for Edge-Cloud Continuum requirement indicates the kinds of connectivity exist in the physical layer to support the flexible communication between nodes in the Edge-Cloud Continuum. **Action**: Ensure seamless integration of various connectivity interfaces. **Object**: Connectivity Interfaces (CT) should support reliable data transfer. **Constraint and/or Value**: Must support at least 5 various connectivity interfaces, like 5G, Wi-Fi, Ethernet, LPWAN, and Satellite, with minimum throughput of 10 Mbps and minimum data throughput of 1 Gbps for computational expensive IoT tasks per type.	
	Affected components	Physical Layer Node Manager - Device Registration (NDR) Node Manager - Device Monitoring (NDMo)
	Contributing Partner	MEDITECH
	Comment	The connectivity interfaces defined will guarantee that the CoGNETs platform can communicate efficiently over different network environments, ensuring robustness and flexibility of the system. Connectivity interfaces should be resilient and adaptable to network conditions, like variable latency or bandwidth fluctuations, to guarantee continues service and data synchronization over the Edge-Cloud Continuum.
	Classification	Must Have (M)
CT.NFN.002	The connectivity interfaces supported by the system should exhibit high reliability and adaptability to fluctuating environmental conditions. Ensure a minimum connectivity uptime of 99.9% and a failover mechanism that switches between connectivity interfaces within 1 second of detecting a failure. **Action:** Ensure a minimum connectivity uptime and failover mechanism. **Object:** Connectivity interfaces. **Constraint:** Failover mechanism must switch between connectivity interfaces within 1 second of detecting a failure. **Value:** Minimum connectivity uptime of 99.9%..	
	Affected components	Physical Layer - Connectivity Interfaces (CT)
	Contributing Partner	MEDITECH
	Comment	Enhances system robustness, particularly in highly dynamic and distributed IoT environments.
	Classification	Should Have (S)

Funded by the European Union
Project funded by

7.5.3.3 Business Requirements (BUS)

Table 92: CT.BUS requirement

Req. Id	Requirement Description	
CT.BUS.001	As a business stakeholder, I want the system to support a diverse range of connectivity options, so that seamless service delivery can be ensured across various operational environments. **Action:** Support multiple connectivity technologies **Object:** Connectivity ecosystem **Constraint:** Support at least 5 different connectivity interfaces to accommodate different deployment scenarios **Value:** System deployment across various operational environments	
	Affected components	Connectivity Interfaces (CT)
	Contributing Partner	CERTH
	Comment	-
	Classification	Should Have (S)

7.5.3.4 Business Business Technical Requirements (BTC)

Table 93: CT.BTC requirement

Req. Id	Requirement Description	
CT.BTC.001	As a system integrator, I need a standardized interface for managing multiple connectivity interfaces, so that seamless interoperability across diverse IoT infrastructures is ensured. **Action:** Define and implement standardized APIs for managing different connectivity interfaces **Object:** Connectivity management framework **Constraint:** Support multiple connectivity interfaces seamlessly **Value:** Seamless interoperability across diverse IoT infrastructures	
	Affected components	Connectivity Interfaces (CT) Node Manager – Device Registration (NDR)
	Contributing Partner	CERTH
	Comment	-
	Classification	Must Have (M)

Funded by the European Union
Project funded by

8 APPLICATIONS IN COGNETS (PUCS)

To validate the achievement of the main objectives of the proposed CoGNETs system in real-world settings, three PUCs will be introduced and finally will be deployed for interacting with the developed middleware. The first PUC will focus on robotic systems within an industrial environment to address tasks like autonomous packing/unpacking and detection of lost objects. This demonstrator aims to enhance the automation, resource allocation, process time, and compatibility with legacy systems, showcasing the impact of cloud-based middleware on industrial automation. The second PUC will focus on building self-adaptive thermal management systems for sustainable and resilient battery electric vehicles. High fidelity simulations of the entire vehicle on the cloud will be enhanced with data generated by surrogate models in the edge device which all will be orchestrated by the proposed middleware. Finally, the third PUC will integrate IoT solutions with Edge/Cloud computing to enhance healthcare by reducing time, cost, and networking demands, while improving patient care quality through responsive IoT/Edge processes and precise Cloud-assisted AI services. The demonstrator showcases how CoGNETs middleware enables advanced analytics and collaborative computations, ensuring uninterrupted healthcare monitoring, compliance with privacy regulations, and seamless integration of wearable devices, Edge hubs, and virtual health assistants within a household setting. Altogether, the three PUCs cover extensively the needs and requirements of modern Cloud-to-Edge ecosystems, and hence serve as a strong proof of the utility and importance of the proposed CoGNETs architecture.

8.1 PUC1 - MANUFACTURING: CONNECTED FACTORIES

Pilot Use-Case 1 is an industrial Use-Case, which focuses on autonomous robots in factories. In the works of this Use-Case, a demonstrator will be developed, to verify the theoretical findings and test the middleware layer in a protected environment. This document gives the architectural plans and partner's background information to the planned demonstrator. For the development of the demonstrator the industrial and theoretical knowledge of both corresponding partners (FhG-IPK and SIEMENS) are combined. First, industrial problems that will be covered with the demonstrator are introduced. Two tasks are identified. Those are autonomous packing/unpacking tasks and autonomous detection of lost objects (+cleaning up). The tasks are worked on in three stages. First, with the mobile robot (tend-o-bot), then with the stationary robot (picasso cell) and finally with a combination of both. The two robots are located in the FhG-IPK laboratory. Second, the requirements are identified and existing systems (tend-o-bot and picasso cell) are introduced. Finally, KPIs and assessment plan are given to show how the demonstrator will be put up and evaluated. The demonstrator should enhance the automatization of partially-automatized tasks in factories and enhance process time and compatibility to legacy systems. This will show the impact of cloud-based middleware systems on automation tasks in industrial environments.

8.1.1 Objective

PUC1 validates through experiments how the CoGNETs middleware can improve following challenges in factories' production lines:

1) **Optimize the performance of production lines** by managing and exchanging resources and data autonomously. Hereby multiple manufacturing machines, like robot arms, decide how to best distribute resources and data to adapt to performance of production in real-time.

Funded by the European Union

Project funded by

2) **Predictive maintenance of production lines** to facilitate maintaining the production lines regarding system faults, energy outrages and security bridges. The predictions are based on the actual condition of each asset.

3) **Monitoring costs of production lines** and showing potentials of cost reduction. Hereby factories pay for specific services only for the time the specific service is running (up-time).

8.1.2 Why is it relevant for CoGNETs?

Contribute the experimental validation for the overall project objective, which is: "*to develop a Middleware Framework that will empower devices to autonomously organize dynamic IoT-to-Cloud swarm continuums for optimal data processing and seamless service provisioning*."

In particular, PUC1 will build on a distributed robot control system, that is capable of real-time deployment of computation services within the CoGNETs ecosystem. Experiments regarding engineering time, process execution time, process quality, data integrity and overall service availability will show the potentials of networked IT infrastructure in the manufacturing industry domain.

In addition, PUC1 will be extended by using CoGNETs core and user services, like AI models and computing functions. As the computing power of industrial control equipment is limited, there will be a huge advantage of connecting manufacturing sensors and equipment to an AI backbone and allowing an autonomous inference and control feedback respectively. This will show the potentials of AI-driven feedback control in the manufacturing industry domain. The PUC1 will open the CoGNETs middleware and infrastructure to industrial applications.

8.1.3 Actors

For the proposed PUC1 scenario, six actors have been identified, as described in the following Table 94.

Table 94: PUC1 Actors description

Name	Type	Description	Further information related to this PUC
Robot Operator	Role	Manufacturing process engineer, responsible for robot task allocation, programming and evaluation	
AI Service Provider	Role	Stakeholder responsible for the services performing visual data analysis and robot task planning	
Tendobot mobile manipulation robot	Resource	Mobile robot including mobile platform, lightweight collaborative robot arm, 2D/3D camera, gripper, wireless communication devices	
pICASSO static robot	Resource	Industrial equipment including KUKA	

Funded by the European Union
Project funded by

assembly cell		robot arm, 2 conveyor Belts and hydraulic actuators, wired communication devices	
Edge Server	Resource	IoT-enabled device for sensor data processing, robot path planning and communication with the CoGNETs middleware	Add “Edge Device” = NUC
CoGNETs Middleware	Resource	Distributed middleware framework that provides dynamic cloud resource organization, optimal data processing and seamless service provisioning	

8.1.4 Requirements and assumptions

PUC1 is an industrial Use-Case, which uses industrial robots to automatize not-yet (fully) automatized tasks in production lines and factory environments. To set up the demonstrator that is developed for this Use-Case, hardware, software, logical and logistical requirements have to be given. For every Requirement, the assumptions have to be fulfilled to guarantee a smooth setting up and integration of the demonstrator into the CoGNETs project. The software and hardware requirements are conditional.

Hardware requirements: The actors that are introduced in the previous section map directly onto the needed hardware. The hardware is provided by FhG-IPK and has to be accessible and functional for the development of the PUC1-demonstrator. Additionally, cameras and objects (e.g. tools) for training are needed.

Software requirements: Fraunhofer's contribution is in the field of hardware (robotics) and currently doesn't have a need for a software development infrastructure. The need potentially encompasses CoGNETs cloud for data storage (data accumulated by mobile robots). Object detection infrastructures are already available in the FhG-IPK laboratory. The CoGNETs middleware developments will be integrated into the existing systems.

Logical requirements: For the connection of the robots and the decision making of which robot does which task, the CoGNETs middle ware layer and cloud based infrastructures are needed. The developed ideas - like bidding of resources - will be needed and used. Logical structures of these ideas are necessary to distribute tasks amongst the robots.

Logistical requirements: The logistics will be provided by SIEMENS and FhG-IPK. No further CoGNETs infrastructure is needed.

Funded by the European Union
Project funded by

8.1.5 Workflows between actors

The following scenarios were identified:

Figure 29: PUC1 scenarios

The following scenarios / (user stories) were identified:

1. Picasso robot cell scenario alone
    a) Assembly and handling using object detection and localization and path planning from AI layer
2. tendobot mobile robot alone
    a) Machine tending using camera, self-localization and anomaly detection
    b) Harvest tomatoes
3. both together (“complex” interaction)
    a) Tendobot searches for missing tools and deliver them to the storage rack (=picasso) using object detection, localization and path planning from AI layer.
    b) Handling of fruits and vegetables between mobile robot (logistic) and static robot (storage).

The Tend-o-bot (UR) will move freely between the production lines of the factories to gather image processing data. The production lines are the picasso cell (KUKA) and SIEMENS experimental production line. The image data is collected using an on-board camera. Other data, like workers task schedules and production line machine states, are collected by the machine nearest by (the tend-o-bot for example). The gathered data will be distributed and processed at the near edge via the CoGNETs Collaborative Federated Learning paradigm. If processing (computing) resources don’t suffice, the swarm is extended to the far-edge through the dell power edge server. If the gathered information becomes even too large for

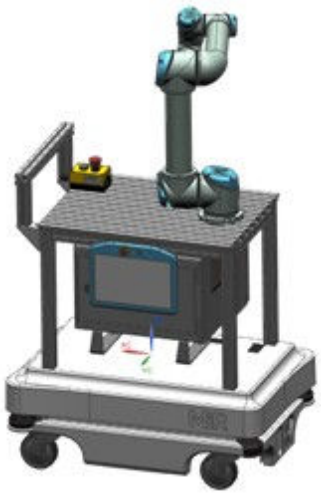
Funded by the European Union
Project funded by

the edge server, it will extend to swarm to the Cloud level by connection to INTRA Stream-handler.

8.1.6 Initial PUC State

PUC1 in the Fraunhofer IPK test-bed consists of two combined scenarios for industrial manufacturing and intra-logistics. The first part is related to the mobile manipulation of work pieces, especially for loading and unloading machine tools. The mobile robotic system is called TENDOBOT. The second part is related to a static robot cell for assembly tasks. The static robotic system is called PICASSO.

Figure 30: Mobile manipulator TENDOBOT (illustration)



The mobile manipulator TENDOBOT (see Figure 30 and Figure 31) is a combination of an Automated Guided Vehicle (AGV) with a robotic arm and accompanying value-added components. These value-added components include a cloud-based trajectory planning component as well as camera technology and interchangeable grippers. In combination, the system enables the automatic loading of machine tools with different workpieces without the need for manual and order-specific programming of the robot.

The collision-free trajectory planning of the robot with the workpiece depends on the actual positioning of the AGV in front of the machine, as well as the respective interior space and the existing clamping device in the machine. The localization of the AGV relative to the machine is achieved by capturing and processing features on and within the machine using 2D and 3D camera technology.

The trajectory calculation for robot control takes place on a powerful, external computing unit in the building (Edge Device) to reduce the capacity requirements of the onboard computing unit. The trajectory can allow for accuracy tolerances to start the robot movement as early as possible, thus saving additional time. During the approach to the target point, with higher accuracy requirements, the tolerances are reduced, and the positioning of the robot is refined.

Funded by the European Union

Project funded by

Figure 31: TENDOBOT in the lab

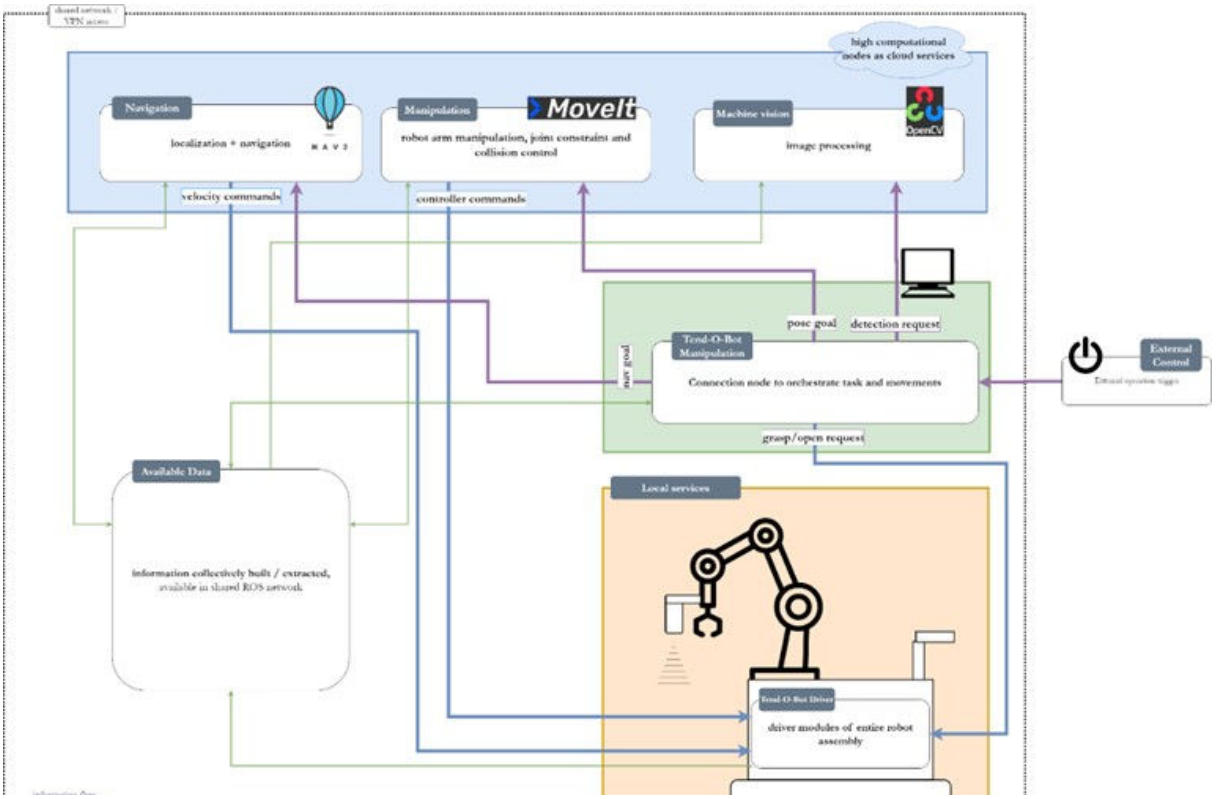
The finished demonstrator independently retrieves workpieces from a structured storage area and delivers them to the respective target machines according to the overarching production planning, placing them in the clamping device there. After processing the workpieces, they are taken to a second storage location and handed over. The system can handle multiple different workpieces in one pass, thereby reducing the number of transport routes. The automatic trajectory planning in the edge device replaces the previous manual teach-in process for each component-machine combination. This significantly increases the flexibility of the system and simplifies the adaptation to further application cases. The centralization of the necessary computing power in the edge device allows for a reduction in the required planning time and simultaneously lowers the costs of the individual vehicles. New geometries for raw and finished parts, as well as the models of machine tools or environmental conditions, such as structural obstacles or bulky components, are centrally managed and seamlessly integrated into the control of the handling system.

The control systems for TENDOBOT is implemented using the ROS2 framework and middleware. Therefore, the hardware functions and the control functions are independently developed and they are communicating with each other over the built-in DDS. ROS2 additionally provides community-driven planning, simulation and visualisation tools for all types of robots. Figure 32 shows the TENDOBOT system architecture.

Funded by the European Union

Project funded by

Figure 32: TENDOBOT system architecture



The static robot PICASSO is used for the second part of the Testbed. The scenario under consideration involves a Kuka Agilus industrial robot, two conveyor belts with electric drives and corresponding frequency converters of types Siemens V90 and G120C, as well as a Festo valve compound with four pneumatic drives. The control of the devices is implemented via Profinet, and the robot is controlled via the mxAutomation protocol also over Profinet.

The use case demonstrates the time-synchronous interaction of the industrial robot with the conveyor belt. The workpieces transported by the conveyor belt are manipulated by the robot, and the positioning of the workpieces is time-synchronized with the conveyor belt. The following sequence is considered in this scenario:

1) The robot lifts a workpiece from the conveyor belt by activating a suction gripper after approaching.

2) The empty conveyor belt moves to the other side of the robot cell while the robot simultaneously moves the workpiece above the conveyor belt in the same direction. Therefore, the movements are synchronized, and the process continues only after both components have completed their movements.

3) The robot places the workpiece on the conveyor belt by deactivating the suction gripper.

Funded by the European Union
Project funded by

4) The conveyor belt moves the workpiece back to the starting point, and the robot moves in the same direction with an empty gripper. Again, the movements are synchronized. The process continues only after both components have completed their motions.

5) The sequence is repeated from point 1.

An extended use case has already been conceptualized and mechanically planned, with some parts already built. In this scenario, the workpieces will be moved back via a second conveyor belt and potentially sorted out. To enable the circular transport of workpieces, a second conveyor belt has been installed. This represents the return transport of defect-free workpieces. Workpieces that are assessed as faulty can, however, be ejected. The assessment of workpieces using camera technology and image processing in the Edge Device as well as the corresponding reaction of the control system are work in progress. However, all involved devices and components have already been projected and put into operation in a virtual PLC. The following sequence is planned for the extended use case of the control:

1) Conveyor belt 1 moves the workpiece from right to left.

2) The robot places the workpiece from conveyor belt 1 onto conveyor belt 2.

3) Camera and image processing assess the condition of the workpiece.

4) Conveyor belt 2 moves the workpiece to the right.

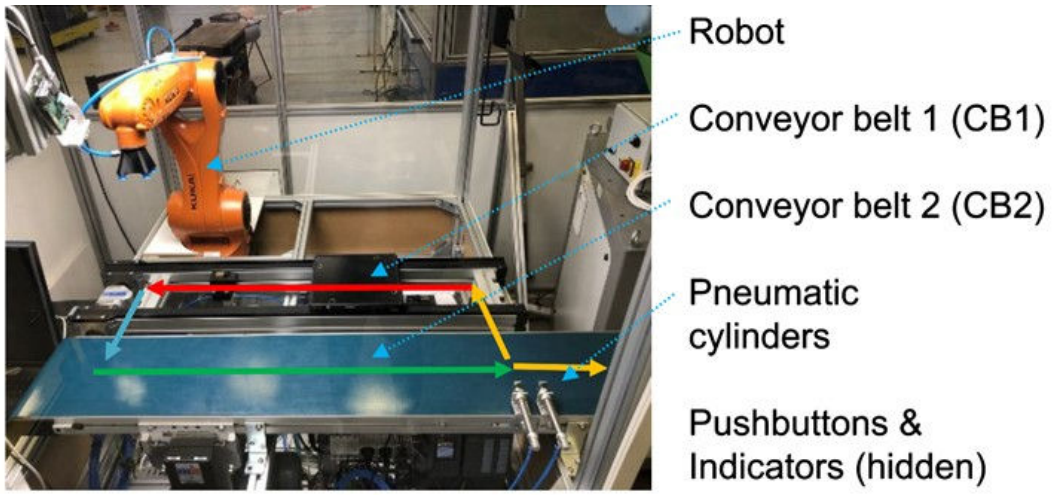
5) The visual workpiece inspection takes place.

6) Pneumatic workpiece sorting based on visual classification.

7) Repeat the sequence from point 1.

This sequence is illustrated in the following figure.

Figure 33: Static robot cell PICASSO with synchronized motion sequence



Funded by the European Union
Project funded by

8.1.7 Expected outcomes

In the initial state of PUC1, manufacturing equipment (e.g. TENDOBOT and the PICASSO cell) makes only partial use of local or on-premise edge servers. By integrating CoGNETs at this stage, computing task allocation becomes dynamic and data processing more efficient. For this purpose, flexible resource sharing will be one of the expected outcomes. With CoGNETs agents coordinating device-level intelligence, it becomes easier to scale or shift computation between on-premise edge servers and cloud resources whenever workloads spike or energy constraints arise. Another outcome is related with the improved production throughput. The system can automatically offload complex tasks such as AI-based vision algorithms to high-capacity nodes, reducing delays in robot local processing and allowing production lines to sustain higher throughput. In addition, cost-effective scalability will be expected because resources can be tapped on demand (locally, at the far-edge, or in the cloud), so only the additional costs will be paid only for extended processing power when needed. Stronger data security will be enforced because CoGNETs infrastructure includes features like secure data integration, so even as robots and edge servers share data, sensitive information can remain protected—helping manufacturers comply with security requirements without compromising the performance.

8.1.8 Specific facilities

Connected scenarios: benefit for PUC1

1. use specific AI model to identify objects in images
2. improved interaction between robots (higher automation degree using CoGNETs data models or computing functions)

Try to establish it this way:

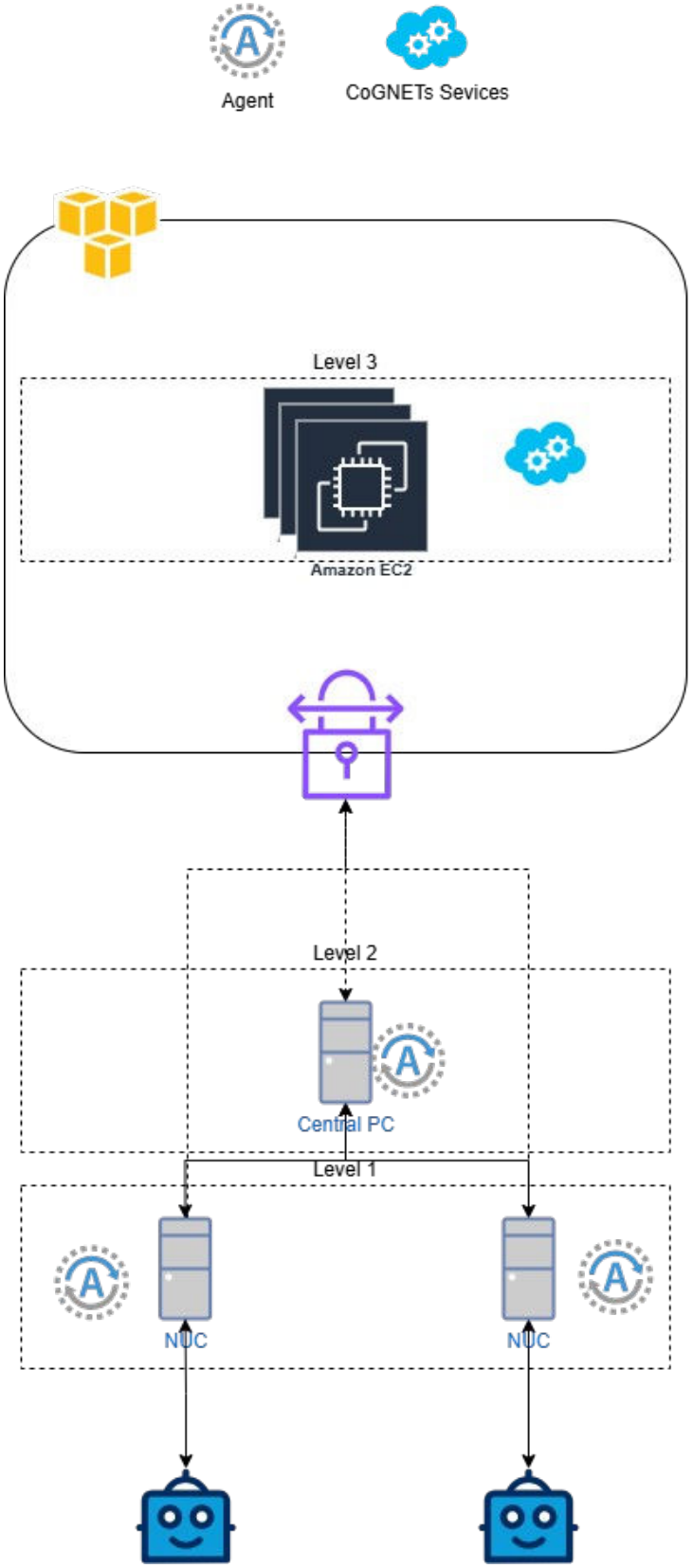
1. Each robot has a node.
2. Each communication framework has a node.
3. Use this to communicate.

As depicted in Figure 34, the deployed CoGNETs agents will interact with the middleware to expand the swarm to the far-edge cloud whenever the robot's computing demands or energy usage surpass predefined thresholds, or if data processing at the Dell server grows too large. Contribute calculation resources to the other PUCs. (2nd stage)

Funded by the European Union

Project funded by

Figure 34: System diagram presenting the integration with CoGNETs components



Funded by the European Union
Project funded by

8.1.9 Production needs

Table 95: PUC1 production needs description.

ID	Name	Description
PUC1.PN.001	Device Scalability	IoT-enabled robots and edge devices for real-time sensor data collection and motion command communication, equipped with sensors and communication modules for reliable data transmission in diverse industrial settings.
PUC1.PN.002	Edge Infrastructure	Compact and powerful computing units for preliminary data processing and running lightweight AI models, with secure communication to the cloud and compliance with data privacy regulations.
PUC1.PN.003	High-Performance Cloud Computing	High-performance cloud computing systems for advanced AI model execution, complex analytics and image processing to ensure secure data processing without transferring raw data from edge devices.
PUC1.PN.004	Software/Hardware Security	Robust encryption, anomaly detection and secure access control mechanisms to protect sensitive manufacturing data and maintain system integrity across IoT-Edge-Cloud infrastructures.
PUC1.PN.005	Software Optimization	Development of AI models optimized for edge and cloud deployment, middleware for seamless data flow across components and user-friendly interfaces for robot operators.
PUC1.PN.006	Interoperability Standards	Compatibility across the computing continuum to adapt to diverse manufacturing workflows, supporting scalable and reliable system deployment.

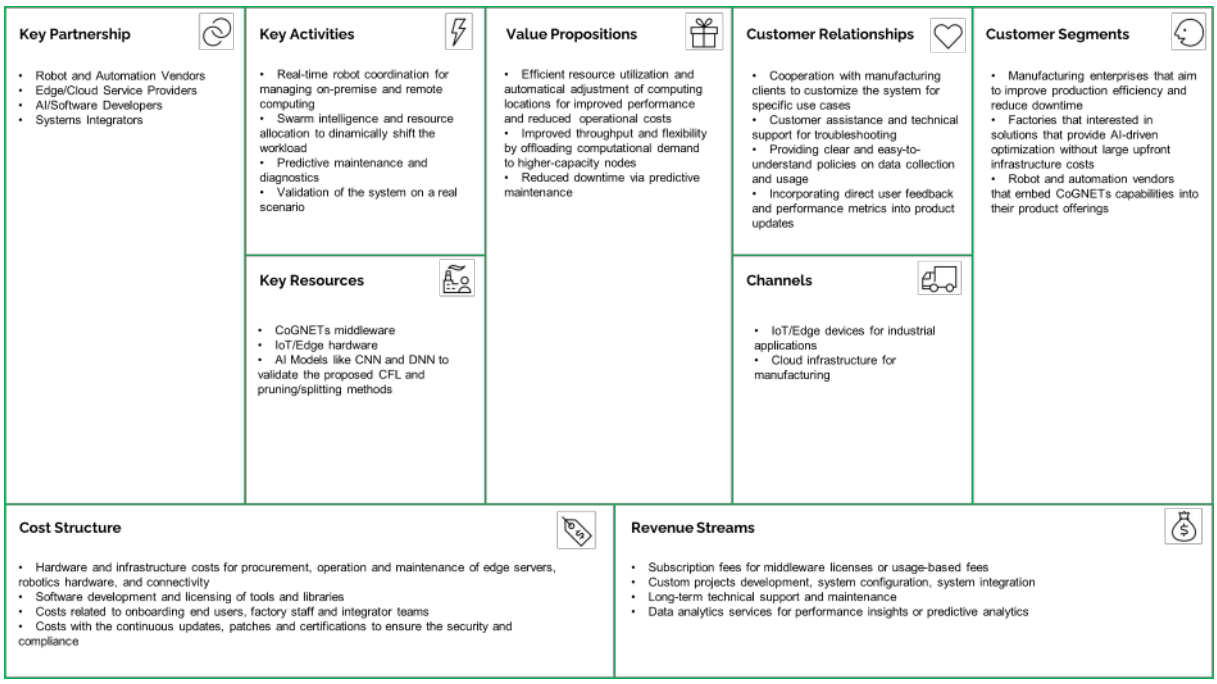
8.1.10 Business model

Business Model Canvas[1] for PUC1 which applies CoGNETs within an Industry 4.0 context is presented in Figure 37. It outlines how robot-integrated edge computing and cloud-enabled AI come together to create a flexible, high-performance manufacturing environment.

[1] https://www.strategyzer.com/library/the-business-model-canvas

Funded by the European Union
Project funded by

Figure 35: Business model canvas for PUC1



8.1.11 KPIs and performance thresholds

For the PUC1 ten KPIs were defined which are presented in Table 96.

Table 96: PUC1 KPIs

ID	Description
PUC1.KPI.01	100% automation in swarm decision-making and formation
PUC1.KPI.02	100% optimization process sustainability
PUC1.KPI.03	>90% balance between QoS, Security and Energy Consumption
PUC1.KPI.04	>90% data privacy preservations
PUC1.KPI.05	100% customization and human accessibility
PUC1.KPI.06	Service response: <1ms for robot-to-robot & <30ms for robot-to-server communication
PUC1.KPI.07	>90% service accuracy
PUC1.KPI.08	>95% compatibility with legacy systems
PUC1.KPI.09	<120% process cycle time (the increase is okay if it stays below 20%)
PUC1.KPI.10	service availability

Funded by the European Union
Project funded by

PUC1.KPI.01. This KPI ensures that all decisions related to forming and reconfiguring the swarm occur autonomously without human intervention. The system is designed to dynamically assess available resources and environmental conditions, enabling robots and devices to self-organize efficiently. This full automation minimizes delays, reduces manual error, and allows the system to respond rapidly to changing operational demands, ensuring optimal resource allocation and seamless coordination across the production environment.

PUC1.KPI.02. This KPI reflects the system's ability to continuously optimize performance over time, balancing computational load, energy consumption, and operational efficiency. By sustaining the optimization process at 100%, the system ensures that it adapts to both short-term fluctuations and long-term trends in workload and resource availability. This constant refinement drives sustainable performance improvements, reduces waste, and maintains peak operational efficiency across the entire manufacturing process.

PUC1.KPI.03. Achieving over 90% balance among quality of service, robust security measures and efficient energy consumption indicates that the system effectively manages competing priorities. This KPI demonstrates that the CoGNETs platform is capable of delivering high-performance service without compromising data protection or incurring excessive energy costs. The harmonious integration of these elements results in reliable, secure and energy-efficient operations which is critical for maintaining a competitive edge in modern manufacturing environments.

PUC1.KPI.04. This KPI ensures that more than 90% of data privacy protocols and protections are maintained throughout all system operations. By strictly enforcing data security measures and adhering to regulatory standards, the CoGNETs platform protects sensitive information against unauthorized access or breaches. Consistent high-level privacy preservation builds trust with customers and stakeholders, ensuring that the system complies with industry and legal requirements while safeguarding critical production data.

PUC1.KPI.05. This KPI guarantees that the system remains fully customizable and accessible to human operators, allowing for tailored configurations that meet specific operational needs. By providing a user-friendly interface and flexible adjustment options, the CoGNETs platform enables end-users to modify parameters, integrate legacy systems, and adapt workflows with ease. This complete accessibility ensures that the system can be seamlessly integrated into diverse manufacturing settings, empowering operators to fine-tune performance and respond swiftly to real-world challenges.

PUC1.KPI.06. This KPI measures the system's latency in communication, ensuring that inter-robot messaging occurs in less than one millisecond and that communications with the server are completed within 30 milliseconds. Such rapid response times are critical for real-time decision-making and coordination, enabling robots to operate synchronously and execute tasks without noticeable delays. This high-speed connectivity supports efficient workflow integration and contributes to overall production reliability.

PUC1.KPI.07. Achieving over 90% service accuracy signifies that the system's AI-driven processes, sensor analyses, and decision-making protocols produce highly reliable and precise outcomes. This KPI reflects the system's capability to deliver consistent results in tasks such as object recognition, path planning, and anomaly detection. High service accuracy minimizes operational errors, enhances process efficiency, and directly contributes to improved production quality and reduced downtime.

Funded by the European Union
Project funded by

PUC1.KPI.08. This KPI ensures that the CoGNETs solution can seamlessly integrate with over 95% of existing legacy systems and industrial protocols. By maintaining high compatibility, the platform can leverage current investments in manufacturing infrastructure while introducing advanced automation and data analytics capabilities. This broad interoperability minimizes disruption during deployment and accelerates adoption, facilitating a smoother transition to modern, connected manufacturing environments.

PUC1.KPI.09. This KPI is aimed at ensuring that the integration of the CoGNETs platform does not extend the process cycle time by more than 20% compared to current operational benchmarks. Maintaining cycle times within these limits is essential for preserving production efficiency. By optimizing task scheduling and resource allocation, the system helps prevent delays that could slow down overall manufacturing throughput, ensuring that automation enhancements translate into real productivity gains.

PUC1.KPI.10. High service availability is a critical KPI that guarantees the continuous operation of the CoGNETs platform, minimizing downtime and ensuring that critical services remain accessible at all times. This is achieved through redundant systems, proactive monitoring, and rapid incident response mechanisms. Consistent service availability is fundamental to maintaining production continuity and ensuring that the manufacturing processes can run smoothly, even in the face of unforeseen challenges or technical disruptions.

8.1.12 Guidelines to validate the KPIs

PUC1.KPI.01. This KPI cannot be directly measured by our internal systems. Qualitative user feedback or third-party evaluations should be used to assess the degree of automation achieved in swarm decision-making and formation.

PUC1.KPI.02. Long-term performance monitoring and trend analysis over extended periods should be combined with qualitative assessments to determine whether the system consistently sustains its optimization processes.

PUC1.KPI.03. Energy consumption should be measured using dedicated devices on both the robot cell and mobile robot hardware, with additional logging of the computing and communication overhead introduced by CoGNETs. QoS can be validated by recording latency and bandwidth within the use case setup, while security metrics though more challenging to quantify should be assessed using appropriate security testing tools or third-party evaluations.

PUC1.KPI.04. This KPI cannot be directly measured internally. Its validation should rely on compliance with regulations such as GDPR and assessments of encryption, anonymization and access control measures to ensure that data privacy is maintained at the required level.

PUC1.KPI.05. Validate this KPI by testing the system's external configuration dimensions. Assess which features are locked versus those that can be customized through the user interface and configuration settings, ensuring that the system remains fully adaptable to user needs.

PUC1.KPI.06. During controlled robotic use case experiments, use precise time-stamping and logging tools to measure the latency and response times for both robot-to-robot and robot-to-server communications to ensure they meet the target thresholds.

PUC1.KPI.07. Measure service accuracy by evaluating the positioning and path accuracy of the robots using an external optical tracking system (such as OptiTrack). Compare the actual robot trajectories with planned ones, taking into account variations caused by different control systems and network conditions.

Funded by the European Union
Project funded by

PUC1.KPI.08. Conduct thorough integration and interface tests to ensure that after new interfaces are implemented, all legacy systems remain fully functional. Compatibility should be verified by testing all legacy protocols and interfaces.

PUC1.KPI.09. Record the end-to-end process cycle times (e.g. pick&place or intra-logistic operations) during controlled experiments and compare these against baseline values.

PUC1.KPI.10. Continuously monitor and log the system's uptime and downtime across all CoGNETs middleware components. Availability should be calculated as the ratio of operating time to the total observation period, ensuring that the system remains accessible and operational throughout all manufacturing activities.

8.1.13 Data Models

There is a use case in which related data are stored and possibly distributed in the system (e.g. Environment map, 3D model of objects, control method parameters). The systems and infrastructures exist already partially in the FhG-IPK laboratory. Following questions regarding data models will be answered by the development of the demonstrator:

- How are images transferred to middleware layer?
- How is the data processed by the middleware layer (AI model/network)?
- How are the results transferred back to the CoGNETs edge node (= Celsius edge cloud)?

PUC1 is a hardware-based Use-Case. For this reason, data models and the AI training plays a secondary role. Image recognition is done by existing structures (in the FHG-IPK lab) with point cloud based methods. The collected data, that is collected by the tend-o-bot robot, will be stored as raw data in the CoGNETs cloud and can be used there for trainings. For this, the data models provided by the middle layer infrastructure will be used. It is possible to work with either CNN or DNN or different networks.

Used Hardware:

- Edge Device = NUC (connected to Edge Cloud via ROS2/OPC UA)
- Edge Server = Celsius (is CoGNETs node, connected to CoGNETs core via Streamhandler)

8.1.14 End-user Service Components

All actors should have interfaces to the system.

- The robot operator should interact with the robotic systems and the CoGNETs middleware through sophisticated graphical user interfaces. The integration of the definition of automation task, robot programming and configuration of CoGNETs services, e.g. using AI models and functions, should be seamless.
- The AI service provider should interact with the CoGNETs middleware, AI models and functions as well as the data sources, e.g. cameras and other sensors. The integration of the graphical user interface for data management, sensor connection and access to AI models and functions should be seamless.

Funded by the European Union
Project funded by

- The Tendobot robot relies on external control, thus giving sensor information to the control system and accepting driving commands. As the control system, or parts of it, are running within the CoGNETs framework, the interfaces should be configurable using different data transmission paths. The robot operator has to be able to select the available control functions and paths within the user interface respectively.

- The pICASSO robot relies on external control, thus giving sensor information to the control system and accepting driving commands. As the control system, or parts of it, are running within the CoGNETs framework, the interfaces should be configurable using different data transmission paths. The robot operator has to be able to select the available control functions and paths within the user interface respectively.

- The service components have to be implemented as deployable assets that are accessible from mobile devices near the robotic systems, e.g. Edge Server or Edge Device, to configure related behaviour.

The interfaces to all actors are set up in the development process of the PUC1 demonstrator.

8.1.15 Risk Assessment & Mitigation Plan

The implementation of the CoGNETs PUC1 comes with some risks. In Table 97 we present the list of identified risks and the mitigation plan for each of them

Table 97: Risks and mitigation plan for PUC1

ID	Name	Probability	Damage score	Mitigation plan
PUC1.R.01	System downtime and hardware failure	Medium	High	Implement redundant hardware, schedule maintenance, real-time monitoring tools to detect and resolve issues
PUC1.R.02	Cybersecurity threats and data breaches	Medium	Very high	Robust encryption, multi-factor authentication, secure data exchange protocols
PUC1.R.03	Integration with legacy systems	High	Medium	Develop compatibility layers, collaborate with experienced systems integrators
PUC1.R.04	Scalability Issues with Increasing Workloads	Medium	Medium	Design a modular, scalable architecture with dynamic resource allocation and load balancing. Continuously monitor system performance to adjust capacity as needed

PUC1.R.01. This risk involves unexpected hardware issues or system outages that could significantly disrupt manufacturing operations. With a medium probability and high potential damage, downtime can result in costly delays and loss of productivity. To mitigate this, redundant hardware should be implemented and regular maintenance scheduled. Additionally,

Funded by the European Union
Project funded by

deploying real-time monitoring tools will help detect and resolve issues swiftly, ensuring continuous system operation.

PUC1.R.02. Cybersecurity threats carry a medium probability but have very high potential damage, as breaches can compromise sensitive production data and disrupt operations. Robust encryption protocols, multi-factor authentication and secure data exchange protocols are essential to safeguard the system. Regular security audits and continuous monitoring are also critical to promptly identify and remediate vulnerabilities.

PUC1.R.03. Integration with existing legacy systems poses a high probability risk with medium damage impact, potentially leading to operational incompatibilities and inefficiencies. To mitigate this risk, compatibility layers should be developed to ensure seamless communication between the new CoGNETs solution and legacy components.

PUC1.R.04. As production demands increase, scalability issues may arise, presenting a medium probability risk with medium damage impact. Without proper management, the system could struggle under peak workloads, leading to performance degradation. Designing a modular, scalable architecture with dynamic resource allocation and load balancing is key to addressing this risk. Continuous monitoring of system performance will allow for timely adjustments, ensuring the infrastructure can handle fluctuating workloads efficiently.

8.2 PUC2 - MOBILITY: CONNECTED VEHICLES

The second PUC is focused on developing AI-driven solutions for software-defined electric vehicles within the edge-cloud continuum. Due to ever increasing pressure on lowering the CO_2 emissions, novel and sustainable technological solutions need to be developed and introduced to the traditional European automotive industry which is now facing huge challenges from overseas competition and strict regulatory authorities. We propose a seamless Edge-Cloud architecture to optimally control the thermal management system of electric vehicles by deploying state-of-the-art reinforcement learning algorithms. Next subsection will outline all the necessary requirements and components of the PUC that will be interacting with the CoGNETs Middleware.

8.2.1 Objective

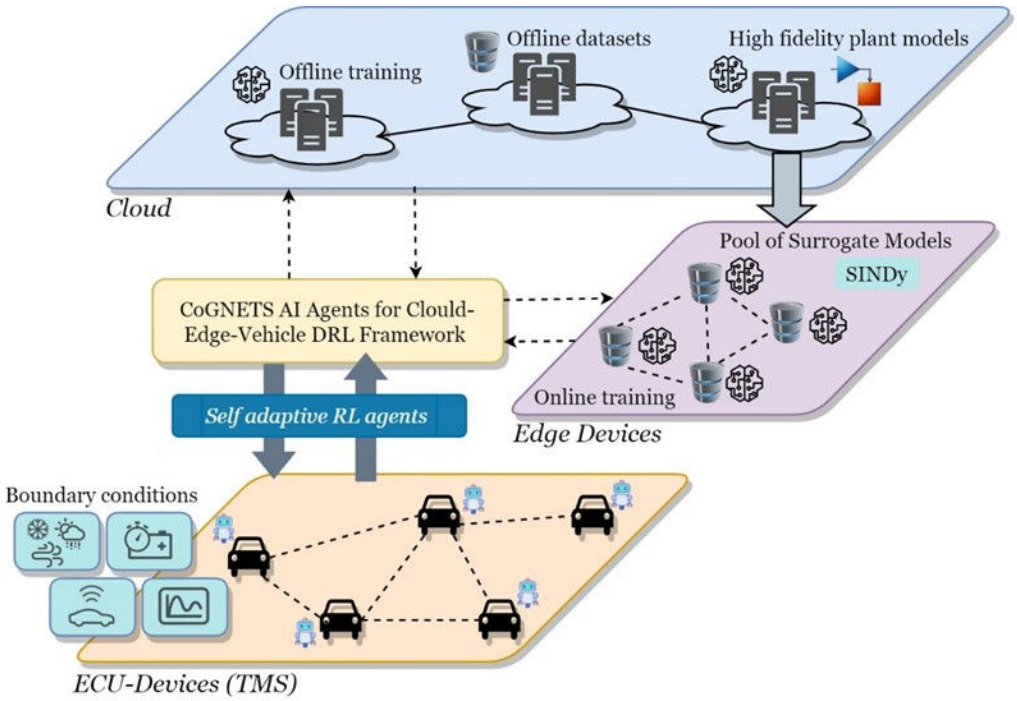
PUC2 – “Mobility: Connected Vehicles (Adaptive powertrain strategies for Battery Electric Vehicles” – aims to leverage state-of-the-art Reinforcement Learning and Edge-to-Cloud technologies to enhance performance and sustainability of modern automotive electric powertrains. Existing control strategies in thermal management systems of BEVs, on the one hand, may lack of robustness to continuous charging/discharging operating conditions, and, on the other side, may be time consuming and costly for experts to calibrate the controllers. In addition to the above challenges, high fidelity simulation and multiple testing is needed, which might further increase engineering overhead. Furthermore, training of intelligent RL-agents in the edge device of the vehicle for complex real-world driving scenarios is still a huge challenge in the industry. The proposed solution integrates advanced computation and AI technologies with an on-demand approach over the entire IoT-to-Cloud continuum to seamlessly optimize overall performance and robustness in the thermal management system of the vehicle. Ultimately, these kinds of novel technologies lay the foundation for the software-defined vehicles of the future.

The following figure illustrates the architecture of the CoGNETs middleware and its interaction with all necessary components and modules for the PUC2 connected vehicles scenario. In the lower physical level, on-board sensors and IoT-enabled devices serve as primary data collection units, continuously monitoring vehicle-battery performance, driver behaviour, and

Funded by the European Union
Project funded by

other external conditions (i.e., weather). In addition to these parameters, vehicle connectivity and battery range are key indicators for defining the boundary conditions transferred to the CoGNETs Middleware. In the upper cloud layer, huge amount of data is stored which are also utilized to feed high-fidelity simulations to model the thermal management system of the vehicle. Data will be then grouped according to various operational scenarios, and sent to edge units in the vehicle that can train reduced-order model for the defined environment. This is essential to guarantee uncorrupted and robust decision-making in the vehicle's controllers.

Figure 36: PUC2 architecture overview



Principally, PUC2 validates through experiments how the CoGNETs middleware can improve following challenges in thermal management control systems of BEV:

- **Deploy a RL-based self-learning control strategy** by utilizing both data from first principle and reduced order models in-cloud and on-edge, respectively.
- **AI Game Agents** that will monitor and aggregate data based on vehicle's/fleet's operating conditions between cloud and edge devices in the vehicle
- **Real-time monitoring and validation** of agent's training performance via the dashboard

8.2.2 Why is it relevant for CoGNETs?

Contribute the experimental validation for the overall project objective, which is: "*to develop a Middleware Framework that will empower devices to autonomously organize dynamic IoT-to-Cloud swarm continuum for optimal data processing and seamless service provisioning.*"

Running AI-related algorithms for control systems in vehicles is a very challenging endeavour from both functional and computational perspective. Since, current PUC will be focussing on

Funded by the European Union
Project funded by

RL-based scenarios, training the agents so that to achieve continuously high adaptability and robustness needs a seamless availability and connectivity between the cloud and the edge device on the vehicle.

Other challenge that such AI-based controllers on vehicle are facing is the timely data availability that need to be utilized for training the models. In case that RL agents will not gain new experience, will potentially compromise the overall performance and ultimately will lead to false decision making of the AI solution. CoGNETs Middleware can close this gap by offering continuos connectivity with the pool of data and other model resources between cloud and the edge device on the vehicle.

8.2.3 Actors

From the technical side regrading the optimization of the control strategy in the BEV's thermal management system,

- Sensors and actuators (RL agent) in HVAC's electric vehicle.
- Plant model of thermal system ("data generator" for model-based RL environment in Cloud).
- Base HVAC ECU which communicates with the thermal system controls via a communication protocol (edge).

Table 98: PUC2's Actors

Name	Type	Description	Further information related to this PUC
Human Driver	Role	Main actor that actively participating for the optimization of the TMS in the vehicle	Active actor
AI Agent	Resource	RL-based agent to optimize BEV's HVAC operation	Active actor
Edge Device	Resource	IoT-enabled device for preliminary fleet data processing and communication with the CoGNETs middleware	Passive actor
Sensors/Actuators	Resource	Electro-Mechanical devices with embedded sensors of the TMS in the vehicle	Passive/Active actor
CoGNETs Middleware	Role	Distributed middleware framework that provides dynamic cloud resource organization, optimal data processing and seamless service provisioning	Passive actor

Funded by the European Union
Project funded by

8.2.4 Requirements and assumptions

Control systems for EV's thermal systems are highly complex electromechanical systems with very large impact on the overall safety and operation of the vehicle. Strict regulations and requirements need to be fulfilled to achieve optimal and robust performance in diverse real-world operating conditions. The CoGNETs PUC2 scenario will focus to build a RL-based controller in TMS that will operate seamlessly regardless operating conditions.

A very important aspect is the real-time performance of the RL-agent on the control system in order to respond with low latencies during dynamic changes in the vehicle's environment. The framework should be able to collaborate and exchange information with multiple vehicles simultaneously to enhance the learning of the agent in various action-state scenarios.

In case of system failures due to loss of connectivity or other resources, the system should continue to operate uninterruptedly based on existing agents' experience and the data that are generated by the surrogate models. These additive models will have low complexity and can be stored in the edge devices to support the training in the above mentioned fail scenarios.

To ensure proper interoperability with the CoGNETs framework the following system and functional assumptions should be considered:

1. Sensor data availability in the vehicle. Obtaining correct and accurate measuring signals from the TMS is of paramount importance, otherwise wrong decision making in the controller will lead to catastrophic implications in the vehicle. Hence, anomaly detection algorithms must ensure that any deviation from feasible measurements will be immediately detected and mitigated.

2. Stable connectivity ensures reliable and low-latency communication between edge and cloud services.

3. Extreme weather conditions will be included to ensure system stability and robustness, as this might have high impact to cabin comfort and battery range of the vehicle.

4. Cabin environments (plant models) will be approximated with surrogate models that will run on edge devices, since complex ones consume a lot of resources with high costs. The later ones should be only available for training purposes in the cloud.

5. Sufficient computing power is available in the vehicle to train onboard low-complexity models to train the agents

6. A wide range of operational modes will be considered to cover the entire state space of the system. Some examples are active/passive cabin cooling, battery heating etc. According to the selected mode by the driver the corresponding models/data will be selected for training/inference.

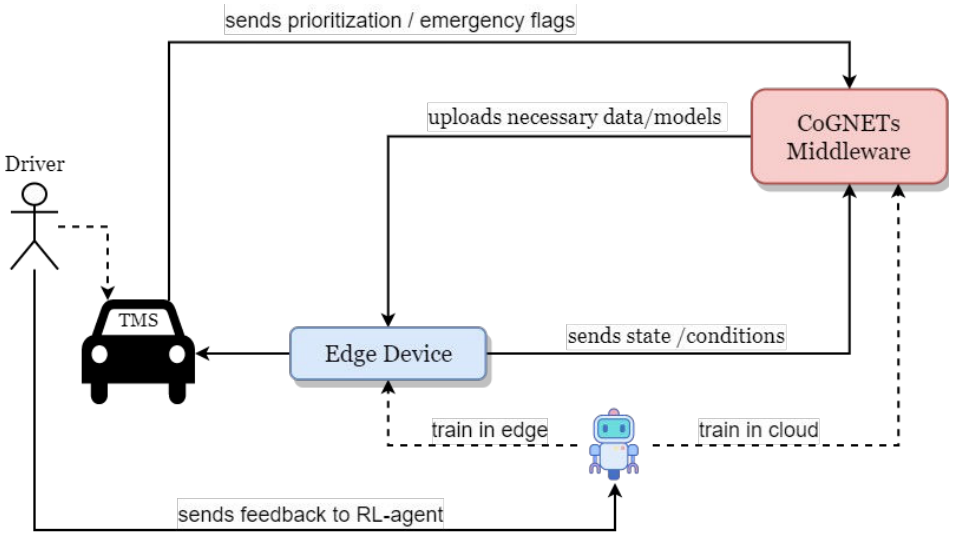
8.2.5 Workflows between actors

To better illustrate the orchestration and the interoperability of the different actors, a diagram is introduced with the roles and their contributions.

Funded by the European Union

Project funded by

Figure 37: PUC2 actors' role diagram



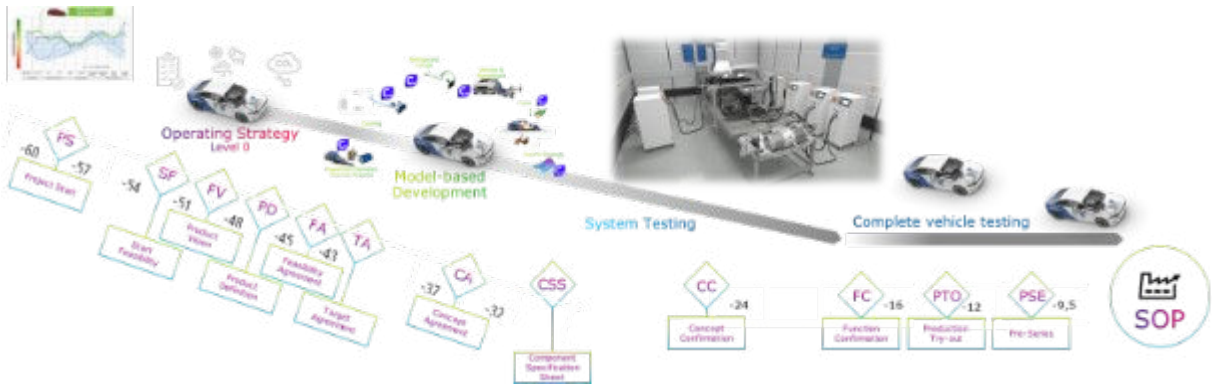
A core role plays the driver as her/his feedback on the thermal comfort must be integrated into the training process (reward function) of the RL agent. The driver will further influence the performance of the TMS based on the type of driving (aggressive or conservative) and the charging mode (full or partial charging). In emergency cases (i.e., battery draining), CoGNETs middleware should be aware of it so that actions to be taken regarding availability of the specific vehicle's contribution for local training in the edge device. A further capability of CoGNETs middleware should be decision making on either train the RL agent in the edge device or in the cloud according to environmental or other internal conditions. For instance, under extreme weather conditions (too cold/hot), the training of the RL agent should take in place in the cloud as more accurate and precise predictions of the agent are required. Either case Federated Learning scenarios that will be happening in the vehicle fleet should facilitate the collaborative learning process of the agents in the BMS.

8.2.6 Initial PUC State

The current approach to refrigerant circuit control for TMS in BEVs primarily relies on a Proportional-Integral (PI) controller combined with feed-forward strategies. This control methodology is widely used to achieve the desired operational behaviour by proactively adjusting system parameters based on expected conditions while simultaneously compensating for deviations in real time. Although this approach provides a stable and effective means of regulating the thermal system, it lacks the flexibility required to seamlessly adapt to different vehicle configurations and external conditions. As a result, significant effort is required to calibrate the control strategy for each specific application, increasing both development time and complexity.

Funded by the European Union
Project funded by

Figure 38: Initial PUC2 state diagram



In particular, a key challenge in implementing thermal system control is the adaptation of the control strategy and its parameters to fit individual vehicle systems. This process alone takes at least a year, as engineers must meticulously optimize the system to meet performance, efficiency, and reliability requirements. Each vehicle platform has distinct thermal management needs, requiring extensive calibration of the control algorithms to ensure smooth operation under varying loads. Since PI controllers rely on predefined gain values, careful tuning is necessary to achieve the right balance between responsiveness and stability. This phase is critical, as any misalignment can lead to inefficiencies, suboptimal cooling performance, or excessive energy consumption.

Beyond initial calibration, the thermal system must undergo rigorous testing under diverse environmental conditions, which further extends the development timeline by another year. Vehicles operate in widely varying climates, from extreme cold to intense heat, requiring extensive validation to confirm that the system remains effective across all conditions. Testing involves both controlled laboratory simulations and real-world field trials, where the system's response to different ambient temperatures, humidity levels, and driving scenarios is analysed. This step ensures that the thermal system performs reliably under dynamic conditions, preventing potential failures that could impact vehicle safety and passenger comfort.

Even after laboratory and field testing, additional validation efforts are required to guarantee proper operation in real-life conditions for end users. Vehicles are exposed to unpredictable external factors, including variations in driving behaviour, traffic patterns, and load conditions, all of which can influence cooling demands. To ensure consistent performance, manufacturers must conduct extensive real-world testing, refine the control algorithms, and, in some cases, implement post-production software updates. This continuous evaluation process is resource-intensive, as any failure to adequately account for real-world variability could lead to customer dissatisfaction and increased warranty claims.

The complexity of the thermal system validation is further exacerbated by the increasing diversity of final vehicle configurations. Even seemingly minor design differences, such as the vehicle's exterior colour, can significantly impact cooling requirements. Dark-coloured vehicles absorb more heat from sunlight, increasing the thermal load on the thermal system compared to lighter-coloured models. This added variability necessitates additional testing and fine-tuning of control strategies to ensure uniform performance across all configurations. However, balancing these additional validation requirements with the industry's push for reduced development costs and shorter time-to-market cycles presents a major challenge. As auto makers strive to optimize both cost efficiency and product quality, finding the right trade-

Funded by the European Union
Project funded by

off between comprehensive validation and streamlined development processes becomes increasingly complex.

8.2.7 Expected outcomes

As highlighted in the previous sections, an expected outcome for PUC2 is to build self-adaptive RL-controllers, which their training can be balanced between the optimal collaboration of cloud and the edge devices that are orchestrated by the CoGNETs Middleware. According to operating and other environmental conditions in the vehicle, the computation burden can be respectively allocated without compromising the energy resources of the vehicle's powertrain system it self.

It is expected that with viable allocation of computational resources and data, RL-agents can perform sequential decision making in the BMS significantly better than the conventional PID controllers. Another outcome is less calibration efforts from the engineers, which leads to more economically sustainable and robust products for the automotive industry.

8.2.8 Specific facilities

Implementation and validation of CoGNETs PUC2 will be performed within specific in-house simulation environment. AVL vSUITE™ offers a combination of physical vehicle and powertrain system modelling capabilities including energy management and overall vehicle performance. Furthermore, it enables the seamless use from concept to test and calibration phase.

8.2.9 Production needs

Table 99: PUC2 production needs

ID	Description
PUC2.PN.01	Energy consumption of the edge devices in the vehicle's ECU
PUC2.PN.02	High performance cloud computing for offline training
PUC2.PN.03	Development of state-of-the-art RL algorithms
PUC2.PN.04	Fail-safe mechanisms

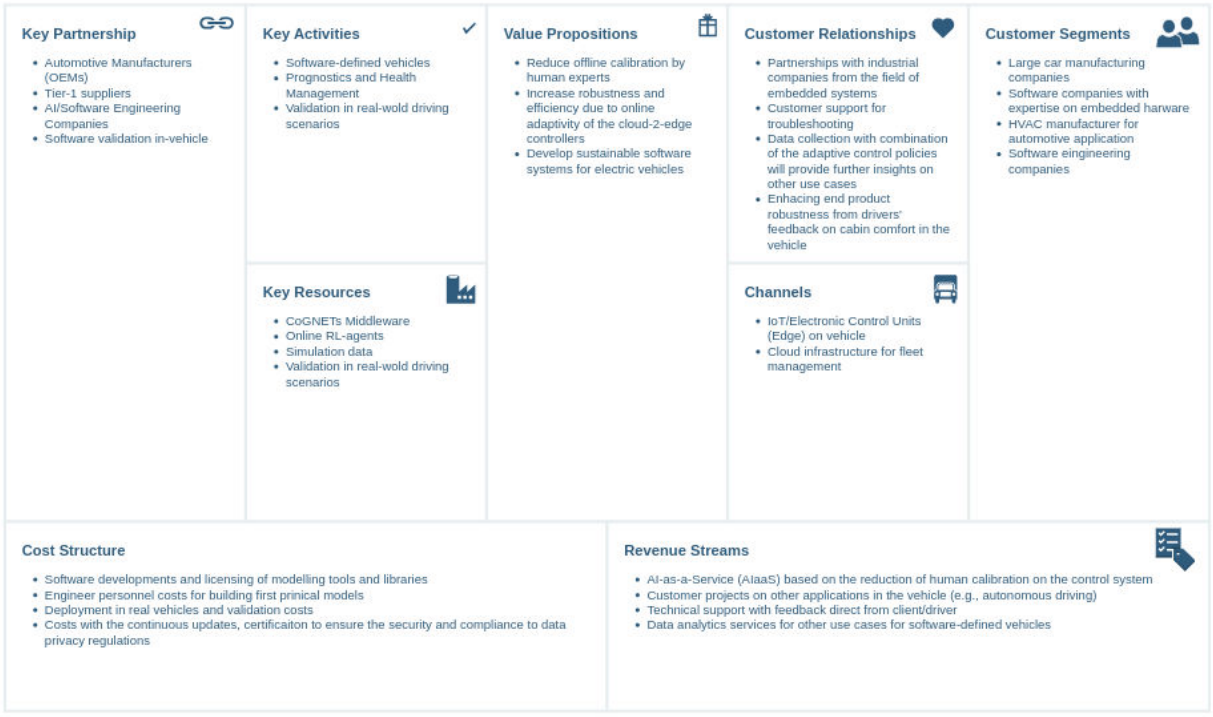
8.2.10 Business model

A high potential business model can be unfolded for PUC2 by integrating the CoGNETs middleware. The following figure outlines all the important business aspects in the field of the automotive industry.

Funded by the European Union

Project funded by

Figure 39: Business model canvas for PUC2



8.2.11 KPIs and performance thresholds

For PUC2, four KPIs were defined are presented in Table 100.

Table 100: PUC2 KPIs

ID	Description
PUC2.KPI.01	Fully automated allocation of data and computational load for adaptive learning of the RL agents
PUC2.KPI.02	Reduction of development and validation costs
PUC2.KPI.03	Increase of security resiliency
PUC2.KPI.04	Enhance energy efficiency and sustainability

8.2.12 Guidelines to validate the KPIs

- **PUC2.KPI.01**. Successful deployment of CoGNETs middleware to PUC2

  a) Seamless operation of the trained RL-agent by measuring main cabin comfort model output with battery performance.

  b) Minimize offline simulation with high fidelity models in the cloud.

Funded by the European Union
Project funded by

c) Utilize effectively other vehicles' experiences from past simulations.

- **PUC2.KPI.02**. Reduction of development costs

  a) Measuring the reduction of the calibration effort on the controllers of the TMS

  b) Utilization of AVL Data Analytics platform to validate RL-based controllers

- **PUC2.KPI.03**. Security Threat Detection

  a) Simulate cyberattack scenarios both on the state and the control inputs to verify the robustness of the RL-agent. Anomaly detection methods ensure that timely detection of such attacks will be identified and mitigated.

- **PUC2.KPI.04**. Enhance energy efficiency

  a) Measuring efficiency of HVAC systems for electric vehicles

  b) Measuring of improvement on the battery ageing due to less consumption towards the TMS

8.2.13 Data Models

Generally, we are planning to utilize a similar structure from an existing smart data model which shares similar high-level objectives as in PUC2. Principally, the PUC2 represents in real-world settings a complex engineering system of the electric vehicle consisted of multiple subsystems and modules. Each subsystem of the vehicle with its Thermal Management System will be mapped to the main classes below which should encapsulate several attributes and functions.

- **DrivingCycle**: class representing the different driving scenarios and boundary conditions of the vehicle. Driving-specific attributes can be GPS route, weather, profile speed, while environmental-specific ones are ambient air temperature, ambient air humidity, solar intensity and target cabin temperature.

- **VehiclePowertrain**: class representing which encompasses the main drive system for producing and distributing the mechanical energy to the wheels.

- **PowertrainThermalElements**: attributes from the class **VehiclePowertrain** that represents the mechanical components which produce-dissipate heat losses

- **CoolantCircuitCoolingPackage**: represents the class with the attributes coming from the water pump and other valve components that are responsible to absorb and direct the heat from and to the battery-powertrain.

- **RefrigerantCircuitCoolingPackage**: represents the class of the refrigerant that transfers the heat for efficient cooling and heating of both battery-motor and the cabin.

- **VehicleCabin**: represents the class of the cabin, which needs to be either cooled or heated depending on the ambient temperature. This includes the number of passengers with the control desired temperature.

Funded by the European Union
Project funded by

All above class are essential in order to defined the plant model of the high fidelity models that will be utilized to build the RL environment.

8.2.14 End-user Service Components

The CoGNETs PUC2 scenario deploys state-of-the-art RL algorithms to vehicle's thermal control systems to achieve robust operation and high performance for the e-mobility. The main component that serves as a validation and monitoring interface is a user-friendly visualization dashboard for the vehicle fleet which will provide real-time information on the AI agent's performance. Its key features besides providing the main KPIs to the dashboard, is alerting the fleet operator/driver on emergency events, such as system outages or electric battery depletions.

Specifically, we propose AVL Data Analytics™ for performance monitoring of the developed solution as it may provide comprehensive capabilities, including real-time tracking, event detection, and advanced data visualization. It enhances operational efficiency and transparency through predictive maintenance and cross-comparisons, integrating metadata for thorough analysis. The system's automation significantly reduces analysis effort and ensures early detection of anomalies, optimizing fleet performance and management. CoGNETs middleware will provide a lower level of monitoring capabilities in terms of data and model management and resource allocation. This will guarantee safe and robust deployment to the vehicle's control system.

8.2.15 Risk Assessment & Mitigation Plan

One of the main risks that may be anticipated is the loss of network connectivity which will not allow data transmission and over-the-air update of the AI models on the vehicle's ECU. However, surrogate models with low energy consumption and complexity will provide the sufficient data that might be needed for training the RL agents. Besides this option, originally each vehicle will upfront be mapped with specific driving behaviours and environmental conditions that will be utilized to train offline the agents and upload whenever network is available. Such models should provide the most viable solution to run on the vehicle.

Finally, security risks from external cyber attacks on the vehicle's ECU might have a large impact not only to the overall performance but also to the safety of the electric vehicle itself. By employing state-of-the-art anomaly detection methods with robust encryption algorithms and secure data exchange protocols, such threats can be isolated and mitigated.

8.3 PUC3 - HEALTH: CONNECTED HEALTHCARE

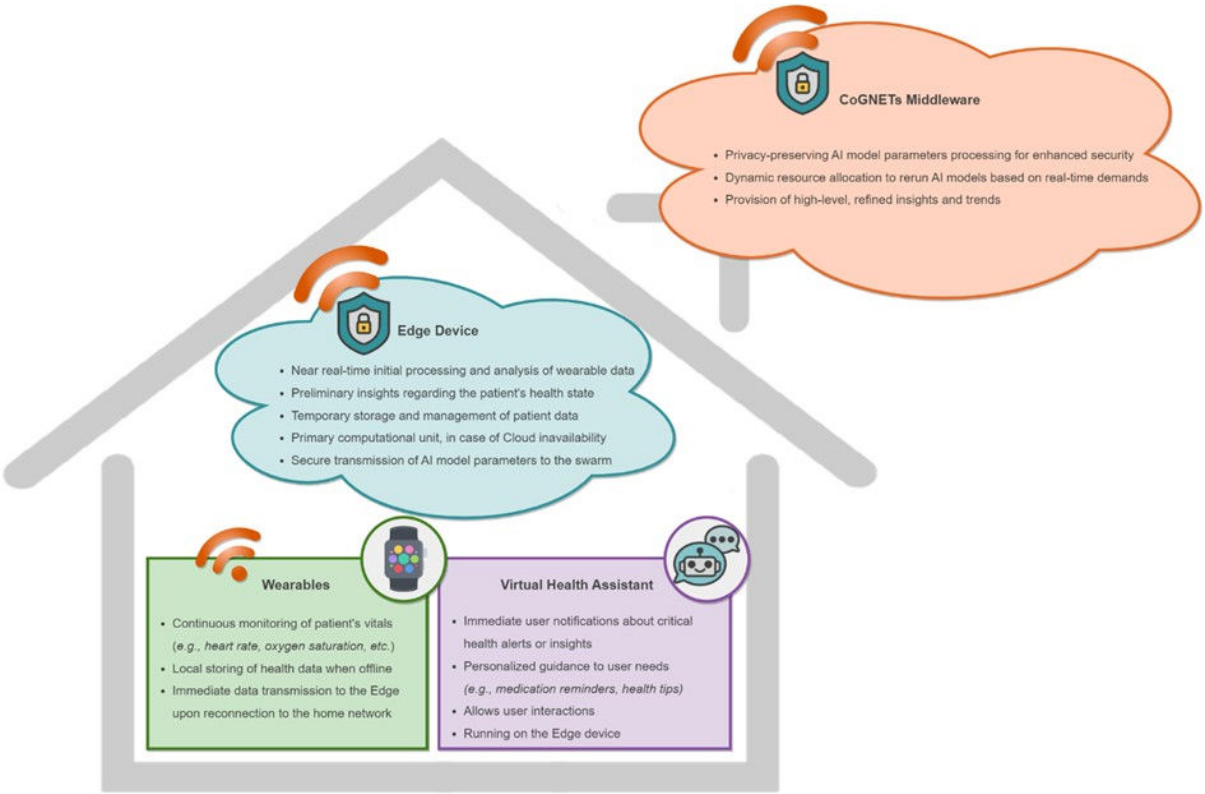
PUC3 – "Connected Healthcare (Collaborative AI for Medical Data Analytics in Health 4.0)" – through integrating IoT solutions with Edge/Cloud computing, opens up novel possibilities for the healthcare supply chain by introducing a new generation of Health 4.0 applications. This new approach not only leads to reduced time, cost and networking demands, but also improves patient care quality, ultimately delivering data-driven, cognitive federated computing solutions that capitalize on the IoT-to-Cloud continuum to enable responsive, locally executed IoT/Edge processes and precise, Cloud-assisted AI-enabled services. As a result, patient access and results are highly improved, while healthcare professionals are now allowed to focus on more critical tasks by automating routine ones.

Figure 40 depicts all the required components for the PUC3 – "Connected Healthcare" scenario, along with their corresponding roles. As seen in the figure, all PUC3 components are

Funded by the European Union
Project funded by

expected to be placed within the same household to enable seamless communication, near real-time processing and secure data exchange among the components. The wearable devices operate as the primary data collection units, continuously monitoring patient vitals and storing data locally when offline. Once connected to the household network, they transmit the collected health data to the Edge device, positioned as the central hub within the household, for preliminary analysis, to generate preliminary insights in real time. The Edge device also ensures that all collected data are processed locally, before transmitting summarized model parameters to the Cloud infrastructure (CoGNETs middleware) for advanced analytics and collaborative computations. Finally, the virtual health assistant, integrated within the Edge device, notifies the patient about critical alerts, health risks or reminders, ensuring personalized guidance and interaction. Meanwhile, the Cloud infrastructure, operating remotely, handles resource-intensive AI model refinements, ensuring compliance with privacy regulations by focusing solely on model parameters without accessing raw health data. The combined operation of these components ensures uninterrupted functionality and reliable healthcare monitoring, even in the event of network disruptions or cloud unavailability.

Figure 40: PUC3 architecture overview



8.3.1 Objective

As mentioned above, the main goal of PUC3 is to demonstrate how the CoGNETs solution can address key challenges in the healthcare sector by leveraging advanced IoT-to-Cloud capabilities to improve medical diagnostics, enhance security and enable scalable telehealth services, thus creating a more efficient and reliable healthcare ecosystem. For the proposed PUC3 scenario, three objectives have been proposed:

Funded by the European Union
Project funded by

1) **Assisted diagnostics for medical data analysis from patients:** Through implementing AI-based assisted diagnostics, the proposed PUC3 scenario aims to showcase the effective use of patient medical data in healthcare, which, under different circumstances, remain underutilized due to their large volume. The proposed AI-based algorithms will allow for rapid and near real-time processing of large datasets to identify patterns, predict health trends and generate valuable, personalized insights for each patient. Since health data are inherently sequential and time-dependent, time series forecasting techniques will be integrated for temporal pattern recognition and anomaly detection, predicting potential health risks and deviations from normal physiological patterns. As a result, proactive healthcare interventions will be offered, while healthcare professionals will be able to analyse and act upon previously inaccessible or unstructured data, leading to improved diagnostics, informed decision-making and tailored treatment plans, enhancing overall patient outcomes.

2) **Secured hospital/home care with more trustworthy diagnosis results:** By incorporating robust cybersecurity measures, including data encryption, anomaly detection and access controls, the proposed PUC3 connected healthcare scenario ensures that healthcare professionals and patients can trust the data used for diagnoses and treatment planning. This is particularly important for both hospital environments and home care scenarios, where the risks of cyberattacks on medical systems and IoT de-vices can compromise patient safety and care quality.

3) **Next-generation telehealth services with AI-assisted services for health monitoring and diagnostics:** By running as home-based applications, the proposed health services provide scalable, accessible and efficient care options, especially in resource-constrained settings. AI-enabled telehealth solutions can monitor patient vitals, analyse symptoms and provide real-time feedback to patients and healthcare providers, reducing the burden on traditional healthcare facilities, thereby offering more inclusive healthcare and ensuring that quality care reaches patients regardless of location or socioeconomic barriers.

8.3.2 Why is it relevant for CoGNETs?

The PUC3 scenario is a vital validation point for CoGNETs, demonstrating how its scalable and interoperable IoT-to-Cloud middleware and Edge can address the challenges of medical data that is underutilized from lack of reliable data analysis, and the lack of robust security mechanisms to protect patients' private data.

In this domain, the vast volume of medical data often remains underutilized due to a lack of advanced processing and integration capabilities. CoGNETs' decentralized federated architecture will enable a near real-time data analysis and assisted diagnostics, hence ensuring improvements in patient access and quality of care. Furthermore, the healthcare sector demands robust security mechanisms to protect sensitive patient information and ensure trust in diagnosis and treatment processes. CoGNETs' middleware integrates end-to-end security features, including RISC-V hardware security, anomaly detection, and secure resource management, ensuring the safety of medical data and devices in both hospital and home care environments.

Additionally, PUC3 emphasizes energy-efficient and sustainable operations by leveraging edge computing and low energy cost medical IoT devices, aligning with CoGNETs' mission to optimize computing, energy, and security holistically. This is crucial for scaling telehealth services, which require reliable, low-latency IoT-to-Cloud infrastructures to provide equitable and accessible healthcare. By showcasing how CoGNETs can deliver AI-driven, secure, and

Funded by the European Union
Project funded by

energy-efficient solutions in healthcare, PUC3 validates the project's ability to enhance diagnostic accuracy, improve patient outcomes, and support Europe's digital sovereignty while ad-dressing the growing demands of modern healthcare ecosystems.

8.3.3 Actors

For the proposed PUC3 scenario, eight actors have been identified, as described in the following Table 41.

Table 101: PUC3 actor description

Name	Type	Description	Further information related to this PUC
Healthcare Professional	Role	Doctors, nurses and other medical staff who can use the system to monitor patient conditions and make better-informed decisions	Active actor
Patient	Role	Individuals receiving healthcare who can benefit from improved access to higher quality of care, enabled by the system	Active actor
AI Service Operator	Role	Stakeholder responsible for the analysis of the AI services output and the collaboration with the healthcare professional	Active actor
AI Service Provider	Role	Stakeholder responsible for the services performing patient health data analysis	Active actor
Virtual Health Assistant	Resource	Virtual assistant application running on the edge device to provide preliminary diagnoses	Passive actor
Edge Device	Resource	IoT-enabled device for preliminary health data processing and communication with the CoGNETs middleware	Passive actor
Wearables	Resource	Smartwatches for continuous and real-time health monitoring	Passive actor
CoGNETs Middleware	Role	Distributed middleware framework that	Passive actor

Funded by the European Union
Project funded by

		provides dynamic cloud resource organization, optimal data processing and seamless service provisioning	

As described in the table above, **healthcare professionals** act as the connection between the patient's condition and the necessary care provided. In the PUC3 context, the healthcare professional collaborates with the system and is responsible for analysing and interpreting the medical insights generated by the system, if needed. On the other hand, **patients** play a more central role in PUC3, benefiting directly from the advancements and capabilities of the system. As the recipients of the proposed healthcare services, their main responsibility lies on providing consent for continuous data collection, actively participating in monitoring activities and engaging with the system through the virtual assistant to receive personalized care plans and insights, which can significantly improve their access to quality care.

The **AI service operator** and **AI service provider** are responsible for overseeing the analysis of the system's outputs, to ensure that the proposed system produces accurate insights based on the patients' data, and maintaining the proposed connected health service, respectively. Both of these actors work closely with one another, and if needed with the healthcare professionals, to ensure that the proposed AI algorithms are producing reliable insights to improve patient care.

Finally, all remaining actors, namely the **virtual health assistant**, the **edge device**, the **wearables** and the **CoGNETs middleware**, are all necessary systems and equipment that serve as passive actors, providing essential support or resources without directly engaging in decision-making or interaction with the other actors. While they do not engage in decision-making, their role is vital in ensuring that active actors can access timely and accurate information for making informed decisions and delivering care.

8.3.4 Requirements and assumptions

The CoGNETs PUC3 scenario focuses on combining IoT, Edge and Cloud computing to offer healthcare applications with advanced AI-driven solutions. In this context, PUC3 is highly dependent on the CoGNETs swarm infrastructure, serving as the central hub where initial insights and model specifications are transmitted for comprehensive processing, requiring high-performance computations, robust data security and seamless scalability.

The proposed swarm infrastructure must support advanced AI algorithms, particularly reinforced (RL) and deep learning (DNN) models, to analyse vast amounts of patient data with precision, in order to predict health trends and generate personalized insights. By utilizing higher computing power and improved AI models, the swarm ensures that patients receive actionable and accurate health diagnostics.

To better safeguard patient data against any potential cyber-attacks, such sensitive information will not be transmitted to the swarm infrastructure. Nonetheless, the CoGNETs swarm continuum is expected to incorporate state-of-the-art encryption, anomaly detection and access control to ensure that any essential information transmitted, including the AI algorithms' specifications and essential modelling parameters, are well protected. This functionality is key to maintaining trust in both hospital and home-care environments where secure data exchange is critical.

Funded by the European Union
Project funded by

Additionally, since the CoGNETs swarm continuum infrastructure will incorporate state-of-the-art gaming agents and biding functionality to adapt to varying computational demands, PUC3 re-lies on this functionality to provide real-time health monitoring and telehealth services that are based on continuous data streams from the wearables and IoT devices. For this reason, minimal latency and uninterrupted service delivery, especially in resource-constrained or remote environments, are required.

However, to ensure the successful implementation of the CoGNETs PUC3 connected healthcare scenario, certain functional and operational assumptions must be established, aligned with the scenario's objectives of delivering precise diagnostics, secure data handling and efficient telehealth services. For this purpose, the following assumptions are considered:

- **Stable home network** connectivity to enable continuous data transmission between the wearables and the Edge device. Any disruption in connectivity can potentially lead to delayed or incomplete data processing that could compromise the accuracy of diagnostics and patient care.
- **Charged** and fully **operational** health **devices** worn on the patient's wrist throughout the day. Patients must be informed of the importance of maintaining the device's readiness and usage compliance as part of their engagement with the system.
- Active and fully functional data **security and encryption** mechanisms to safeguard sensitive patient data against unauthorized access or cyberattacks. This assumption refers to both the Edge device and the CoGNETs middleware infrastructure.
- Informed **patient consent** is required for continuous data collection, monitoring and analysis. This is essential to ensure compliance with ethical guidelines and data privacy regulations, while building trust and engagement between the patients and the connected healthcare service.
- Effective use of the **virtual health assistant** application by the patients, including inter-actions with the application to receive personalized care plans, respond to health-related prompts and access diagnostic insights. In accordance, the virtual assistant application must feature an intuitive and user-friendly interface that minimizes the need for formal training or guidance.
- Sufficient processing **power and device storage** capacity to reduce latency and system downtime. This assumption refers to both the wearable devices, where an app stores data locally until the user's reconnection to the home network will be implemented, and the Edge device.

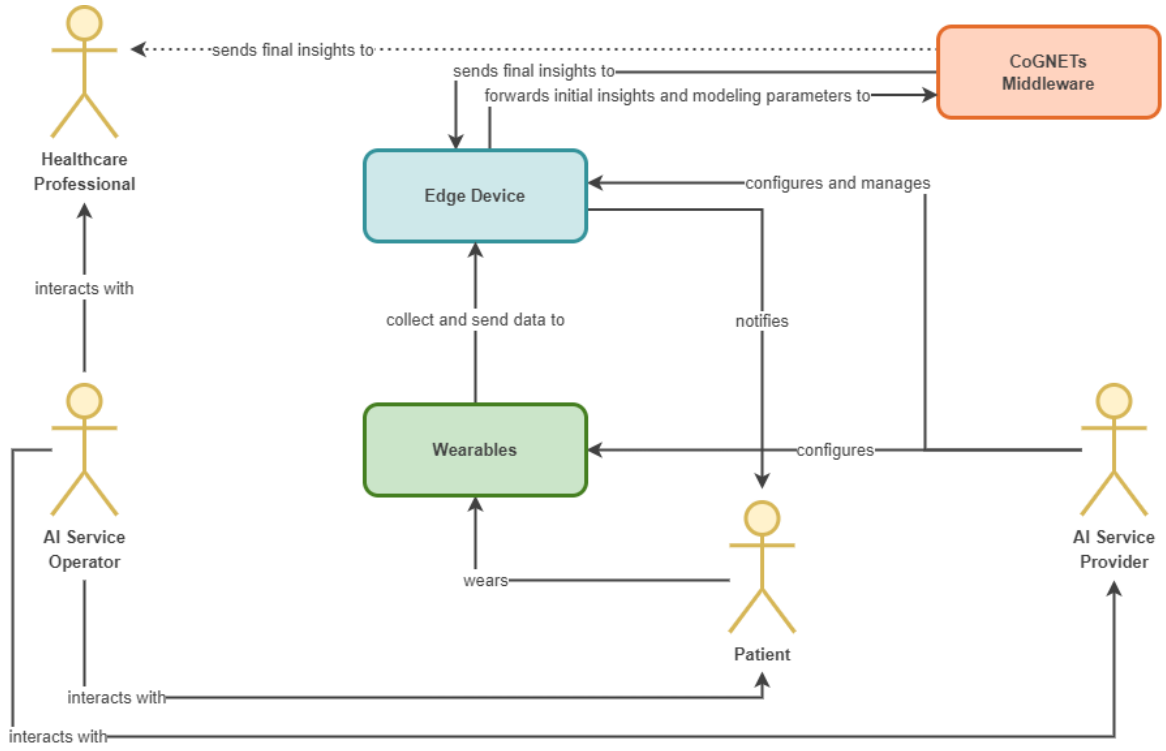
8.3.5 Workflows between actors

To further showcase the relationships between the identified actors from Table 41, a role model diagram was established, as depicted in Figure 41, which illustrates the key interactions between both active and passive actors.

Funded by the European Union

Project funded by

Figure 41: PUC3 role model diagram



As depicted in the diagram, the AI service operator interacts with both the healthcare professionals and the patients to gather initial insights and establish the end-user's needs, respectively, prior to the system's development. This accumulated knowledge is then passed on to the AI service provider who is responsible for configuring the wearable devices, as well as developing and managing the proposed healthcare service on the Edge device, ensuring that the system delivers accurate predictions and health insights, while the CoGNETs middleware per-forms cloud-assisted analytics and further processing. The wearables and edge device form the foundation of the IoT-Edge-Cloud continuum by ensuring continuous data flow, thereby al-lowing patients to benefit from near real-time monitoring and personalized care, while healthcare professionals rely on actionable insights for improved diagnostics and treatment.

8.3.6 Initial PUC State

At the initial state of the CoGNETs PUC3 scenario, healthcare infrastructure remains fragmented, with limited integration between IoT devices, edge computing, and cloud-based medical analytics. Traditional healthcare systems face several key challenges, including underutilization of patient data, a lack of near real-time AI-driven diagnostics, cybersecurity vulnerabilities, and accessibility barriers for remote healthcare services.

Currently, medical data is collected through medical wearables, hospital monitoring systems, and electronic health records, but it is often stored in isolated spaces, limiting its potential for predictive analytics and proactive patient care. Many hospitals and home healthcare settings rely on localized processing, making it difficult to derive comprehensive insights from large-scale patient data. As a result, early detection of diseases and personalized treatment plans remain suboptimal, reducing the overall efficiency of healthcare delivery.

Funded by the European Union
Project funded by

Security concerns further exacerbate these challenges, as IoT medical devices are frequently targeted by cyberattacks, creating risks for patient data integrity and system reliability. Existing security frameworks struggle to provide end-to-end protection, particularly in home-based healthcare environments where data exchanges occur over less secure networks.

In addition, telehealth services remain limited in their ability to provide near real-time, AI-assisted diagnostics due to connectivity issues, inadequate computational resources at the edge, and the absence of robust AI models that can operate efficiently in decentralized environments. Healthcare providers lack the necessary tools to integrate real-time patient monitoring with advanced AI insights, making remote diagnosis and treatment less effective.

The **initial state of PUC3** reflects these existing gaps, where:

- **Medical data is underutilized** due to a lack of integrated AI-driven analytics.
- **Security mechanisms are insufficient** to ensure safe and trustworthy diagnostics.
- **Telehealth services lack scalability and efficiency**, particularly in resource-constrained environments.
- **Healthcare professionals and patients have limited access to personalized insights**, reducing the effectiveness of preventative care and real-time monitoring.

These limitations highlight the need for CoGNETs' middleware framework, which aims to introduce scalable, AI-enhanced, and secure IoT-to-Cloud solutions to overcome these inefficiencies. The transition from this fragmented healthcare ecosystem to a CoGNETs-enabled, intelligent infrastructure represents a major leap toward predictive, data-driven, and secure healthcare services.

8.3.7 Expected outcomes

The key outcomes expected from this CoGNETs PUC3 scenario are focused on improving patient care, increasing system efficacy and ensuring sensitive data security. For this purpose, highly accurate health diagnostics, better overall patient care and data security and patient privacy are among the most significant expected outcomes of this PUC scenario, realized through advanced AI solutions and the implementation of proactive threat detection mechanisms. Other expected outcomes also include the provision of scalable and easily accessible telehealth services, cost efficiency, as well as treatment efficacy and patient satisfaction. In addition to that, from a cybersecurity perspective, CoGNETs will establish robust data protection and trust frameworks, ensuring that sensitive patient information is securely processed and stored. This will enhance confidence in digital healthcare solutions and accelerate their adoption. Finally, the initiative will contribute to sustainable and intelligent healthcare infrastructure, promoting energy-efficient computing while minimizing the environmental footprint of digital health technologies.

8.3.8 Specific facilities

The implementation of CoGNETs PUC3 is supported by dedicated infrastructure and real-world data sources that enable the development and validation of its healthcare security solutions. A key aspect of this effort is the use of nodes and IoT-enabled medical devices that were deployed for data collection, providing valuable insights into network activity within healthcare environments.

Funded by the European Union
Project funded by

HMU plays a central role in enhancing security for CoGNETs PUC3 by developing AI-powered anomaly detection mechanisms. HMU's infrastructure processes a real-world dataset of network flows collected from hospital infrastructures, that include IoT medical devices, which serves as the foundation for training machine learning models capable of identifying cyber threats, unauthorized access attempts, and abnormal network behaviours in healthcare networks. This dataset was previously collected and utilized within the SPHINX Project[2], ensuring a high level of reliability and relevance for security research.

By leveraging high-performance computing infrastructure, including a Dell R960 server, three Dell R640 servers, two Fujitsu Primergy TX150 S7 systems, and an NVIDIA A100 GPU, Pasiphae Lab is capable of running complex AI-based security algorithms in-house. This infrastructure enables scalable and privacy-preserving training and deployment of anomaly detection models, improving real-time security monitoring and response capabilities.

Using real-world network traffic data and high-performance AI models, the lab contributes to several key areas:

- **Advanced Threat Detection**: AI-driven models trained on real-world anonymised network flows enhance the identification of cyber threats in hospital and home-care networks, enabling proactive security measures.
- **Real-Time Security Monitoring**: AI-based anomaly detection is seamlessly integrated into CoGNETs' infrastructure, enabling continuous monitoring of potential threats.
- **Scalable and Adaptive Protection**: Intelligent security mechanisms dynamically adapt to emerging cybersecurity risks while maintaining low computational overhead, ensuring efficiency across different healthcare environments.

By combining AI-driven security, real-world datasets, and high-performance computing, CoGNETs PUC3 establishes a resilient, scalable, and intelligent cybersecurity framework, safeguarding healthcare operations, patient data, and medical infrastructures.

8.3.9 Production needs

The established production needs for the proposed PUC3 connected healthcare scenario are described in the following Table.

Table 102: PUC3 production needs description.

ID	Name	Description
PUC3.PN.001	Device Scalability	IoT-enabled wearables and edge devices for real-time health monitoring and data collection, equipped with sensors and communication modules for reliable data transmission in diverse healthcare settings.
PUC3.PN.002	Edge Infrastructure	Compact and powerful computing units for preliminary data processing and running lightweight AI models, with secure communication to the

[2] https://sphinx-project.eu/

Funded by the European Union
Project funded by

		cloud and compliance with data privacy regulations.
PUC3.PN.003	High-Performance Cloud Computing	High-performance cloud computing systems for advanced AI model execution, complex analytics and CFL to ensure secure data processing without transferring raw data from edge devices.
PUC3.PN.004	Software/Hardware Security	Robust encryption, anomaly detection and secure access control mechanisms to protect sensitive healthcare data and maintain system integrity across IoT-Edge-Cloud infrastructures.
PUC3.PN.005	Software Optimization	Development of AI models optimized for edge and cloud deployment, middleware for seamless data flow across components and user-friendly interfaces for healthcare professionals and patients.
PUC3.PN.006	Interoperability Standards	Compatibility across the computing continuum to adapt to diverse healthcare workflows and regulatory environments, supporting scalable and reliable system deployment.

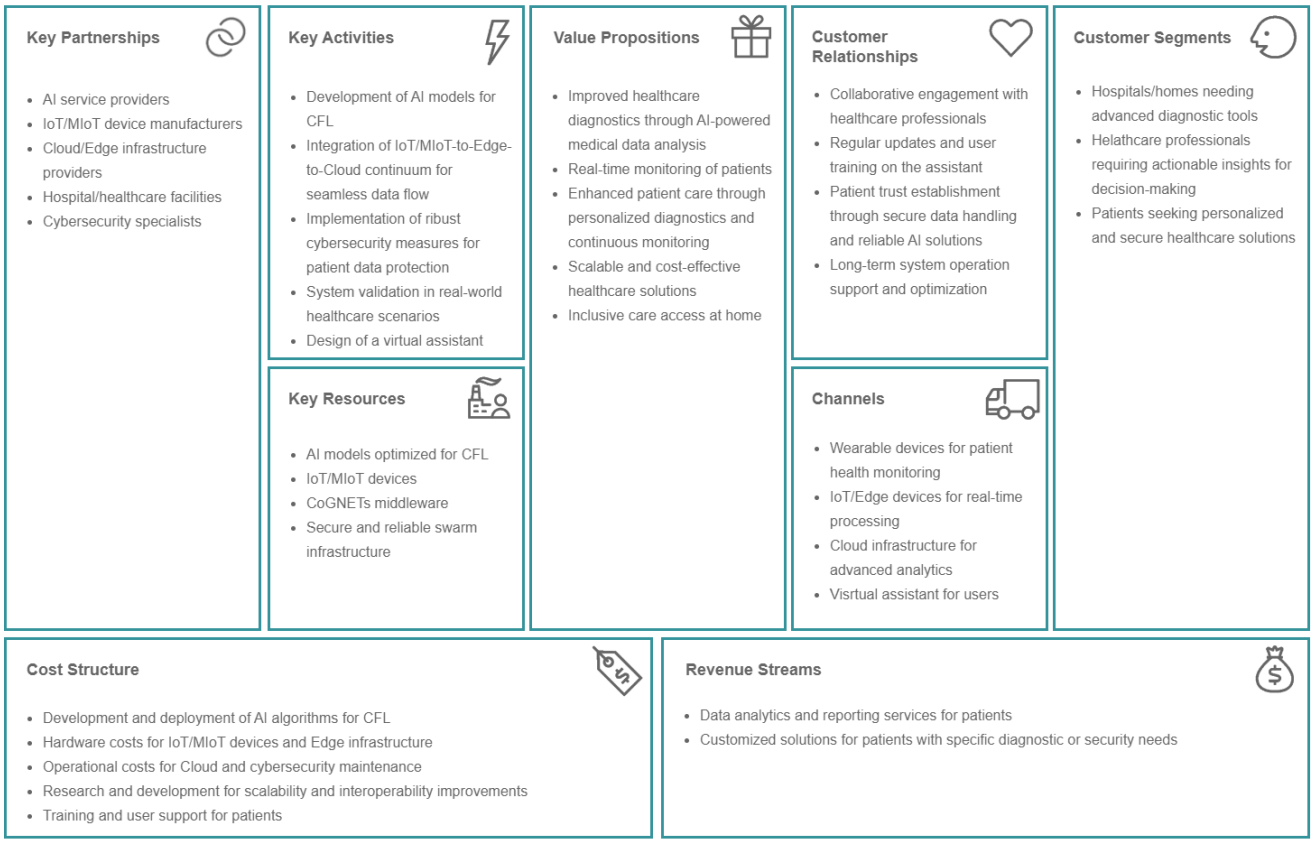
8.3.10 Business model

To develop and describe the business model for the PUC3 connected healthcare scenario, the business model canvas[3] approach was selected thanks to its versatility, simplicity and effectiveness in capturing the essential elements of the use case scenario. As seen in Figure 8, the canvas provides a structured framework that allows the visualization and organization of all the key components of PUC3, offering a more comprehensive and coherent business model that can be easily communicated to all CoGNETs stakeholders.

[3] https://www.strategyzer.com/library/the-business-model-canvas

Funded by the European Union
Project funded by

Figure 42: Business model canvas for PUC3



8.3.11 KPIs and performance thresholds

For the proposed PUC3 scenario, six KPIs have been proposed, described in the following Table.

Table 103: PUC3 KPIs description

ID	Name	Description	Reference to PUC objectives
KPI_3.1	Deployment of CoGNETs RL/Deep-RL Algorithms	Successfully deploy CoGNETs RL/Deep-RL versions of CNN/ DNN algorithms for CFL learning on medical devices	(1), (2)
KPI_3.2	Cost Savings	Achieve ≥ 25% cost savings through the implementation of di-gital and interconnected health-care technologies	(1), (3)

Funded by the European Union
Project funded by

KPI_3.3	Treatment Efficacy and Patient Satisfaction	Achieve ≥ 25% improvement in treatment efficacy and patient satisfaction (measured via questionnaires)	(1), (3)
KPI_3.4	Security Threat Detection	Detect ≥ 95% of security threats, ensuring the safety and reliability of medical data and systems	(2)
KPI_3.5	Data Privacy Compliance	Ensure 100% compliance with GDPR and other healthcare data privacy regulations.	(2)
KPI_3.6	Energy Efficiency	Reduce energy consumption of IoT devices and edge computing processes by ≥ 20% compared to baseline systems.	(1)

KPI_3.1 ("Deployment of CoGNETs RL/Deep-RL Algorithms") is proposed for the evaluation of the successful deployment of CoGNETs RL/Deep-RL versions of CNN/DNN algorithms on medical devices for CFL, by measuring the ability of these algorithms to enhance the performance of edge devices. By integrating AI-powered decision-making processes into healthcare applications, this KPI demonstrates how advanced learning algorithms can optimize resource usage, improve real-time responsiveness and support the adoption of AI in healthcare systems, while maintaining decentralized data privacy.

KPI_3.2 ("Cost Savings") is related to the evaluation of the financial efficiency of implementing digital and interconnected healthcare technologies within CoGNETs, by measuring the percentage of cost savings achieved, targeting a minimum of 25%, compared to traditional healthcare systems. By assessing operational costs, resource utilization and the economic impact of automating routine tasks and enabling remote care, this KPI highlights the cost-effectiveness of adopting digital technologies, while maintaining or improving healthcare de-livery standards.

KPI_3.3 ("Treatment Efficacy and Patient Satisfaction") is related to assessing the improvements in treatment efficacy and patient satisfaction resulting from the integration of CoGNETs solutions in healthcare workflows, quantifying a target of at least 25% improvement, measured through patient and healthcare provider questionnaires. The objective of this KPI is to ensure that the adoption of AI-powered healthcare solutions directly translates into better health outcomes and increased patient confidence in the care they receive.

KPI_3.4 ("Security Threat Detection") is focused on evaluating the capability of the CoGNETs project to identify and mitigate security threats to healthcare data and systems, by measuring the percentage of detected threats (successful detection ≥ 95%). Through assessing the robustness of encryption, anomaly detection and other cybersecurity measures integrated into medical devices and networks, this KPI ensures that healthcare systems remain resilient against cyberattacks, thereby safeguarding sensitive medical information and maintaining patient trust in digital healthcare solutions.

Funded by the European Union
Project funded by

KPI_3.5 ("Data Privacy Compliance") is focused on ensuring that the CoGNETs project adheres to GDPR privacy regulations regarding the patients' sensitive information by evaluating the effectiveness of data encryption, anonymization and secure storage techniques implemented across IoT devices, edge systems and the swarm infrastructure. The objective of this KPI is to maintain patient trust, safeguard sensitive medical data and avoid potential legal and reputational risks, thereby supporting the long-term viability of the PUC3 scenario.

KPI_3.6 ("Energy Efficiency") is proposed for assessing the reduction in energy consumption achieved by PUC3, with a target of ≥ 20% compared to baseline systems. By evaluating the energy efficiency of IoT devices and edge computing processes, this KPI optimizes resource usage and lowers energy demands, thereby addressing environmental concerns, reducing operational costs and contributing to a more sustainable and economically viable healthcare ecosystem.

8.3.12 Guidelines to validate the KPIs

To ensure the successful implementation and evaluation of the CoGNETs PUC3 scenario, the following validation guidelines are established for each KPI. These guidelines define the method tools, and assessment criteria necessary to measure performance effectively.

- **KPI_3.1 ("Deployment of CoGNETs RL/Deep-RL Algorithms"):**
  - **Validation Approach**:
    - **Deployment Testing**: Ensure that RL/Deep-RL versions of CNN/DNN algorithms are successfully deployed on edge and medical devices used in the PUC3 scenario.
    - **Computational Performance Assessment**: Measure processing latency, model inference time, and computational overhead introduced by AI algorithms compared to baseline non-AI healthcare solutions.
    - **Accuracy and Adaptability**: Evaluate model performance in detecting health patterns, predicting trends, and making decisions based on real-world patient data.
    - **Decentralized Learning Validation**: Assess the effectiveness of CFL by monitoring model updates and data privacy preservation.
  - **Success Criteria**:
    - **RL/Deep-RL algorithms** operate efficiently within the computational constraints of medical edge devices.
    - **Performance improvements** in resource utilization and real-time decision-making are observed.
    - **AI models** correctly interpret patient data with an acceptable accuracy threshold.
    - **CFL mechanisms** function effectively without compromising privacy or requiring centralized data storage.
- **KPI_3.2 ("Cost Savings"):**
  - **Validation Approach**:
    - **Cost-Benefit Analysis**: Compare the operational costs of the CoGNETs PUC3 healthcare system against conventional healthcare solutions.

Funded by the European Union
Project funded by

- **Breakdown of Cost Savings**: Analyse reductions in expenses related to hospital visits, medical staff workload, infrastructure maintenance, and energy consumption.
- **Automation and Efficiency Gains**: Evaluate the economic impact of automating routine medical procedures, remote patient monitoring, and digital healthcare services.
- **Return on Investment (ROI) Analysis**: Assess financial benefits relative to initial deployment and operational costs.

- **Success Criteria**:
  - A minimum of 25% **cost reduction** compared to traditional healthcare systems is achieved.
  - Digital healthcare technologies demonstrate **operational efficiency** without compromising quality.
  - The system supports **scalable healthcare services** with lower resource utilization.

- **KPI_3.3 ("Treatment Efficacy and Patient Satisfaction"):**
  - **Validation Approach**:
    - **Patient Outcome Evaluation**: Measure treatment effectiveness based on clinical indicators and recovery rates.
    - **Patient and Healthcare Provider Surveys**: Conduct structured questionnaires to assess satisfaction, ease of use, and perceived improvements in healthcare services.
    - **Comparative Study**: Analyse historical patient data before and after CoGNETs PUC3 implementation to quantify improvements in treatment efficacy.
    - **AI-Driven Diagnostics Impact**: Evaluate how AI-assisted diagnostics contribute to better decision-making and personalized care plans.
  - **Success Criteria**:
    - A minimum of 25% **improvement in treatment outcomes and patient satisfaction** is reported.
    - **Positive feedback** from healthcare providers on system usability and effectiveness is recorded.
    - AI-powered insights lead to measurable **enhancements in diagnostic accuracy and treatment planning**.
- **KPI_3.4 ("Security Threat Detection"):**
  - **Validation Approach:**
    - **Security Testing and Penetration Testing**: Conduct simulated cyberattacks on the PUC3 infrastructure to assess the robustness of security mechanisms.
    - **Anomaly Detection Performance**: Validate the effectiveness of AI-driven anomaly detection systems in identifying potential threats.
    - **Real-Time Security Monitoring**: Evaluate the system's ability to detect and mitigate security breaches in live healthcare environments.

Funded by the European Union
Project funded by

- **Incident Reporting and Response Time**: Measure the time required to identify and respond to cybersecurity threats.

- **Success Criteria:**
  - A detection rate of ≥ 95% for **security threats** is achieved.
  - The system demonstrates **effective encryption, anomaly detection, and access control mechanisms**.
  - **Security incidents are logged, analysed and mitigated** within acceptable response times.

- **KPI_3.5 ("Data Privacy Compliance"):**
  - **Validation Approach:**
    - **Regulatory Compliance Audit**: Verify that the PUC3 system adheres to GDPR and other relevant healthcare data protection regulations.
    - **Data Encryption and Anonymization Assessment**: Evaluate the implementation of encryption protocols and anonymization techniques across IoT and cloud environments.
    - **Access Control and Authentication Testing**: Ensure role-based access control (RBAC) mechanisms effectively restrict unauthorized data access.
    - **Data Retention and Deletion Policy Review**: Confirm that patient data is stored and erased according to legal and ethical standards.
  - **Success Criteria:**
    - 100% compliance with **GDPR and healthcare data privacy regulations** is maintained.
    - **No data breaches or unauthorized access incidents** occur within the validated timeframe.
    - **Secure communication protocols** effectively protect sensitive patient information.
- **KPI_3.6 ("Energy Efficiency"):**
  - **Validation Approach:**
    - **Energy Consumption Benchmarking**: Measure the power usage of IoT devices, edge computing units, and cloud infrastructure before and after CoGNETs integration.
    - **Optimization Analysis**: Evaluate the impact of AI-driven optimizations on energy efficiency.
    - **Comparative Study**: Assess reductions in energy consumption relative to conventional healthcare monitoring systems.
    - **Environmental Impact Assessment**: Analyse how improved energy efficiency contributes to sustainability goals.
  - **Success Criteria:**
    - A r**eduction of ≥ 20% in energy consumption** compared to baseline healthcare solutions is achieved.

Funded by the European Union
Project funded by

- AI-driven optimizations demonstrate measurable **improvements in power usage efficiency**.
- The system aligns with **sustainable computing and environmental impact reduction initiatives**.

8.3.13 Data Models

The PUC3 healthcare data model consists of 15 main classes, 25 object properties and 3 data properties, which are analysed below. Of the 15 classes defined for the system ontology, 10 classes were imported from the SAREF4health ontology[4], which is based on the Semantic Annotation for Real-world Entities Framework (SAREF) and is specifically designed for the healthcare domain. Using the SAREF4health ontology as a base, additional properties and units of measure were also included, catered specifically to the needs of the PUC3 connected healthcare scenario. The remaining 5 classes (GeolocationFunction, GeolocationProperty, EdgeDevice, Event and VirtualHealthAssistant) were created based on the system's design requirements.

Specifically, the main classes defined for the PUC3 scenario, as well as their relationships, are presented and described as follows:

- **foaf:Agent**: class representing individuals in the scenario. For the needs of the PUC3 healthcare scenario, two agents (hasRoleOf) were defined, the caregiver and the patient. For each patient, their medical history is also recorded (hasHistory).
- **s4health:ChronicDisease**: class associated with patient medical history. For the purposes of this scenario, three core diseases were defined, namely asthma, diabetes and chronic obstructive pulmonary disease, which are continuously monitored through systematic measurements from the wearable devices.
- **s4health:Device**: class describing health devices within the system. Since the PUC3 scenario is focused on healthcare applications, only the “HealthDevice” subclass and the “HealthWearable” device type were used. Each device in the scenario has a key functionality (hasFunction) of collecting health measurements (measures) and the patient’s geolocation, calculating their location and matching the collected measurements to any pre-existing chronic diseases (monitorsPropertiesRelatedTo).
- **s4health:FeatureOfInterest**: class representing any real-world entity from which a property is measured. It includes two properties, a direct connection to the health measurements (hasMeasurement), as well as an inferred property from the “Property” class (inferred object property isFeatureOfInterestOf).
- **s4health:Function**: class describing the functionality of each device within the system. Since PUC3 has a key objective of employing wearable devices for health data collection from patients, the subclasses used were the “Measurement Function” and “Actuating Function” subclasses. The actuating function specifically targets stress level measurements (initiatesProperty/actsOnProperty), which are not automatically measured by the device.
- **s4health:Location**: class related to the patient’s precise location, defined by geometric coordinates and associated timestamp.

[4] https://saref.etsi.org/saref4ehaw/v1.1.1

Funded by the European Union
Project funded by

- **s4health:Measurement**: class collecting measurement values from the PUC3 scenario devices. Each collected value has a specific unit of measurement (isMeasuredIn) and is related to a specific property (relatesToProperty). For measurements related to chronic diseases, lower threshold values are systematically checked (is linked to event thresholds) to prevent potential adverse events.

- **s4health:Property**: describes an entity's property associated with wearable device measurements (inferred object property isMeasuredBy). The collected values include a feature of interest (hasFeatureOfInterest), related to the measurements, and additional information about the originating device (inferred object property isMeasuredBy).

- **s4health:Task**: class defining the device's purpose, triggered (inferred object property isTriggeredBy) by potential events derived from the Edge device model's decisions.

- **s4health:UnitOfMeasure**: class representing the specific measurement units used in data collection.

- **EdgeDevice**: class related to the primary IoT device in the PUC3 scenario, containing AI models for generating continuous user insights. It includes properties related to data collection (inferred object property obtainsDataBy), as well as process insights (sendsInsightsTo) showcased to the patient via the virtual health assistant application.

- **Event**: class corresponding to the AI prediction models for specific medical events, including tachycardia, bradycardia, low oxygen saturation, increased respiratory rate and low active minutes.

- **GeolocationFunction**: class responsible for calculating the patient's location using geometric coordinates from the wearable device.

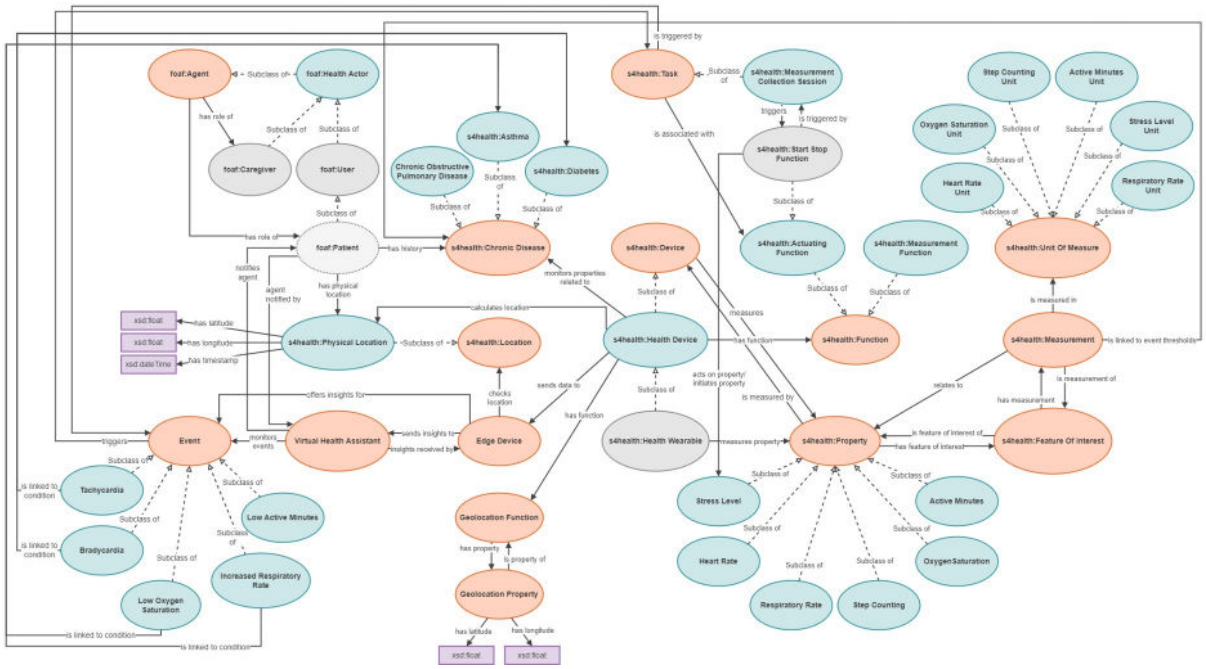
- **GeolocationProperty**: class representing the patient's location in geometric coordinates, automatically calculated by the wearable device.

- **VirtualHealthAssistant**: class related to the proposed system's UI, responsible for monitoring events (monitorsEvents), displaying the results of the prediction models (inferred object property insightsReceivedBy) and providing insights (notifiesAgent) to the patient or the caregiver.

A comprehensive representation of the PUC3 healthcare scenario data model is depicted in Figure 43, where the dashed lines indicate subclass associations. The main classes of the system are represented in blue, their subclasses in red, and their respective subclasses are depicted in light grey. The "Patient" subclass is represented as a dashed entity, as it is a subclass of the "User" subclass, which is one level down.

Funded by the European Union
Project funded by

Figure 43: PUC3 data model based on the SAREF4health ontology



Finally, the following tables provide a comprehensive overview of the properties required for the PUC3 healthcare scenario. Specifically, Table 104 lists the object properties used in the data model schema, presented above.

Table 104: PUC3 object properties

Object Properties			
Name	**Domain**	**Range**	**Inverse of**
actsOnProp-erty	StartStopFunction	StressLevel	isActedOnBy
calculatesLo-cation	HealthDevice	PhysicalLocation	-
checksLoca-tion	EdgeDevice	Location	-
hasFea-tureOfInterest	Property	FeatureOfInterest	isFeatureOfInterestOf
hasFunction	HealthDevice	Function GeolocationFunction	-
hasHistory	foaf:Patient	ChronicDisease	-

Funded by the European Union

Project funded by

hasMeasure-ment	FeatureOfInterest	Measurement	isMeasurementOf
hasPhysical-Location	foaf:Patient	PhysicalLocation	-
hasProperty	GeolocationFunction	GeolocationProperty	isPropertyOf
hasRoleOf	foaf:Agent	foaf:Patient foaf:Caregiver	-
initiatesProp-erty	StartStopFunction	StressLevel	propertyInitiatedBy
isAssociated-With	Task	ActuatingFunction	-
isLinkedTo-Event-Thresholds	Measurement	ChronicDisease	-
isLinkedTo-Condition	IncreasedRespiratoryRate LowOxygenSaturation Bradycardia Tachycardia	Asthma Asthma Diabetes Diabetes	-
measures	Device	Property	isMeasuredBy
isMeasuredIn	Measurement	UnitOfMeasure	-
measure-sProperty	HealthWearable	HeartRate StepCounting OxygenSaturation RespiratoryRate ActiveMinutes StressLevel	-
monitorsEv-ents	VirtualHealthAssistant	Event	-
monitorsProp-erties-RelatedTo	HealthDevice	ChronicDisease	-

Funded by the European Union
Project funded by

notifiesAgent	VirtualHealthAssistant	foaf:Patient	agentNotifiedBy
offersInsights-For	EdgeDevice	Event	-
relatesToProp-erty	Measurement	Property	-
sendsDataTo	HealthDevice	EdgeDevice	obtainsDataBy
sendsInsight-sTo	EdgeDevice	VistualHealthAssistant	insightsReceivedBy
triggers	MeasurementCollectionSession Event	StartStopFunction Task	isTriggeredBy

Similarly, Table 105 lists the data properties of the PUC3 information model. Specifically, there are three properties: “hasLatitude” and “hasLongitude”, which are associated with the patient's geolocation and physical location, respectively; and “hasTimestamp”, which records the time at which the user's location was acquired, and is of type dateTime.

Table 105: PUC3 data properties

Data Properties		
Name	**Domain**	**Range**
hasLatitude	GeolocationProperty PhysicalLocation	xsd:float
hasLongitude	GeolocationProperty PhysicalLocation	xsd:float
hasTimestamp	PhysicalLocation	xsd:dateTime

8.3.14 End-user Service Components

The CoGNETs PUC3 scenario integrates multiple service components designed to enhance healthcare delivery through AI-driven decision-making, real-time patient monitoring, and seamless communication between different actors in the system. These components are classified based on their role in the system and their interaction with end-users, ensuring efficient data processing, privacy compliance, and intelligent healthcare support.

1) **Healthcare Professional Interface:**

a) **Description**: A secure, user-friendly, and responsive dashboard accessible by doctors, nurses, and medical staff to monitor patient conditions in real time.

Funded by the European Union
Project funded by

b) **Key Features**:

- Visual representation of patient health metrics.
- AI-assisted decision support for diagnostics.
- Secure communication with AI services and medical devices.
- Alerts and recommendations for critical patient conditions.
- Provides feedback to the Virtual Health Assistant

c) **Role in the System**: Enables healthcare professionals to leverage AI insights for enhanced patient care.

2) Patient Interface:

a) **Description**: A secure, user-friendly, and responsive dashboard that allow patients to access their health data, receive personalized recommendations, and communicate with the Virtual Health Assistant.

b) **Key Features**:

- Near real-time health status updates.
- AI-driven insights for proactive healthcare management

c) **Role in the System**: Empowers patients to actively engage in their own healthcare management.

3) Virtual Health Assistant:

a) **Description**: An AI-powered assistant with the purpose of assisting patients with preliminary diagnoses and health monitoring.

b) **Key Features**:

- Integration with wearable health devices
- Dynamic response taking into consideration healthcare professionals' feedback
- Symptom Analysis

c) **Role in the System**: Provides automated and tailored healthcare assistance based on AI-driven insights

4) AI Service Operator Platform:

a) **Description**: A backend service for AI analysts and data experts to monitor and refine AI models deployed in the healthcare system.

b) **Key Features**:

- Visualization and assessment of AI-generated insights.
- Tools for collaboration with medical professionals.
- Performance monitoring and validation of AI decision-making accuracy.

Funded by the European Union
Project funded by

c) **Role in the System**: Ensures AI service outputs align with healthcare standards and patient safety.

5) AI Service Provider Engine:

a) **Description**: The core computational system responsible for executing AI models, processing patient data, and delivering actionable insights.

b) **Key Features**:

- Federated learning for privacy-preserving AI training.
- Data preprocessing and anomaly detection.
- Secure integration with the CoGNETs middleware.
- Explainable AI for system's insights in decision making.

c) **Role in the System**: Delivers AI-powered diagnostics and predictive analytics to support medical professionals.

6) Edge Devices:

a) **Description**: IoT-enabled devices deployed at healthcare facilities and patient homes to facilitate real-time data collection and processing.

b) **Key Features**:

- Localized health data analysis for reduced cloud dependency.
- Secure data transmission to the CoGNETs middleware.
- Support for AI-driven health monitoring applications.

c) **Role in the System**: Enhances real-time healthcare services by processing patient data at the edge.

7) Wearables:

a) **Description**: Smartwatches and biometric sensors that continuously track patient health metrics.

b) **Key Features**:

- Real-time monitoring of heart rate, oxygen levels, and activity.
- Secure transmission of health data to edge devices and the cloud.
- AI-assisted alerts for potential health risks.

c) **Role in the System**: Supports continuous patient monitoring for proactive healthcare.

8) CoGNETs Middleware:

a) **Description**: A distributed framework that dynamically organizes cloud resources, optimizes data processing, and enables seamless integration of AI services.

b) **Key Features**:

Funded by the European Union
Project funded by

- Secure data routing between healthcare professionals, AI services, and patients.
- Scalable infrastructure for AI model execution and real-time analytics.
- Compliance with data privacy regulations (e.g., GDPR).

c) **Role in the System**: Provides an intelligent, secure, and efficient backbone for healthcare data management.

8.3.15 Risk Assessment & Mitigation Plan

The implementation of the CoGNETs PUC3 connected healthcare scenario comes with several potential risks that could affect its seamless functionality but can, however, be mitigated through strategic planning and the integration of robust technical measures.

One of the main risks is related to the loss of network connectivity when the patient leaves their home, which can cause data transmission inaccuracies or data loss, resulting in gaps to monitoring. To mitigate this issue, an application to the patient's wearable device will be integrated, so as to store the collected health data locally, until the patient returns home and re-connects to the local network. This will ensure that no critical data are lost and that continuity in patient monitoring is maintained.

Another concern lies on the reliance on the wearable devices being turned on, charged and worn on the patients' wrist throughout the day. To address any battery-related issues, patients should be instructed to charge their wearables while they are sleeping, ensuring that the de-vice is fully operational during the day. Furthermore, to address other potential user-related is-sues, like the possibility of patients forgetting to wear the device or using it incorrectly, the proposed PUC3 system will send periodic alerts or notifications to remind patients to wear the device.

Finally, potential hardware or software malfunction in the wearables, edge devices or cloud infrastructure is another risk that should be noted. To avoid such risks, regular maintenance and backups will be implemented to ensure continuity of service in the event of device failures or software glitches.

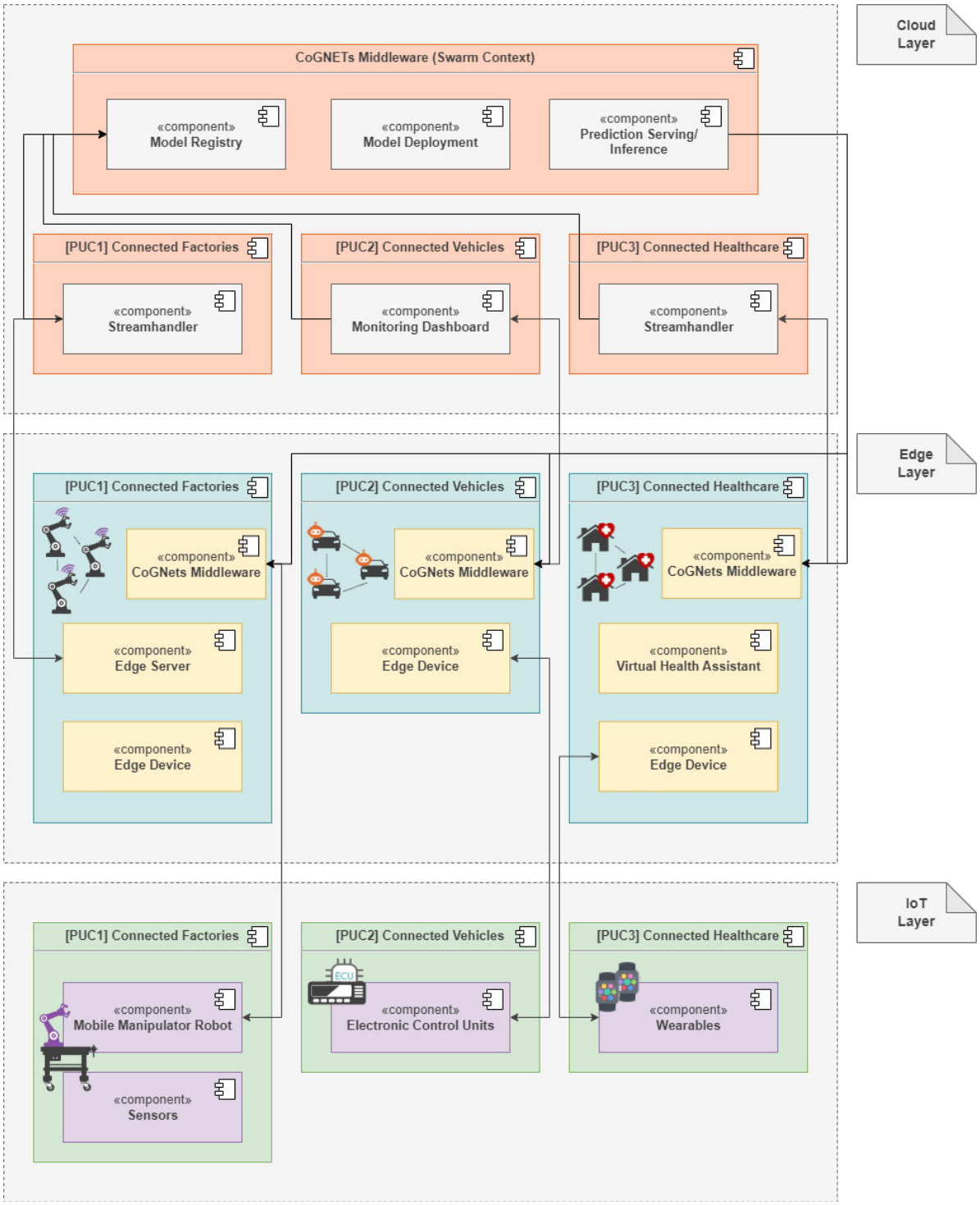
8.4 2ND STAGE - CROSS-VERTICAL SUPPLY CHAIN

CoGNETs' cross-vertical PUC – "Connected Verticals: Cross-Vertical Supply Chain (Universal Adaptability)" scenario is aimed to address key infrastructure optimization challenges across traditionally siloed industry domains. By creating an interconnected framework spanning industry (PUC1 – "Connected Factories"), mobility (PUC2 – "Connected Vehicles") and healthcare (PUC3 – "Connected Healthcare"), this approach enables dynamic resource sharing, collaborative security and enhanced operational efficiency, handling the underutilization of computing resources during off-peak hours, while simultaneously strengthening cybersecurity through cross-vertical threat intelligence sharing.

The following Figure 44 illustrates the architecture overview of the aforementioned cross-vertical PUC, depicted as a comprehensive three-layer approach, based on CoGNETs' IoT-Edge-Cloud continuum design. As seen in the figure, the CoGNETs middleware has a central role in the architecture, coordinating each domain-specific component across both the Edge and Cloud layers, demonstrating how similar technological components can be specialized across the three different domains while maintaining a consistent structural framework.

Funded by the European Union
Project funded by

Figure 44: Cross-vertical supply chain PUC architecture overview



Funded by the European Union

Project funded by

8.4.1 Objective

The key objective of the proposed cross-vertical use case scenario is to demonstrate how CoGNETs can effectively enhance interoperability, scalability, security and sustainability across different industrial sectors, by using AI-driven IoT-Edge-Cloud capabilities. As this second-stage PUC focuses on optimizing data handling, ensuring secure collaboration and fostering environmental sustainability among the three aforementioned vertical use cases, PUC1 – "Connected Factories", PUC2 – "Connected Vehicles", and PUC3 – "Connected Healthcare", the following objectives have been identified:

1) **Cross-vertical operationalisation for AI-driven data correlation**: By processing large volumes of AI-driven data from diverse sources, including industrial sensors, vehicle telematics and healthcare wearables, this PUC can enable industries to extract meaningful insights and optimize operations, ultimately leading to better decision-making and enhanced operational efficiency across different verticals, reducing data silos and maximizing AI potential. This is particularly important for facilitating seamless data integration and cross-correlation across all three vertical PUC scenarios.
2) **Cross-infrastructure scalability for dynamic resource allocation**: Through the dynamic allocation of computing resources across factories, vehicles and healthcare environments, this second stage PUC scenario ensures seamless service delivery, ensuring efficient resource distribution and minimal latency. As a result, a scalable and adaptive infrastructure that meets fluctuating demands is feasible.
3) **Cross-infrastructure accessibility for secure data sharing**: Through the development of access control mechanisms and secure data-sharing protocols to prevent unauthorized data leakage, this cross-vertical PUC ensures that sensitive data remain secure within organizational premises, while still enabling valuable cross-sector collaboration, ultimately ensuring a higher level of security and trust among users.
4) **Cross-infrastructure security for proactive cyberattack mitigation**: By integrating anomaly detection systems and decentralized access verification, enhanced security can be established across all three different industrial sectors, ensuring more robust operational efficiency. AI-powered threat detection helps in this notion, offering countermeasures even to the most sophisticated cyber-attacks.
5) **Cross-infrastructure optimization for AI-powered supply chain efficiency**: By exploring how AI-driven analytics can optimize logistics within CoGNETs, as well as predict potential bottlenecks and enhance real-time decision-making, the cross-vertical PUC is able to identify and resolve potential inefficiencies in the supply chain, ultimately contributing to improved supply chain performance.
6) **Cross-infrastructure greenification for sustainable AI operations**: Through collaborative AI efforts, energy usage optimization in manufacturing plants, efficiency enhancement of vehicle fleets and waste minimization in healthcare logistics can be realized, thereby reducing greenhouse gas (GHG) and carbon dioxide emissions and promoting an eco-friendly approach to industrial operations.

8.4.2 Why is it relevant for CoGNETs?

The cross-vertical PUC scenario is highly relevant to the CoGNETs platform as it demonstrates its ability in enabling seamless AI-driven collaboration across three key industrial sectors, manufacturing, mobility and healthcare, by providing a unified middleware solution that

Funded by the European Union
Project funded by

facilitates real-time data exchange, secure insights and optimized resource allocation.

One of the key contributions of the cross-vertical supply chain PUC scenario is to showcase CoGNETs' ability in enabling cross-correlation of AI models, thereby offering a holistic approach to operational intelligence that improves efficiency and reduces downtime across different industrial sectors. In addition, its security and accessibility features that allow for data to remain protected while shared though the network, ensure compliance with data privacy regulations, while also allowing for secure collaborations. Finally, the use case's commitment to sustainability through advanced AI optimizations and efficient Edge computing, leads to a significant reduction in unnecessary resource usage, contributing to greener industrial operations.

8.4.3 Requirements and assumptions

The CoGNETs cross-vertical PUC scenario aims to showcase the seamless integration of IoT-Edge-Cloud continuum across multiple sectors, including manufacturing, mobility and healthcare, with CoGNETs' swarm infrastructure allowing for cross-industry collaboration and facilitating real-time data exchange, AI-powered decision-making and secure, scalable operations.

The proposed CoGNETs swarm continuum must be capable of processing large-scale, multi-source data streams from factories, vehicles and healthcare systems, employing advanced AI techniques, such as DNNs, RL and CFL, to extract meaningful insights while ensuring data privacy. By utilizing AI models that continuously learn from cross-sector interactions, the system can enhance operational efficiency, detect anomalies and optimize supply chain processes across all verticals.

Given the sensitivity of industrial, vehicular and medical data, robust security measures and system decentralization are imperative to allow for sensitive data from each sector remain within their respective organizations, complying with privacy regulations. Furthermore, CoGNETs will implement state-of-the-art encryption, access control and anomaly detection to safeguard AI models, metadata and essential processing parameters exchanged across verticals.

Finally, since CoGNETs' swarm infrastructure must also support dynamic computational resource allocation to address varying industry-specific workloads, the proposed middleware must provide real-time scalability and minimal latency to ensure uninterrupted service delivery.

To ensure the successful implementation of the CoGNETs cross-vertical supply chain PUC scenario, certain functional and operational assumptions must be established, aligned with the scenario's objectives. For this purpose, the following assumptions are considered:

1. Reliable network connectivity across all industries to facilitate continuous data exchange between IoT devices, Edge devices and the Cloud infrastructure. Any disruption in connectivity can potentially lead to data loss, delayed processing or misaligned cross-sector AI operations.
2. Strict data privacy and security compliance across all verticals, ensuring that sensitive information remains within organizational premises, while allowing AI models to securely access cross-sector insights.
3. Interoperable AI and data-sharing standards across all participating sectors, ensuring seamless integration of heterogeneous data sources without compromising efficiency.
4. Sufficient computing power and storage capacity across both the cloud and the swarm context to handle large-scale processing without bottlenecks. This includes real-time inference on Edge devices for latency-sensitive applications and high-performance Cloud processing for more extensive AI training and optimization tasks.

Funded by the European Union
Project funded by

8.4.4 Expected outcomes

The key outcomes expected from this CoGNETs cross-vertical PUC scenario are focused on enhancing interoperability, optimizing operations, improving cybersecurity and promoting sustainability across multiple industries. For this purpose, real-time data-driven decision-making, cross-vertical collaboration, as well as secure and efficient information exchange are among the most significant expected outcomes for this supply chain scenario, realized through CoGNETs' computing continuum architecture. Other expected outcomes also include the increase in system resilience against adversarial attacks through advanced AI security mechanisms, robust encryption, access control and proactive anomaly detection. Finally, this initiative is also expected to promote green AI and sustainability by optimizing resource utilization and reducing unnecessary computational overhead, thereby contributing to a more sustainable digital ecosystem.

8.4.5 KPIs and performance thresholds

For the proposed cross-vertical supply chain PUC scenario, seven KPIs have been identified, as described in the following Table 106.

Table 106: Cross-vertical PUC KPIs description

ID	Name	Description	Reference to PUC objectives
KPI_CV1	Common Computing Space	Successfully establish a common computing space between PUC sites via Global versions of the decentralized Game	(1), (2), (5)
KPI_CV2	Supply Chain Risk Correlation	Achieve 100% correlation of all supply-chain risk events and threat intelligence across different PUC sites	(1), (3), (4)
KPI_CV3	Synchronized AI Co-processing	Ensure 100% synchronized co-processing between PUC sites for > 2 AI service models trained in parallel	(1), (3), (4)
KPI_CV4	Cross-Vertical Data Negotiation Response Time	Maintain an average response time of < 30 sec in data/resource sharing negotiations between Edge/Cloud devices across PUC sites	(2), (3)
KPI_CV5	Computing-Energy-Security Thresholds	Achieve > 90% satisfaction of the minimum Computing-Energy-Security thresholds set by each vertical user	(2), (4), (6)

Funded by the European Union
Project funded by

KPI_CV6	Transaction Recording	Ensure 100% recording of every transaction within the common vertical infrastructure through DDAG DLT	(3), (4)
KPI_CV7	Global Game Optimization	Achieve > 90% convergence of the Global Game optimization between PUCs	(2), (4), (6)

KPI_CV1 ("Common Computing Space") is proposed to evaluate the successful establishment of a common computing space between different PUC sites, realized through the Global versions of the decentralized Game, measuring the ability of interconnected verticals (manufacturing, mobility and healthcare) to efficiently share computational resources and insights while maintaining operational independence. By enabling cooperative computing, this KPI ensures optimized resource utilization, scalability and enhanced cross-vertical interoperability.

KPI_CV2 ("Supply Chain Risk Correlation") is focused on achieving 100% correlation of supply-chain risk events and threat intelligence across different PUC sites by evaluating the effectiveness of shared security intelligence in detecting, analysing and mitigating threats that affect multiple sectors. By synchronizing risk event reporting across manufacturing, mobility and healthcare, this KPI enhances resilience and proactive response against cybersecurity threats, minimizing disruption across supply chains.

KPI_CV3 ("Synchronized AI Co-processing") is related to the evaluation of CoGNETs' ability to achieve 100% synchronized AI model training across different PUC sites, ensuring that at least two AI service models can be trained in parallel. This KPI measures the efficiency of decentralized learning frameworks in enabling real-time AI model updates across industries, leading to improved predictive analytics, faster adaptation to changing conditions and increased AI model accuracy for cross-sector applications.

KPI_CV4 ("Cross-Vertical Data Negotiation Response Time") is related to the maintenance of an average response time of less than 30 seconds for data/resource sharing negotiations between Edge/Cloud devices of different PUC sites. By reducing latency in critical decision-making processes and improving overall efficiency in cross-sector collaboration, this KPI ensures that computational resources, data access and AI models can be rapidly and securely exchanged between all verticals.

KPI_CV5 ("Computing-Energy-Security Thresholds") is related to the evaluation of CoGNETs' ability to meet at least 90% of the minimum computing, energy and security thresholds defined by users across different verticals. Through its alignment with industry-specific constraints and regulatory requirements, this KPI ensures that the cross-vertical infrastructures provide balanced resource allocation, secure data handling and energy-efficient computing.

KPI_CV6 ("Transaction Recording") is focused on ensuring 100% recording of all transactions within the common vertical infrastructure using DAG-based DLT, thereby allowing for traceability, transparency and auditability of all cross-vertical interactions, enhancing trust, security, and compliance with industry standards and regulatory frameworks.

KPI_CV7 ("Global Game Optimization") is proposed for measuring the success of achieving > 90% convergence of the Global Game optimization between PUCs, by assessing the efficiency of decentralized optimization mechanisms in balancing resource distribution, computational loads and security requirements across all different verticals. By achieving high

Funded by the European Union
Project funded by

convergence rates, this KPI ensures that the cross-vertical infrastructure operates optimally, benefiting all participating industries.

8.4.6 Risk Assessment & Mitigation Plan

The implementation of the cross-vertical supply chain PUC scenario presents several potential risks that may impact the seamless integration, operation and security of the interconnected manufacturing, mobility and healthcare verticals, which can, however, be mitigated through strategic planning, technical enhancements and robust security measures.

One of the main risks is related to inefficiencies in cross-vertical data synchronization, as this PUC depends on the real-time exchange and processing of data across different verticals. To mitigate this risk, decentralized time-stamping protocols and distributed caching mechanisms will be implemented to ensure real-time synchronization between PUC sites. Additionally, the supplementary use of Edge processing will reduce the dependence on centralized cloud systems, minimizing delays.

Another key concern is related to cybersecurity threats and data breaches caused by CoGNETs' shared supply-chain operations, where unauthorized access or cyberattacks pose a significant risk to the integrity of the system. To address this, multi-factor authentication (MFA), AI-based anomaly detection and encrypted data transactions using DDAG DLT will be integrated to help detect and mitigate potential threats.

Inconsistent computing resource allocation across different verticals is another potential risk, where each vertical may have varying computing, energy and security requirements, leading to imbalanced resource distribution and inefficiencies in processing. To mitigate this, AI-driven workload distribution algorithms will be deployed to dynamically allocate computing resources based on demand, ensuring optimized Computing-Energy-Security balance for each vertical.

Finally, slow response time in cross-vertical negotiations is another potential risk that can be prevented through smart contract-based automation and network bandwidth optimization, to ensure fast and seamless data transmissions.

Funded by the European Union
Project funded by

9 CONCLUSIONS

The document offers a comprehensive exploration of the CoGNETs project from both architectural and requirements perspectives. It investigates the foundational architectural principles, analyses key logical building blocks, and clarifies the various layers and their intercommunication.

Additionally, from a technical standpoint, the CoGNETs project represents a complex and ambitious initiative designed to create a seamless connection across the cloud-edge continuum. It emphasizes self-organization and security self-configuration within the mesh network, while also addressing decentralized identity management and security protocols at the hardware, software, and AI levels. The clearly defined architecture and logical building blocks are crucial for achieving the project's objectives, as they facilitate a thorough understanding of the distinct layers. This clarity enables a deeper comprehension of the overall CoGNETs architecture, ensuring that all elements work harmoniously to support the project's vision and goals.

Moreover, the introduction of collaborative learning as the solution to be implemented in the CoGNETs architecture has been analysed through a review of the literature and the collection of requirements from external experts. Additionally, specific logical building blocks has been introduced with the purpose to facilitate the splitting of neural networks and/or deep-learning processes. CoGNETs architecture facilitates the execution of AI submodules separating the data plane and the processing plane and assuring the security communication of the different AI modules. The next steps will consist on the development of the logical building blocks as well the data structures to facilitate this operation in work package 3 and 4.

The adoption of standardized interfaces is a crucial factor in expanding the CoGNETs ecosystem, as evidenced by the platform's requirement definitions introduced in this document, that facilitate seamless integration with third-party services. This approach allows users to capitalize on the benefits of the various logical building blocks while circumventing the limitations associated with non-open APIs and specifications, thereby enhancing system flexibility and interoperability. Such flexibility ensures that the CoGNETs platform can evolve towards new requirements in the future, paving the way for its application in diverse scenarios.

Additionally, the deliverable details various Pilot Use Cases, offering insights into the systems and solutions intended for analysis and how to get advantages in the adoption of the CoGNETs solution. It emphasizes that the document serves as a snapshot of the current definitions of these pilots, which may be refined during implementation of the CoGNETs logical building blocks, particularly in response to changes within the CoGNETs architecture. The next steps involve developing and testing these logical building blocks to evaluate their desirability and technical feasibility, as well as the requirements outlined by the pilot use cases. This process will also include measuring the KPIs defined for each use case, with the ultimate goal of achieving the target KPIs.

Furthermore, the defined pilot use cases have outlined the specific data that needs to be managed. CoGNETs will thoroughly analyse these data requirements and focus on developing detailed specifications for data model schemas to enhance the interoperability of the data utilized by them. This initiative aims to ensure that the data can be seamlessly shared and integrated across different systems and applications. In addition, this effort will be aligned with the Smart Data Models program, which seeks to disseminate the work produced within CoGNETs and promote the adoption of these data model schemas beyond the project's conclusion. By actively engaging with the Smart Data Models program, CoGNETs not only aims to extend the impact of its findings but also to create a sustainable framework that encourages ongoing collaboration and innovation in data management practices. This approach will

Funded by the European Union

Project funded by

ultimately support a broader ecosystem and facilitate the integration of CoGNETs' solutions in various future applications and scenarios.

Therefore, a clear understanding of definitions and requirements is essential for the successful development and implementation of any project. In the context of CoGNETs, the established process meticulously defines the project's scope, objectives, and constraints, providing a robust foundation that significantly reduces ambiguity during the implementation of the logical building blocks in Work Packages 3 and 4.

A comprehensive effort has been made to identify and document specific requirements, ensuring they align with the overall needs and expectations of the project. This activity also encompasses the analysis of constraints and values that may affect the implementation of these requirements. During the implementation phase of these logical building blocks, these requirements will be closely monitored and assessed to ensure they are met. If certain requirements cannot be achieved, a detailed explanation will be provided, outlining the measurements obtained and the reasons for any deviations from the expected constraints or values.

This level of clarity empowers project teams to effectively plan, execute, and deliver results that not only meet but can also exceed expectations. Furthermore, it fosters an environment of accountability and continuous improvement, allowing teams to learn from any challenges faced during implementation. By maintaining a transparent process and actively addressing potential issues, CoGNETs can enhance its adaptability and responsiveness, ultimately leading to successful project outcomes and stakeholder satisfaction.

Finally, the ongoing monitoring of these requirements throughout the project execution is crucial for ensuring that the defined specifications are effectively incorporated into the implementation of the CoGNETs logical building blocks. This continuous follow-up not only reinforces accountability but also allows for timely adjustments and refinements as needed. By maintaining a vigilant approach to requirements management, the project can adapt to any unforeseen challenges that may arise, thereby enhancing the likelihood of achieving a successful outcome. Ultimately, this diligent oversight lays the groundwork for delivering results that align with the project's goals and fulfil stakeholder expectations, contributing to the overall success of the CoGNETs initiative.

Funded by the European Union

Project funded by

REFERENCES

[1] Gkonis, P., Giannopoulos, A., Trakadas, P., Masip-Bruin, X., & D'ANode Manager - Device Registration (NDR)ia, F. (2023). A survey on IoT-edge-cloud continuum systems: status, challenges, use cases, and open issues. Future Internet, 15(12), 383.

[2] Carnevale, L., Celesti, A., Galletta, A., Dustdar, S., & Villari, M. (2018, May). From the cloud to edge and IoT: a smart orchestration architecture for enabling osmotic computing. In 2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA) (pp. 419-424). IEEE.

[3] Truong, H. L., & Magoutis, K. (2022, December). Robustness via Elasticity Accelerators for the IoT-Edge-Cloud Continuum. In 2022 IEEE/ACM 15th International Conference on Utility and Cloud Computing (UCC) (pp. 291-296). IEEE.

[4] Taivalsaari, A., Mikkonen, T., & Pautasso, C. (2021, May). Towards seamless IoT device-edge-cloud continuum: Software architecture options of IoT devices revisited. In International Conference on Web Engineering (pp. 82-98). Cham: Springer International Publishing.

[5] Judvaitis, J., Blumbergs, E., Arzovs, A., Mackus, A. I., Balass, R., & Selavo, L. (2024). A Set of Tools and Data Management Framework for the IoT–Edge–Cloud Continuum. Applied System Innovation, 7(6), 130.

[6] Pietri, I. et al. (2025). Decentralized Management of IoT Platform Federations and Data Marketplaces. In: Presser, M., Skarmeta, A., Krco, S., González Vidal, A. (eds) Global Internet of Things and Edge Computing Summit. GIECS 2024. Communications in Computer and Information Science, vol 2328. Springer, Cham. https://doi.org/10.1007/978-3-031-78572-6_7

[7] B. Wang, S. Evergreen, and J. Forest, "Game theory in smart grids: Strategic decision-making for renewable energy integration," Sustainable Cities and Society, vol. 108, p. 105480, 2024.

[8] J. von Neumann and O. Morgenstern, Theory of Games and Economic Behavior. Princeton University Press, 1944.

[9] R. B. Myerson, Game Theory: Analysis of Conflict. Harvard University Press, 1997.

[10] N. Giocoli, "Nash equilibrium," History of political economy, vol. 36, no. 4, pp. 639–666, 2004.

[11] V. Krishna, Auction Theory. Academic Press, 2009.

[12] P. R. Milgrom, "A theory of auctions and competitive bidding," Econometrica, vol. 50, no. 5, pp. 1089–1122, 1982.

[13] W. Vickrey, "Counterspeculation, auctions, and competitive sealed tenders," Journal of Finance, vol. 16, no. 1, pp. 8–37, 1961.

[14] K. Hasker and R. Sickles, "ebay in the economic literature: Analysis of an auction marketplace," Review of Industrial Organization, vol. 37, pp. 3–42, 2010.

[15] T. E. Rockoff and M. Groves, "Design of an internet-based system for remote dutch auctions," Internet Research, vol. 5, no. 4, pp. 10–16, 1995.

[16] J. McMillan, "Selling spectrum rights," The Journal of Economic Perspectives, vol. 8, no. 3, pp. 145–162, 1994.

[17] R. Alvarez and M. Nojoumian, "Comprehensive survey on privacy-preserving protocols for sealed-bid auctions," Computers & Security, vol. 88, p. 101502, 2020.

Funded by the European Union

Project funded by

[18] S. De Vries and R. V. Vohra, “Combinatorial auctions: A survey,” INFORMS Journal on computing, vol. 15, no. 3, pp. 284–309, 2003.

[19] J. Cokelais et al., “Combinatorial auctions for cloud resource allocation,” Journal of Cloud Computing, vol. 8, pp. 1–16, 2020.W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, “Edge computing: Vision and challenges,” IEEE Internet of Things Journal, vol. 3, no. 5, pp. 637–646, 2016.

[20] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, “Edge computing: Vision and challenges,” IEEE Internet of Things Journal, vol. 3, no. 5, pp. 637–646, 2016.

[21] M. Satyanarayanan, “The emergence of edge computing,” Computer, vol. 50, no. 1, pp. 30–39, 2017.

[22] Z. Xu, H. Yu, J. Liang, C. S. Yeo, X. Li, and Y. Zhang, “Auction-based resource allocation for edge computing,” IEEE Transactions on Services Computing, vol. 13, no. 3, pp. 564–577, 2020.

[23] T. Wang, J. Liang, K. Liu, and W. Xu, “Auction-based resource allocation for cloud and edge computing,” IEEE Access, vol. 6, pp. 33540–33550, 2018.

[24] Q. Zhang, L. Cheng, and R. Boutaba, “Cloud computing: State-of-the-art and research challenges,” Journal of Internet Services and Applications, vol. 1, pp. 7–18, 2013.

[25] X. Chen, J. Li, Q. Ma, H. Yin, and H. Wang, “Auction approaches for edge resource management: A survey,” Future Generation Computer Systems, vol. 97, pp. 71–85, 2018.

[26] C. Avasalcai, C. Tsigkanos, and S. Dustdar, “Decentralized resource auctioning for latency-sensitive edge computing,” in 2019 IEEE international conference on edge computing (EDGE), pp. 72–76, IEEE, 2019.

[27] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, “A survey on mobile edge computing: The communication perspective,” IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2322–2358, 2017.

[28] T. X. Tran and D. Pompili, “Joint task offloading and resource allocation for multi-server mobile-edge computing networks,” IEEE Transactions on Vehicular Technology, vol. 68, no. 1, pp. 856–868, 2018.

[29] J. Li, S. Wang, Q. Zhang, and W. Lin, “Learning-based resource allocation for edge computing,” IEEE Transactions on Network and Service Management, vol. 15, no. 3, pp. 856–870, 2018.

[30] P. Mach and Z. Becvar, “Mobile edge computing: A survey on architecture and computation offloading,” IEEE communications surveys & tutorials, vol. 19, no. 3, pp. 1628–1656, 2017.

[31] Y. Shen, C. Shepherd, C. M. Ahmed, S. Shen, X. Wu, W. Ke, and S. Yu, “Game-theoretic analytics for privacy preservation in internet of things networks: A survey,” Engineering Applications of Artificial Intelligence, vol. 133, p. 108449, 2024.

[32] K. Li, “A game theoretic approach to computation offloading strategy optimization for non-cooperative users in mobile edge computing,” IEEE Transactions on Sustainable Computing, 2018.

[33] X. Yang, H. Luo, Y. Sun, J. Zou, and M. Guizani, “Coalitional game-based cooperative computation offloading in mec for reusable tasks,” IEEE Internet of Things Journal, vol. 8, no. 16, pp. 12968–12982, 2021.

[34] Calheiros RN, Ranjan R, Beloglazov A, De Rose CA, Buyya R. CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource

Funded by the European Union

Project funded by

provisioning algorithms. Software: Practice and experience. 2011 Jan;41(1):23-50. https://doi.org/10.1002/spe.995

[35] X. Fan, Q. Chai, L. Xu, and D. Guo, "DIAM-IoT: A Decentralized Identity and Access Management Framework for Internet of Things," in Proceedings of the 2nd ACM International Symposium on Blockchain and Secure Critical Infrastructure (BSCI '20), Taipei, Taiwan, Oct. 2020, pp. 1–6, doi: 10.1145/3384943.3409436.

[36] Shi W, Cao J, Zhang Q, Li Y, Xu L. Edge computing: Vision and challenges. IEEE internet of things journal. 2016 Jun 9;3(5):637-46. 10.1109/JIOT.2016.2579198

[37] Jonas, E., Schleier-Smith, J., Sreekanti, V., Tsai, C.C., Khandelwal, A., Pu, Q., Shankar, V., Carreira, J., Krauth, K., Yadwadkar, N. and Gonzalez, J.E., 2019. Cloud programming simplified: A berkeley view on serverless computing. arXiv preprint arXiv:1902.03383

[38] von Perbandt, C., Tyca, M., Koschel, A. and Astrova, I., 2022. Development support for intelligent systems: test, evaluation, and analysis of microservices. In Intelligent Systems and Applications: Proceedings of the 2021 Intelligent Systems Conference (IntelliSys) Volume 1 (pp. 857-875). Springer International Publishing.

[39] He, J., Chen, Y., Fu, T.Z., Long, X., Winslett, M., You, L. and Zhang, Z., 2018, July. Haas: Cloud-based real-time data analytics with heterogeneity-aware scheduling. In 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS) (pp. 1017-1028). IEEE.

[40] Burns, B., Beda, J., Hightower, K. and Evenson, L., 2022. Kubernetes: up and running: dive into the future of infrastructure. " O'Reilly Media, Inc.".

[41] Mahnke W, Leitner SH, Damm M. OPC unified architecture. Springer Science & Business Media; 2009 Apr 5.

[42] Gazzarata, R., Almeida, J., Lindsköld, L., Cangioli, G., Gaeta, E., Fico, G. and Chronaki, C.E., 2024. HL7 Fast Healthcare Interoperability Resources (HL7 FHIR) in digital healthcare ecosystems for chronic disease management: Scoping review. International journal of medical informatics, p.105507.

[43] Schuman CD, Kulkarni SR, Parsa M, Mitchell JP, Date P, Kay B. Opportunities for neuromorphic computing algorithms and applications. Nature Computational Science. 2022 Jan;2(1):10-9.

[44] Tardieu H. Role of Gaia-X in the European data space ecosystem. InDesigning Data Spaces: The Ecosystem Approach to Competitive Advantage 2022 Jul 22 (pp. 41-59). Cham: Springer International Publishing.

[45] Kairouz, P., McMahan, H., Avent, B., Bellet, A., Bennis, M., Bhagoji, A., Bonawit, K., Charles, Z., Cormode, G., Cummings, R., D'Oliveira, R., Eichner, H., El Rouayheb, S., Evans, D., Gardner, J., Garrett, Z., Gascón, A., Ghazi, B., Gibbons, P., Gruteser, M., Harchaoui, Z., He, C., He, L., Huo, Z., Hutchinson, B., Hsu, J., Jaggi, M., Javidi, T., Joshi, G., Khodak, M., Konecný, J., Korolova, A., Koushanfar, F., Koyejo, S., Lepoint, T., Liu, Y., Mittal, P., Mohri, M., Nock, R., Özgür, A., Pagh, R., Qi, H., Ramage, D., Raskar, R., Raykova, M., Song, D., Song, W., Stich, S., Sun, Z., Theertha Suresh, A., Tramèr, F., Vepakomma, P., Wang, J., Xiong, L., Xu, Z., Yang, Q., Yu, F., Yu, H., & Zhao, S. (2021). Advances and Open Problems in Federated Learning. Now Foundations and Trends.

[46] Vepakomma, P., Gupta, O., Swedish, T., Raskar R. (2018). Split learning for health: Distributed deep learning without sharing raw patient data,". ArXiv preprint arXiv:1812.00564.

Funded by the European Union

Project funded by

[47] Lin, Z., Qu, G., Chen, X., & Huang, K. (2024). Split Learning in 6G Edge Networks. IEEE Wireless Communications, 31(4), 170-176.

[48] Thapa, C., Chamikara, M., Camtepe, S., Sun L. (2022). Splitfed: When Federated Learning Meets Split Learning. ArXiv preprint arXiv:2004.12088.

[49] Joshi, P., Thapa, C.,Camtepe, S., Hasanuzzamana, M., Scully, T., Afli, H. (2021). Split-fed learning without client-side synchronization: Analyzing client-side split network portion size to overall performance. ArXiv preprint arXiv:2109.09246.

[50] Wu, W., Li, M., Qu, K., Zhou, C., Shen, X., Zhuang, W., Li, X., & Shi, W. (2023). Split Learning Over Wireless Networks: Parallel Design and Resource Management. IEEE Journal on Selected Areas in Communications, 41(4), 1051-1066.

[51] Kanjula, K. R., & Kolla, S. M. (2023). Distributed Swarm Intelligence. arXiv preprint arXiv:2301.13276.

[52] Vahidalizadehdizaj, M., Jadav, J., & Tao, L. (2015, July). Security challenges in swarm intelligence. In 2015 6th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-4). IEEE.

[53] Rong, C., Nguyen, S. T., & Jaatun, M. G. (2013). Beyond lightning: A survey on security challenges in cloud computing. Computers & Electrical Engineering, 39(1), 47-54.

[54] Abualigah, L., Falcone, D., & Forestiero, A. (2023). Swarm intelligence to face IoT challenges. Computational Intelligence and Neuroscience, 2023(1), 4254194.

[55] Sun, Z., Wei, M., Zhang, Z., & Qu, G. (2019). Secure routing protocol based on multi-objective ant-colony-optimization for wireless sensor networks. Applied Soft Computing, 77, 366-375.

[56] Alohali, M. A., Elsadig, M., Al-Wesabi, F. N., Duhayyim, M. al, Hilal, A. M., & Motwakel, A. (2023). Swarm intelligence for IoT attack detection in fog-enabled cyber-physical system. Computers and Electrical Engineering, 108, 108676. https://doi.org/10.1016/j.compeleceng.2023.108676

[57] Zhou, J., Shen, Y., Li, L., Zhuo, C., & Chen, M. (2023). Swarm Intelligence-Based Task Scheduling for Enhancing Security for IoT Devices. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 42(6), 1756–1769. https://doi.org/10.1109/TCAD.2022.3207328

[58] Statista, "Internet of Things (IoT) connected devices worldwide 2015–2025," [Online]. Available: https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/. [Accessed: Feb. 25, 2025].

[59] W3C, "Decentralized Identifiers (DIDs) v1.0," W3C Recommendation, 2021. [Online]. Available: https://www.w3.org/TR/did-1.0/. [Accessed: 25-Feb-2025].

[60] Fdhila, W., Stifter, N., Kostal, K., Saglam, C., Sabadello, M. (2021). Methods for Decentralized Identities: Evaluation and Insights. In: Business Process Management: Blockchain and Robotic Process Automation Forum. BPM 2021. Lecture Notes in Business Information Processing, vol 428. Springer, Cham. https://doi.org/10.1007/978-3-030-85867-4_9

[61] Ghesmati, S., Fdhila, W., Weippl, E. (2021). Studying Bitcoin Privacy Attacks and Their Impact on Bitcoin-Based Identity Methods. In: Business Process Management: Blockchain and Robotic Process Automation Forum. BPM 2021. Lecture Notes in Business Information Processing, vol 428. Springer, Cham. https://doi.org/10.1007/978-3-030-85867-4_7

Funded by the European Union

Project funded by

[62] Kortesniemi, Yki, Lagutin, Dmitrij, Elo, Tommi, Fotiou, Nikos, Improving the Privacy of IoT with Decentralised Identifiers (DIDs), Journal of Computer Networks and Communications 2019, https://doi.org/10.1155/2019/8706760

[63] Lagutin, D., Kortesniemi, Y., Fotiou, N., & Siris, V.A. (2019). Enabling Decentralised Identifiers and Verifiable Credentials for Constrained IoT Devices using OAuth-based Delegation. Proceedings 2019 Workshop on Decentralized IoT Systems and Security.

[64] Fedrecheski, G., Costa, L.C.P., Afzal, S., Rabaey, J.M., Lopes, R.D., Zuffo, M.K. (2022). A Low-Overhead Approach for Self-sovereign Identity in IoT. In: González-Vidal, A., Mohamed Abdelgawad, A., Sabir, E., Ziegler, S., Ladid, L. (eds) Internet of Things. GIoTS 2022. Lecture Notes in Computer Science, vol 13533. Springer, Cham. https://doi.org/10.1007/978-3-031-20936-9_21

[65] R. Ansey, J. Kempf, O. Berzin, C. Xi and I. Sheikh, "Gnomon: Decentralized Identifiers for Securing 5G Iot Device Registration and Software Update," 2019 IEEE Globecom Workshops (GC Wkshps), Waikoloa, HI, USA, 2019, pp. 1-6, doi: 10.1109/GCWkshps45667.2019.9024702.

[66] S. Han, J. Kim, H. Lee and E. Hwang, "Signature Analysis of SRAM-PUF for IoT Decentralized Identifier in Large-Scale Networks," 2023 Fourteenth International Conference on Ubiquitous and Future Networks (ICUFN), Paris, France, 2023, pp. 103-105, doi: 10.1109/ICUFN57995.2023.10200238.

[67] Y. Su, J. Wu, C. Long, and L. Wei, “Secure decentralized machine identifiers for Internet of Things,” in Proc. 2020 2nd Int. Conf. Blockchain Technol. (ICBCT '20), New York, NY, USA, 2020, pp. 57–62, doi: 10.1145/3390566.3391670.

[68] Uzair Javaid, Muhammad Naveed Aman, and Biplab Sikdar. 2018. BlockPro: Blockchain based Data Provenance and Integrity for Secure IoT Environments. In Proceedings of the 1st Workshop on Blockchain-enabled Networked Sensor Systems (BlockSys '18). Association for Computing Machinery, New York, NY, USA, 13–18. https://doi.org/10.1145/3282278.3282281

[69] Patil, A. S., Hamza, R., Hassan, A., Jiang, N., Yan, H., & Li, J. (2020). Efficient privacy-preserving authentication protocol using PUFs with blockchain smart contracts. Computers & Security, 97, 101958. https://doi.org/10.1016/j.cose.2020.101958

[70] L. Sanchez, F. Restuccia, and A. Markantonakis, “A Distributed Context Broker for Large-Scale IoT Applications,” IEEE Internet of Things Journal, vol. 7, no. 3, pp. 2180–2190, 2020.

[71] X. Li, Y. Xu, and Z. Zhao, “Semantic Interoperability for Smart City Data: A Case Study with FIWARE,” IEEE Transactions on Industrial Informatics, vol. 15, no. 6, pp. 3642–3650, 2019.

[72] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, “Context Aware Computing for the Internet of Things: A Survey,” IEEE Communications Surveys & Tutorials, vol. 16, no. 1, pp. 414–454, 2014.

[73] M. Botta, W. de Donato, V. Persico, and A. Pescape, “Integration of Cloud Computing and Internet of Things: A Survey,” Future Generation Computer Systems, vol. 56, pp. 684–700, 2016.

[74] M. Aazam, I. Khan, and S. Zeadally, “Fog Computing and its Applications: A Review,” Future Generation Computer Systems, vol. 66, pp. 228–245, 2017.

[75] W3C® Proposed Recommendation 21 May 2024: "RDF Dataset Canonicalization. A Standard RDF Dataset Canonicalization Algorithm". Available at https://w3c.github.io/rdf-canon/spec.

Funded by the European Union

Project funded by

[76] W3C® Candidate Recommendation Draft 20 June 2024: "Verifiable Credential Data Integrity 1.0. Securing the Integrity of Verifiable Credential Data ". Available at https://github.com/w3c/vc-data-integrity.

[77] IETF RFC 8785: "JSON Canonicalization Scheme (JCS)". Available at https://datatracker.ietf.org/doc/html/rfc8785.

[78] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, “A view of cloud computing,” Communications of the ACM, vol. 53, no. 4, pp. 50–58, 2010.

[79] A. Shawish and M. Salama, “Cloud computing: paradigms and technologies,” in Inter-cooperative collective intelligence: Techniques and applications, pp. 39–67, Springer, 2013.

[80] D. Kovachev, T. Yu, and R. Klamma, “Adaptive computation offloading from mobile devices into the cloud,” in 2012 IEEE 10th International Symposium on Parallel and Distributed Processing with Applications, pp. 784–791, IEEE, 2012.

[81] B. Charyyev, E. Arslan, and M. H. Gunes, “Latency comparison of cloud datacenters and edge servers,” in GLOBECOM 2020-2020 IEEE Global Communications Conference, pp. 1–6, IEEE, 2020

[82] W. Z. Khan, E. Ahmed, S. Hakak, I. Yaqoob, and A. Ahmed, “Edge computing: A survey,” Future Generation Computer Systems, vol. 97, pp. 219–235, 2019.

[83] L. Lin, X. Liao, H. Jin, and P. Li, “Computation offloading toward edge computing,” Proceedings of the IEEE, vol. 107, no. 8, pp. 1584–1607, 2019.

[84] R. Morabito, J. Kjällman, and M. Komu, “Hypervisors vs. lightweight virtualization: A performance comparison,” IEEE Transactions on Cloud Computing, vol. 6, no. 1, pp. 1–14, 2017.

[85] A. Saboor, M. F. Hassan, R. Akbar, S. N. M. Shah, F. Hassan, S. A. Magsi, and M. A. Siddiqui, “Containerized microservices orchestration and provisioning in cloud computing: A conceptual framework and future perspectives,” Applied Sciences, vol. 12, no. 12, p. 5793, 2022.

[86] N. Sharghivand, F. Derakhshan, and N. Siasi, “A comprehensive survey on auction mechanism design for cloud/edge resource management and pricing,” IEEE Access, vol. 9, pp. 126502–126529, 2021.

[87] X. Zhang, N. Wuwong, H. Li, and X. Zhang, “Information security risk management framework for the cloud computing environments,” in 2010 10th IEEE international conference on computer and information technology, pp. 1328–1334, IEEE, 2010.

[88] C. Tang and H. Wu, “Optimal computational resource pricing in vehicular edge computing: A stackelberg game approach,” Journal of Systems Architecture, vol. 121, p. 102331, 2021.

Funded by the European Union

Project funded by