D2.4

The OPTIMAI architecture specifications - 1st version

31 December 2021

OPTIMAI

This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 958264

DOCUMENT SUMMARY INFORMATION

Grant Agreement No	958264	**Acronym**	OPTIMAI
Full Title	Optimizing Manufacturing Processes through Artificial Intelligence and Virtualization		
Start Date	01/01/2021	**Duration**	36 months
Deliverable	D2.4: The OPTIMAI architecture specifications - 1st version		
Work Package	WP2 - User requirements, Technical Specifications and Use case analysis		
Nature	Report	**Dissemination Level**	Public
Lead Beneficiary	FORTH		
Authors	Konstantinos C. Apostolakis (FORTH) Dimitrios Arampatzis (FORTH) George Margetis (FORTH)		
Co-authors	Christina Tsita (CERTH) George Bogdos (FINT) Andreas Böttinger (EVT) Fernando Ubis (VIS) Elpiniki Papageorgiou (UTH) Sabrina Verardi (ENG) Manfredi Giuseppe Pistone (ENG) Andrea Gomez (UNIMET) Andrea Guillén (UAB) Emma Teodoro (UAB) Paul Hayes (TRI) Agata Gurzawska (TRI)		

DISCLAIMER

The OPTIMAI Project receives funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 958264. The sole responsibility for the content of this document lies with the authors. It does not necessarily reflect the opinion of the European Union. The European Commission is not responsible for any use that may be made of the information contained herein.

DOCUMENT HISTORY

Version	Date	Changes	Contributor(s)
V0.1	01/10/2021	Initial deliverable structure	Konstantinos C. Apostolakis (FORTH)
V0.2	21/10/2021	UNIMET and UTH component template provision and integration	Andrea Gomez (UNIMET) Elpiniki Papageorgiou (UTH)
V0.3	25/10/2021	CERTH and EVT component template provision and integration	Christina Tsita (CERTH) Andreas Böttinger (EVT)
V0.4	15/11/2021	50% of the deliverable content, including content integration from ENG and FINT	Konstantinos C. Apostolakis (FORTH) Dimitrios Arampatzis (FORTH) George Margetis (FORTH) Sabrina Verardi (ENG) George Bogdos (FINT)
V0.5	10/12/2021	CERTH components' information flow description and ENG component clarifications based on 2nd Architectural Workshop Inputs received.	Christina Tsita (CERTH) Sabrina Verardi (ENG)
V0.6	14/12/2021	Ethics and Legal Compliance by Design Section	Paul Hayes (TRI) Agata Gurzawska (TRI) Andrea Guillén (UAB) Emma Teodoro (UAB)
V0.7	15/12/2021	Digital Twinning subsystem & visual simulation engine components description	Fernando Ubis (VIS)
V0.8	16/12/2021	ENG components' information flow description based on 2nd Architectural Workshop Inputs received.	Manfredi Giuseppe Pistone (ENG)
V0.9	16/12/2021	FINT Middleware components refinement	George Bogdos (FINT)
V1.0	17/12/2021	Internal Review Version compiled and sent to internal reviewers.	Konstantinos C. Apostolakis (FORTH)

			Dimitrios Arampatzis (FORTH) George Margetis (FORTH)
V1.1	21/12/2021	1st version with suggested revisions	Nikolaos Dimitriou (CERTH)
V1.2	21/12/2021	2nd version with suggested revisions	Sabrina Verardi (ENG) Manfredi Giuseppe Pistone (ENG)
V1.3	22/12/2021	3rd version with suggested revisions	George Bogdos (FINT)
V1.5	23/12/2021	First revisions after internal review & Final Version for Quality & Security Review	Konstantinos C. Apostolakis (FORTH)
V1.6	23/12/2021	Quality check	Sabrina Verardi (ENG)
V2.0	28/12/2021	Final version for submission	George Margetis (FORTH)

PROJECT PARTNERS

Logo	Partner	Country	Short name
	ETHNIKO KENTRO EREVNAS KAI TECHNOLOGIKIS ANAPTYXIS	Greece	**CERTH**
	FINT FUTURE INTELLINGENCE LIMITED	Cyprus	**FINT**
	IDRYMA TECHNOLOGIAS KAI EREVNAS	Greece	**FORTH**
	EVT EYE VISION TECHNOLOGY GMBH	Germany	**EVT**
	VISUAL COMPONENTS OY	Finland	**VIS**
	YOUBIQUO SRL	Italy	**YBQ**
	PANEPISTIMIO THESSALIAS	Greece	**UTH**
	ENGINEERING – INGEGNERIA INFORMATICA SPA	Italy	**ENG**
	UNIMETRIK SA	Spain	**UNIMET**
	UNIVERSITAT POLITECNICA DE VALENCIA	Spain	**UPV**
	Carr Communications Limited	Ireland	**CARR**
	UNIVERSIDAD AUTONOMA DE BARCELONA	Spain	**UAB**
	TRILATERAL RESEARCH LIMITED	Ireland	**TRI**
	KLEEMANN HELLAS –INDUSTRIAL COMMERCIAL SOCIETE ANONYME FOR MECHANICAL CONSTRUCTION SA	GREECE	**KLEE**
	TELEVES SA	Spain	**TVES**
	MICROCHIP TECHNOLOGY CALDICOT LIMITED	United Kingdom	**MTCL**

LIST OF ABBREVIATIONS

Abbreviation	Definition
AI	Artificial Intelligence
API	Application Programming Interface
AR	Augmented Reality
BaaS	Blockchain-as-a-Service
BOM	Bill of Materials
CEP	Complex Event Processing
CPPS	Cyber-Physical Production System
DML	Dedicated Manufacturing Line
DoA	Description of Action
DT	Digital Twin
EbD	Ethics by Design
ERP	Enterprise Resource Planning
F4I	FIWARE4Industry
FMS	Flexible Manufacturing System
FPGA	Field-Programmable Gate Array
GA	Grant Agreement
GAN	Generative Adversarial Network
GDTA	Generic Digital Twin Architecture
GE	Generic Enabler
HMI	Human-Machine Interface
HITL	Human-in-the-Loop
HUD	Heads-Up Display
I4.0	Industry 4.0
IaaS	Infrastructure-as-a-Service
ICT	Information & Communication Technologies
IEC	International Electrotechnical Commission
IFD	Information Flow Diagram
IIRA	Industrial Internet Reference Architecture
(I)IoT	(Industrial) Internet of Things

ISO	International Organization for Standardization
IVRA	Industrial Value Chain Reference Architecture
LASFA	LASIM Smart Factory
LASIM	Laboratory for handling, assembly and pneumatics (Acronym in Slovene)
LCtD	Legal Compliance through Design
LDS	Loosely-Defined Standards
LIDAR	Light Detection And Ranging
LSTM	Long Short Term Memory
MES	Manufacturing Execution Systems
MQTT	Message Queuing Telemetry Transport
NGSI	Next Generation Service Interfaces
OMIDES	Operator-Machine Interaction & Decision Support
OPC UA	OPC (Foundation) Unified Architecture
OT	Operation Technologies
PaaS	Platform-as-a-Service
PC	Personal Computer
PCE	Production Control Engineer
QCS	Quality Control Sensors
R&D	Research & Development
RA	Reference Architecture
RAMI	Reference Architectural Model Industrie
RBAC	Role-Based Access Control
REST	REpresentational State Transfer
ROI	Return on Investment
SaaS	Software-as-a-Service
SCADA	Supervisory Control and Data Acquisition
SCI 4.0	Standardization Council Industrie 4.0
SITAM	Stuttgart IT Architecture for Manufacturing
SOA	Service-Oriented Architecture
SoS	System-of-Systems
SSO	Single Sign-On
ToF	Time of Flight

TVOC	Total Volatile Organic Compounds
UC	Use Case
UI	User Interface
VC4.0	Visual Components 4.0

Executive Summary

This document contains the preliminary description of the OPTIMAI smart manufacturing solution architecture based on the elicited stakeholders' requirements and use case scenario definitions that preceded it.

This first version architecture (M12) is the outcome of a three-step design methodology which started with relevant technology exploration in the context of D2.3. This process was followed-up through a closer examination of the most prominent reference architectural models provisioned for smart manufacturing and industrial Internet of Things applications. A top-down design approach was then carried out using the original OPTIMAI architecture proposition as a starting point so as to identify the various components and subsystems that deliver on the specified needs and requirements of the end-users. Through this exercise, the architecture was broken down into 36 basal components. Each one of those base elements was then elaborated by project partners responsible for their implementation through a bottom-up functional specification. Through this process, three architectural viewpoints are defined for the OPTIMAI envisioned solution in this document, namely the *functional*, *information* and *deployment* view.

The functional view delivers a high-level overview of the envisioned system functionality broken down into the identified subsystems and individual components, all of whom are described in terms of their foreseen roles and responsibilities within the runtime operation of the system. Aspects related to integration, such as the interrelationships among platform components are presented, in order to guide the development of the necessary intercommunication mechanisms between components. This process is complemented by means of aligning the resulting architectural components to prominent Industry 4.0 reference architecture models and principles.

The Information view then elaborates on the flow of information through the system, highlighting how components create, communicate and consume information during envisioned system operation to deliver on the use cases' goals. Finally, the deployment view presents topological considerations in terms of defining the execution environment for the various system components at a later stage during the project lifetime.

The contents of this deliverable are provided as a first version documentation of the envisioned system's shape and structure, and are expected to be updated upon completion of the architecture and system specification activities in M18 of the project lifetime.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

1 Introduction

The current deliverable represents the OPTIMAI's consortium preliminary attempt at defining a modular, service-oriented architecture for the OPTIMAI proposed solutions. The document aims to serve as a reference guide for the consortium partners who are engaged in the technical implementation of the project, and are responsible for the development of the components that will comprise the architecture.

The components, information flows and topological considerations described in this deliverable are the result of a combination of activities undertaken in the context of Task 2.3 "System specifications and architecture" during the first 12 months of the project lifetime. These activities supersede the results of the initial architecture definition (during the preparation stage of the project), and encapsulate, among exploratory and design activities (both top-down and bottom-up) also the results of online workshops during which the main components identified to form the architecture were elaborated and agreed upon by all partners. To this end, the functional and non-functional requirements resulting from Task T2.1 "Consolidation of user and ethics and legal requirements", along with the use cases defined in the framework of Task T 2.4 "Use cases definition" have been considered. Specific care has been taken to align the architectural components to the guidelines and perspectives dictated by international standardisation activities and reference frameworks, while procedures for compliance to ethical requirements (resulting from the work conducted under Work Package 9) have also been taken into account.

Since requirements, as well as parts of the architecture are subject to change in the remainder of the Task, this deliverable should hence be treated as an initial reference, one that particularly does not imply specific implementation details, but rather describes the foreseen building blocks, their relationships and their development over time. As a result of these ongoing activities, an updated version of this deliverable ("The OPTIMAI Architecture specifications - 2nd version" – D2.5) will be submitted halfway through the project (M18).

The present document is structured as follows:

- Section 2 complements the contents of D2.3 "State of the art survey" with a focus specifically placed on reference architecture models that are defined in international standardisation initiatives as well as by the academic community, and which have been used as a guide for the design of the OPTIMAI architecture.
- Section 3 briefly presents an overview of the steps followed for the definition of the architecture functional components and their interdependencies.
- Section 4 describes the functional viewpoint of the OPTIMAI architecture, presenting a detailed description of the architectural components and subsystems, and offering insight into the mappings drawn to the most prominent reference architecture models and guiding principles.
- Section 5 presents the information viewpoint of the OPTIMAI architecture, elaborating on the way information flows through the architectural components (presented at both a

high-level overview and further targeted at indicative scenarios described in the identified OPTIMAI use cases).

- Section 6 presents a preliminary overview of the OPTIMAI solution's deployment over physical and virtualised resources, drafting a first topological map based on the smart factory framework, which it extends to accommodate the foreseen edge computing capabilities.

Finally, in Section 7, the key points of the document are summarized.

1.1 Mapping of project outputs

The purpose of this section is to map OPTIMAI Grant Agreement (GA) commitments, both within the formal Deliverable and Task description, against the project's respective outputs and work performed. This mapping is presented in Table 1, below.

Table 1: Adherence to OPTIMAI's GA Deliverable & Tasks Descriptions

OPTIMAI Task	Respective Document Chapters	Justification
T2.3: System specifications and architecture *"In this scope, the definition of specifications for the various components of the system and the whole system as an entity will be attempted, so as to fulfil the existing and future demands of stakeholders".*	Section 4 - OPTIMAI Architecture - Functional view	The detailed description of all identified functional blocks and subsystems underpinning the OPTIMAI stakeholders' requirements are described in Section 4 of this document.
T2.3: System specifications and architecture *"Based on the analysis of user needs and scenarios, conducted in T2.1 and T2.4, system specifications will be defined according to existing standards activities in order to address interoperability and security requirements for each module".*	Section 2 - Reference Architecture models	A comprehensive overview of standards and industry-led reference architectural models is provided in Section 2.
	Section 4.2 - OPTIMAI alignment to standards-led reference architectures	Section 4.2 details the mappings and parallels drawn between the OPTIMAI functional architecture perspective and two leading standardisation initiatives referring to the system as both an In-

		dustry 4.0 solution and an Industrial Internet of Things deployment.
T2.3: System specifications and architecture *"In addition, this task will analyse the requirements and recommendations of the OPTIMAI solution resulted from WP9 regarding the ethics and regulatory framework in order to define an architecture that reflects the concept of "security and privacy by design"".*	Section 3.3 - Ethical compliance – Compliance through Design	Following the description of the architecture definition approach taken in the first year of the project, this Section presents key principles followed to adhere to an Ethics by Design and Legal Compliance through Design approach.
T2.3: System specifications and architecture *"Effort will be put system modularity so as to address diversity in equipment and resources, thus allowing for future changes, updates and upgrades."*	Section 6 - OPTIMAI Architecture - Deployment view	A preliminary description of the foreseen tangible infrastructures necessary for supporting deployment of the system components in service to the project use cases is described in Section 6.
T2.3: System specifications and architecture *"The design of every architectural module will have to take into account all relevant and important elements like system requirements, risk factors, software issues, communication elements, safety issues, hardware requirements and specific application requirements".*	Section 3 - Architecture design approach	This Section covers the approach followed for defining the version of the architecture described in this document, detailing how the design came to be as a result of identifying and elaborating components based on specific functional requirements, as well as those defined from an ethical and legal perspective.
	Section 4 - OPTIMAI Architecture - Functional view	This Section describes the responsibilities, roles and foreseen dependencies of the OPTIMAI functional blocks,

		driven by the elicited functional and non-functional requirements. The perspective of standards' bodies with respect to the OPTIMAI use cases is also taken into consideration.
	Section 5 - OPTIMAI Architecture - Information view	This Section describes the information flow within OPTIMAI, with a specific aim at demonstrating how information flows through the system for specific tasks identified in the project use cases.
	Section 6 - OPTIMAI Architecture - Deployment view	This Section introduces a preliminary network topology for the smart factory solution proposed, with implications on hardware and software requirements, along with a view on the necessary applications' end-user terminals to target.

2 Reference Architecture models

Industry 4.0 (I4.0) represents a huge leap in terms of how manufacturing systems have been viewed, primarily because of the infiltration of information and communication technologies (ICT) into almost every aspect of the manufacturing system's production [1]. At the heart of I4.0 lies the so-called Cyber-Physical Production System (CPPS), a heterarchical architecture of inter-communicating resources [2] vastly different from the typical hierarchical infrastructure associated with Dedicated Manufacturing Lines (DML) and Flexible Manufacturing Systems (FMS) of the Third Industrial revolution (Industry 3.0). This work, in conjunction with the rise of Internet of Things (IoT) and cloud computing technologies over the past decade, have significantly propelled the envisioned end-to-end communication paradigm that underpins the need for all production-relevant assets to intercommunicate within the 4th Industrial revolution concept. Despite its clear vision however, I4.0 remains a challenging concept for manufacturing industries to realize [3].

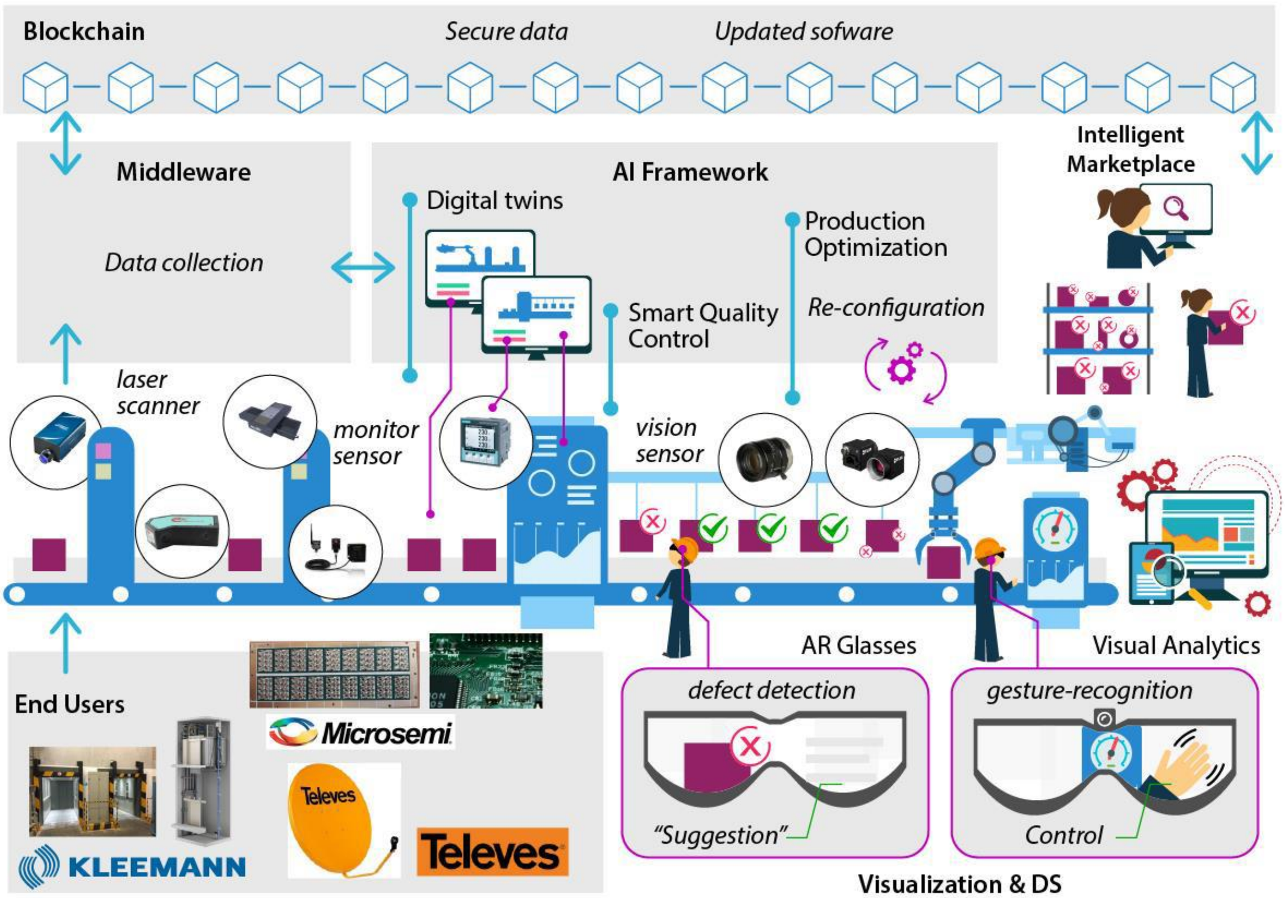

Figure 1: OPTIMAI overall concept (Source: OPTIMAI GA).

ICT solutions (multimodal sensor and data acquisition network; artificial intelligence tools; augmented reality; etc.) towards achieving optimal production conditions for the three selected pilot sites, it is important to identify the reference architectural tools, platforms and technologies that support the OPTIMAI overall vision and objectives, and contextualise those architectures to fit the requirements, as well as support the implementation of the presented use cases. Therefore, in this section we will present the most important reference architecture models intended for I4.0, and will highlight which of their features and characteristics best underpin the envisioned OPTIMAI overall concept (Figure 1). Through this exercise we will derive a customised, use-case-specific functional architecture for OPTIMAI by means of a top-down specification process that takes into account: (i) the conceptual specification described in the OPTIMAI GA (Section 2.3); and (ii) facilitating and maintaining alignment between the reported OPTIMAI architecture and the most appropriate reference model, to substantiate an I4.0-compliant approach. The specific methodology for top-down specification, as well as the ensuing bottom-up refinement will be described in detail in Section 3. This Section will instead be organised as follows:

- **Section 2.1** will present a formal definition of concepts and conventions that will be used throughout the document.
- **Section 2.2** will list the key I4.0 reference architecture models reported in the current literature and provide an overview of their main characteristics and features.
- **Section 2.3** will present the reference implementation model for the OPTIMAI project, derive from the GA specification. This section will specify how OPTIMAI relates to the RA models presented, electing the foremost alignment by highlighting the aspects in which our architecture shall enforce compatibility with the selected I4.0 framework.

2.1 Definitions and conventions used

Already in this document, several key concepts have been established, yet it is important to thoroughly define each one (along with all future concepts and conventions to be used) prior to going into more detail on architecture definition processes and reference implementations. These concepts are highlighted in the following Table (Table 2):

Table 2: General concepts definition.

Concept	Definition
Architecture	An architecture can be defined as the logical and functional structure underpinning implementation of an ICT system. The structure includes components, their main features, functions and how they interconnect.
Reference Architecture	A Reference Architecture (RA) refers to a set of standard principles that can be taken into account during the architectural design of a system. These principles can serve as a starting point, because they offer useful information with regard to the key concepts that an architecture under definition for a specific field should uphold. In addition, a RA provides insight into the basic interactions that should take place within the system. Hence, ICT developers can derive and specify concrete solution architectures for addressing a particular topic, by applying the generic rules and relationships dictated by the RA.

View (or viewpoint)	An architectural view (or viewpoint) represents a representation of the system constructed to accommodate a particular perspective (e.g., shape and structure – *Functional*; handling of information – *Information*; system execution environment - *Deployment*, etc.).

2.2 Reference architectures for Industry 4.0

2.2.1 Standards- and Industry-led Reference Architectures

2.2.1.1 Reference Architectural Model Industry 4.0

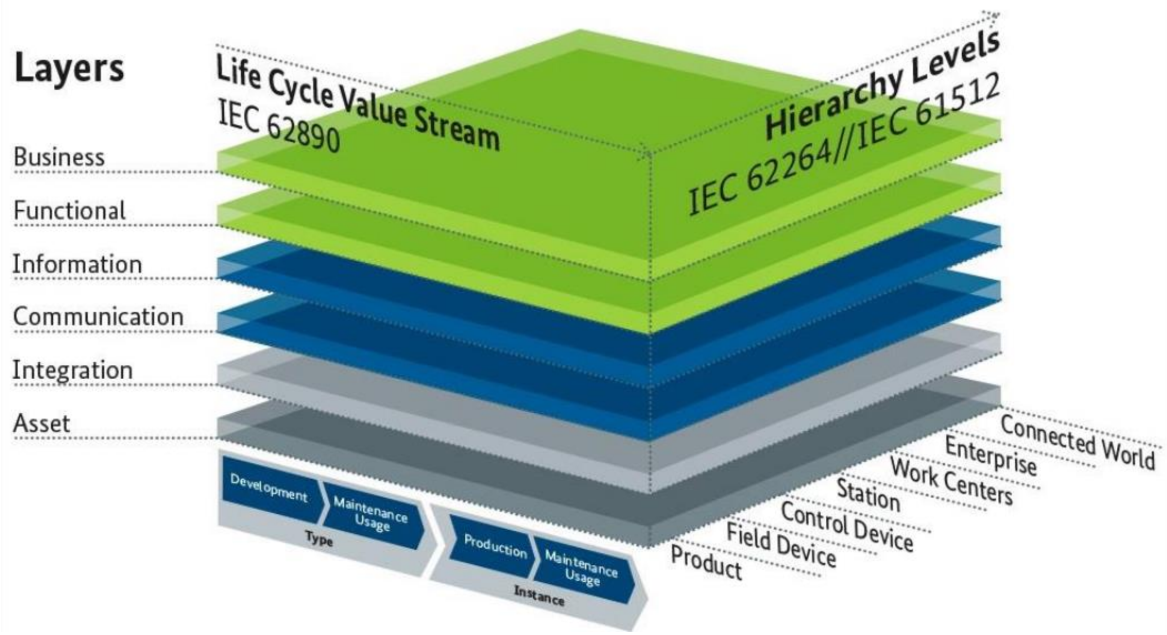

Figure 2: The RAMI 4.0 Layered 3D Architecture. Graphic © Plattform Industrie 4.0 and ZVEI, retrieved from [4]

The Reference Architectural Model Industrie (RAMI) 4.0 is among the first (2015) frameworks proposed for defining a common picture of I4.0, aimed primarily toward providing standards, common terms and rules for describing requirements and structures regarding different smart factory use cases. RAMI 4.0 is a European (German) initiative toward standardisation. As is shown in Figure 2, RAMI 4.0 breaks down the most critical elements found in the Industry 4.0 environment into a three-dimensional coordinate system, whose axes correspond to: (i) the hierarchy levels of a connected manufacturing system; (ii) lifecycle of systems and products, including both development and maintenance; and (iii) different perspectives (or I4.0 components [5]) in a smart factory. These axes are described in more detail in Table 3 below:

Table 3: RAMI 4.0 axes/dimensions.

Axis	Description
Hierarchy Levels	Represents the layers of automation that can be found in a typical factory environment. Often represented as a pyramid, this dimension of RAMI 4.0 is composed of the following layers (from top to bottom): (i) **'Product'**,

	referring to the smart products being manufactured, and hence can communicate with the production system [6]; (ii) **'Field device'**, referring to devices (e.g., sensors and actuators) used to control factory machinery; (iii) **'Control device'**, which includes devices that calculate optimal process parameters and thus trigger processes in the actuators to the layer below; (iv) **'Station'** refers to Supervisory Control and Data Acquisition (SCADA), usually occurring from one integrated location, by means of a User Interface (UI) or Human-Machine Interface (HMI); (v) **'Work Centers'** refers to Manufacturing Execution Systems (MES), which deal with collection and storage, analysis and dispatch of data, along with tracking of the process of turning raw materials into manufactured goods; (vi) **'Enterprise'** refers to Enterprise Resource Planning (ERP), dealing with the data-driven management and decision-making regarding the entire factory; and (vii) **'Connected World'** refers to external to the factory services provisioned through the Internet [5].
Life Cycle & Value Stream	This axis refers to the lifespan of an object that is relevant to the manufacturing environment (e.g., a product, a machine, documentation, etc.). Within the RAMI 4.0, this lifespan is broken down into two phases, i.e., **'Type'** and **'Instance'**. When the object is in the 'Type' Phase, it is considered to be in planning, design and pre-production. While in this Phase, the object's specification is being firstly formulated during *'Development'*, while any design iterations and/or updates are covered in the *'Maintenance Usage'* state. Once the object enters the 'Instance' Phase it has moved into a *'Production'* state. The *'Maintenance Usage'* state for this Phase refers to customer service processes, i.e., services offered for the particular instance of the object (such as maintenance/repairs).
Layers	This axis is used to describe the different I4.0 elements present in a typical smart factory. From the bottom to the top, these are: (i) the **'Asset'** layer, which describes physical resources found on the factory shop floor, e.g., equipment, parts, etc.; (ii) the **'Integration'** layer refers to the means of communication between the Information and Operation Technologies (OT); (iii) the **'Communication'** layer refers to the type of communication protocols and channels used for standards-compliant data transmission; (iv) the **'Information'** layer refers to the means by which information (data) is stored; (v) the **'Functional'** layer refers to decision-making processes for establishing the functions that factory assets should perform; and (vi) the **'Business'** layer refers to organisational and business processes.

RAMI 4.0 is seen as the most mature European-led initiative with strong ties to standardisation initiatives, therefore serves as the greatest influence for guiding the design specification of the

OPTIMAI architecture. Hence, the proposed architecture is foremost aligned to this RA, enabling OPTIMAI system architects the freedom to establish the necessary technological functional blocks in a specific layer-and-hierarchy RAMI 4.0 layer slice, by a mapping of the OPTIMAI use cases (UCs) to the Life Cycle & Value Stream axis. In this respect, the functional components of the OPTIMAI architecture can be analysed and described in accordance to the RAMI 4.0 Layers and Hierarchy axes in a much more simplified 2D representation. This exercise will be more thoroughly explored after the description of the OPTIMAI functional architecture viewpoint in Section 4.2.1.

2.2.1.2 Industrial Internet Reference Architecture

The Industrial Internet Reference Architecture (IIRA) [7] is a US-led initiative toward establishing a common reference frame for the development of Industrial Internet of Things (IIoT) solutions targeted at modern smart factories in different industrial verticals. The framework has been developed by the Industrial Internet Consortium (IIC)[1] in parallel to RAMI 4.0 (first released in 2015), representing a standardisation initiative undertaken on the other side of the Atlantic. The document is continuously updated ("living"), with the ninth version (v1.9) being the last one released. The purpose of the IIRA is to provide guidance for the specification of IIoT system, solution and application architectures [7].

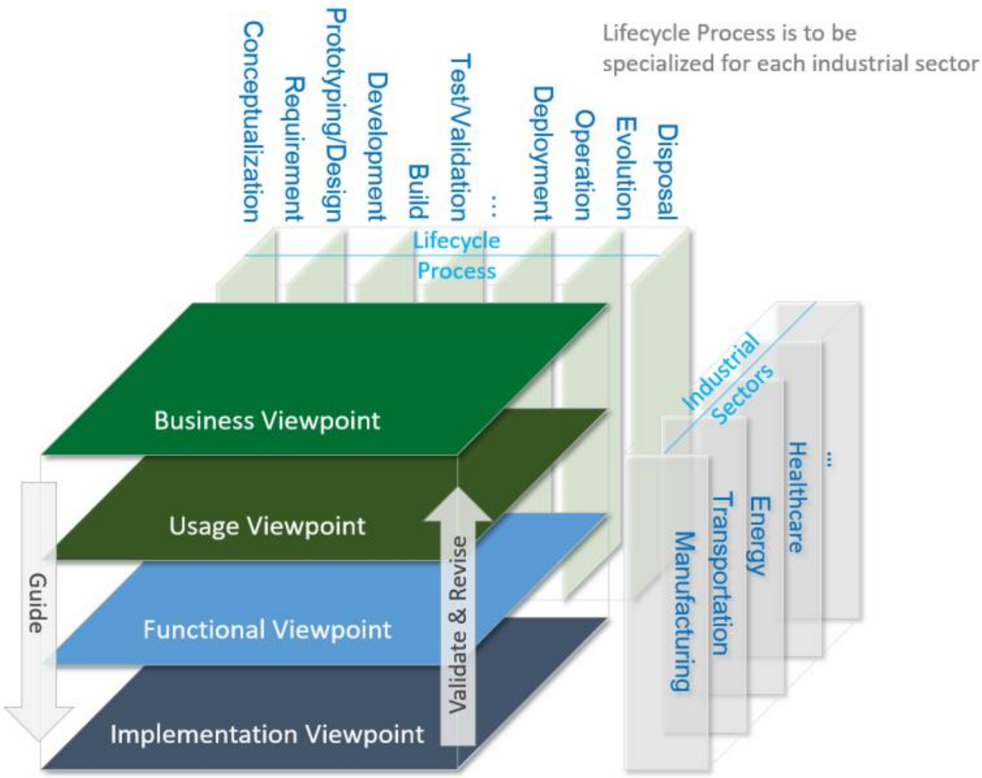

Figure 3: The IIRA Viewpoints-based model framework. Source: [7]

[1] https://www.iiconsortium.org/IIRA.htm

Similarly, to RAMI4.0, the IIRA implies no specific implementation, rather organising IIoT use cases under 4 layers (called 'Viewpoints'), which categorise aspects that the system should address in each use case. The graphical depiction of the IIRA is presented in Figure 3. Each viewpoint is used for analysing specific needs and requirements, with decisions made in a particular viewpoint guiding the way requirements and needs are defined in the viewpoint below. Similarly, the eventual implementation suggested in the lower viewpoints serve as a means to validate ad provide feedback to the top viewpoints, for potentially introducing revisions, starting at the business level. Apart from framing factory operations with respect to IIoT on these four viewpoints lone, the IIRA provides extensibility by means of custom viewpoints that can be defined by system architects following the IIRA model, in order to fit the specific requirements of their own IIoT use case. The four viewpoints that make up the core of the IIRA are organised as follows in Table 4:

Table 4: IIRA defined viewpoints according to the IIRA specification v1.9 [7].

Viewpoint	Definition
Business	This viewpoint is related to decision-making at the business level, where the specific business goals are aligned to system capabilities to deliver on the IIoT vision for a specific factory. Concerns that are addressed in this viewpoint regard e.g., return on investment (ROI), maintenance costs, etc. Decisions in this viewpoint will impact designs in the 'Usage; viewpoint below.
Usage	This viewpoint relates to the expected usage of the system based on the requirements identified in the viewpoint on top, and deals with the way that the IIoT system is built and structured, relating to the activities of its components (i.e., both human and system resources). It basically suggests the use cases (in terms of tasks undertaken by an entity, or "role" in the system) that will guide the way the functional viewpoint components directly below should be designed.
Functional	The functional viewpoint is used to specify the functional components that comprise the overall IIoT system. In addition, it defines the interactions, roles and responsibilities of each one, basically forming the backbone of the system architecture and guiding the activities defined in the Implementation viewpoint below.
Implementation	The implementation viewpoint describes the technology necessary to implement the functional blocks identified in the Functional viewpoint. It specifies elements of system implementation and integration concerns, such as the connectivity communication protocols, deployment considerations, etc.

From the perspective of delivering a uniform interconnected platform, OPTIMAI aligns to a substantial amount to the IIRA, given the similarities and interoperability that the IIRA shares with RAMI 4.0 [8], and both architectures' support for service-oriented architectures (SOAs), such as the one envisioned by the OPTIMAI project. Essentially, the OPTIMAI functional architecture specified in Section 4 can be analysed from a viewpoints-based perspective as indicated by the IIRA, since its values and objectives across the three pilot sites can be summarised in accordance to the IIRA's Business viewpoint guidance. Given the focus of OPTIMAI in delivering a system tightly coupled to a specific phase of the product lifecycle, RAMI 4.0 is considered a superior candidate RA to which the project architecture should be parallelised to. However, the strong correlation of the functional perspectives described in both RAs suggests the capacity to extend the presented architecture to fulfil viewpoint-based requirements following the mapping described in [8]. Key takeaways from such an alignment are highlighted in Section 4.2.2.

2.2.1.3 Industrial Value Chain Reference Architecture

The Industrial Value Chain Reference Architecture (IVRA) is a Japanese initiative [3] towards providing a framework for describing commonalities in smart manufacturing. At its core, this RA is based in the concept of a Smart Manufacturing Unit (SMU), autonomous elements that connect and communicate to deliver the integrated functionality of a smart manufacturing system [9]. Each SMU is defined by means of a three-dimensions viewpoint-based coordinate system, whose components are specified in Table 5, below:

Table 5: IVRA Smart Manufacturing Unity views coordinate system.

View	Definition
Asset	Describes four distinct categories of assets that make up the manufacturing environment: (i) *'Personnel'* (i.e., human resources); (ii) *'Process'* (referring to knowledge about production procedures); (iii) *'Product'* (encapsulating both materials required to manufacture goods, as well as the final goods themselves); and (iv) *'Plant'* (referring to all physical assets on the shop floor, such as equipment, tools, machinery etc.).
Management	Reflects to the four key elements that drive managerial decision-making, i.e., (i) *'Quality'* (an indicator of the fitness of the SMU to fulfil a specific requirement); (ii) *'Cost'* (the amount of money spent for the SMU to provide its functions); (iii) *'Delivery accuracy'* (an indicator of the SMU in meeting specific time constraints set by a customer); and *'Environment'* (referring to the impact the SMU has on the environment, e.g., energy consumption, waste creation etc.).
Activity	Describes the activities undertaken by the SMU as framed in the Plan-Do-Check-Act cycle: (i) *'Plan'* refers to the activity of drafting action items towards reaching a specified goal; (ii) *'Do'* refers to the undertaking of planned activities; (iii) *'Check'* refers to the evaluation of exercised activities with respect to their set goals; and (iv) *'Action'* refers to the activities undertaken to improve the SMU's overall function based on the findings of the previous step In the cycle.

Using the SMU as a 3D building block, the IVRA structures its proposed reference architectural model by defining SMUs in a three-dimensional sandbox defined by the axes described in Table 6. The RA is shown in Figure 4.

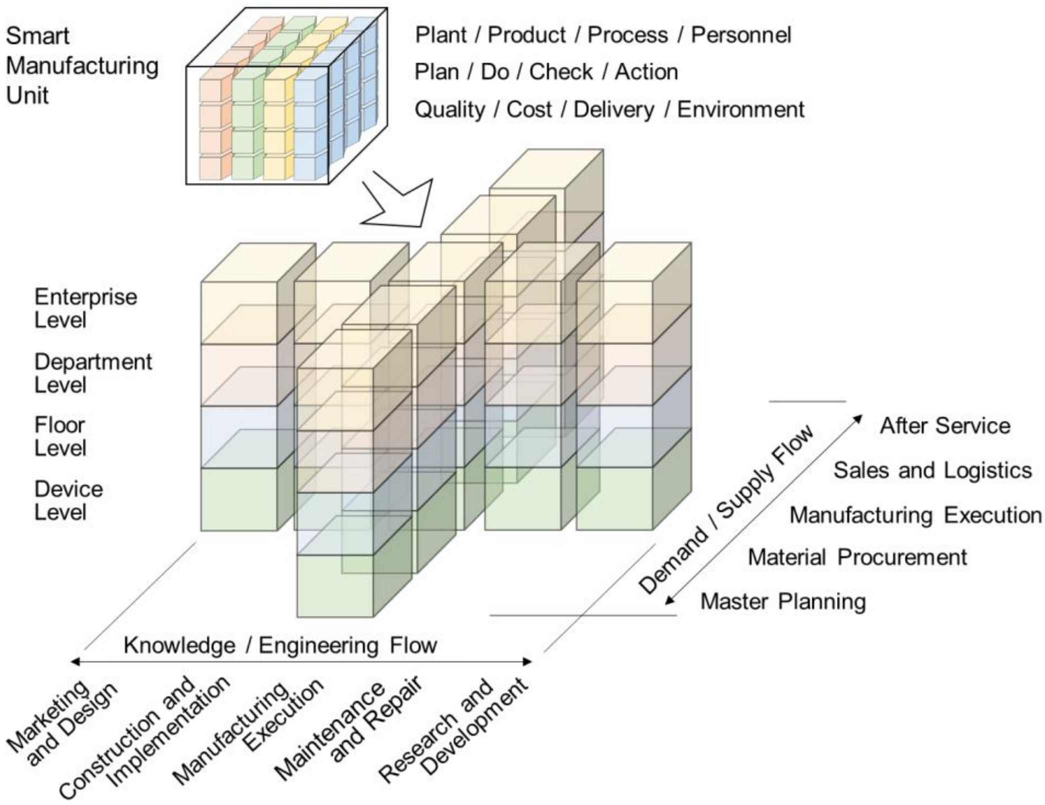

Figure 4: The IVRA SMU-based Reference Architecture. Source: [9]

Table 6: IVRA block-based RA components.

View	Definition
Knowledge and Engineering flow	Knowledge and Engineering Flow refers to general information regarding various functions in smart manufacturing. This axis can be classified into the following phases: (i) *'Marketing and design'* (the process of identifying market needs and designing a product based on these needs); (ii) *'Construction and implementation'* (includes changes and improvements to the production system to ensure efficient production); (iii) *'Manufacturing execution'* (regards the application of knowledge and engineering knowledge toward manufacturing a product - serves as a point of intersection with the Demand/Supply Flow as it also reflects the application of knowledge from various stages of the product lifecycle, such as planning and feedback); (iv) *'Maintenance and repair'* (in this phase actions are taken to ensure the proper functioning of the equipment and the employee efficiency); and (v) *'Research and development'* (refers to the deployment of technologies which will contribute to addressing issues).

Demand and Supply flow	This axis relates to the manufactured product lifecycle. It includes the following elements: (i) *'Master planning'* (refers to the development of plans which ensure that there is alignment of supply and demand); (ii) *'Material procurement'* (is concerned with developing partnerships with the proper suppliers); (iii) *'Manufacturing execution'* (refers to following instructions for the manufacturing of products and offering feedback to the departments); (iv) *'Sales and logistics'* (refers to estimating as accurately as possible the number of products that should be made available on the market as well as expenses connected to their transfer); and (v) *'After service'* (refers to customer service)
Hierarchical Levels	Includes the vertical levels of an enterprise which are the following: (i) *'Enterprise level'* (here the person in authority can monitor the operations within the enterprise); (ii) *'Department level'* (units of an enterprise responsible for a specific function); (iii) *'Floor level'* (the level where people work on machines and manufacture products); (iv) *'Equipment level'* (here the equipment is operated by control modules such as sensors and actuators).

Definition of the OPTIMAI functional architecture under the IVRA framework can be possible by following a functional decomposition of the use cases into the specific components described in Section 4.1, each assuming the role of a SMU with a well-defined "Plan-Do-Check-Act" cycle. However, as the IVRA hints towards departing a standards-compliant approach to defining interconnections between SMUs to build self-evolving platforms under the concept of Loosely Defined Standards (LDS), along with the general complexity introduced by the specification of the platform under the perspective of a system-of-systems (SoS – where an SMU can be either a component or a SoS in itself), the exercise of aligning OPTIMAI to the IVRA is deemed outside the scope of the present deliverable and Task activity.

2.2.1.4 FIWARE Reference Architecture for Smart Industry

The FIWARE4Industry (F4I) RA comprises a reference frame for smart factories based on the FIWARE open source framework for interoperable smart solutions[2]. It is the culmination of the FIWARE4Industry multi-project initiative[3] toward combining a total of 29 FIWARE-compliant Generic Enablers (GEs) toward defining FIWARE-enabled reference implementations for smart, digital and virtual factories, as well as IIoT systems. Through this exercise, the FIWARE Foundation has specified a RA for developing Smart Solutions in the Industrial domain, summarised in Figure 5.

The OPTIMAI overall context specifies integration routines between real and digital smart factory environments that are interconnected by means of IoT-enabled technology (sensors and actuators). Support is foreseen for a construct similar to the FIWARE Context Broker, which will be

[2] https://www.fiware.org/
[3] https://www.fiware4industry.com/

centralised hub dealing with data storage and information exchange responsibilities in the form of a middleware solution responsible for allowing the communication between the IoT agents and the smart factory platform. In this respect, parallels can be drawn between the OPTIMAI architecture and the FIWARE RA. However, despite presence of some compatible elements in its description, the OPTIMAI platform will not be based on FIWARE GEs, therefore departing from the fundamental aspect underpinning this RA.

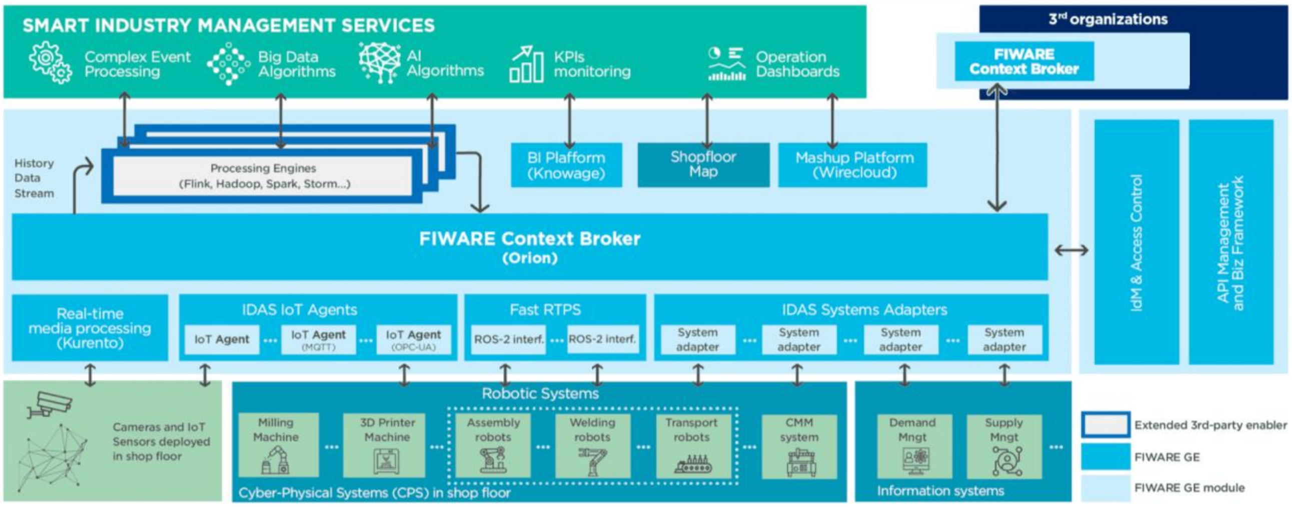

Figure 5: Reference architecture for IoT-enabled smart Industry solutions based on FIWARE Open Source technology. Source: FIWARE Foundation[2].

2.2.1.5 IBM Industry 4.0

One of the more recent (2020)[4] RAs for Industry 4.0 solutions has emerged from a multi-partner collaboration initiative led by technology giant IBM and its subsidiary open source specialisation company Red Hat. The RA is based on three layers, namely the 'Edge', 'Plant' and 'Enterprise'. The diagram for this RA is provided in [10]. IBM provisions the RA from its corporate website, mapping building blocks to its portfolio of commercially distributed components and systems, thus inviting stakeholders to build concrete solution architectures for various use cases using IBM and its subsidiaries' technology. As such, further analysis of this RA is deemed beyond the scope of the present deliverable.

2.2.2 Notable reference architectures in the scientific literature

The following RAs have been reported in the scientific literature as candidate RAs proposed by academic institutions, as alternatives to the ones reported in the sections above. These architectures are supported by a handful of industrial organisations, as a result of collaboration projects between tightly-knit consortia of industrials and academics [3]. Their inclusion in this Section is based on the description of several unique advantages and characteristics they support when compared to major RAs, yet a significant body of work is needed to align OPTIMAI concepts and use cases to the specifics of these experimental models. Hence, the following sections are provided as a means to identify potential relevancy of OPTIMAI foreseen processes with reported

[4] https://www.ibm.com/cloud/architecture/architectures/industry-40/reference-architecture

advancements over current standards initiatives, which would be interesting in the scope of seeking standards contribution opportunities for OPTIMAI itself in the context of Task 8.3.

2.2.2.1 Stuttgart IT-Architecture for Manufacturing

The Stuttgart IT Architecture for Manufacturing (SITAM) was proposed by Kassner et al in 2016 [11] to address flexible integration of IT systems, human-centric manufacturing processes and advanced analytics for data-driven factories. Contrary to the majority of industrial- and standards-led RA initiatives, SITAM is two-dimensional and is based on a composition of three middleware solutions (following a SOA approach) that together form services for creating value for both human operators and machines (provisioned through a desktop/mobile app marketplace). The three middlewares are described in Table 7:

Table 7: SITAM service-oriented middlewares

Service block	Description
Integration Middleware	Incorporates services that are specific to the monitoring and management of physical resources and processes, including data exchanges between OT and ICT structures.
Analytics Middleware	Incorporates services for descriptive, predictive and prescriptive analytics to implement the data-driven factory vision, based on the data obtained from the Integration middleware. It houses also the manufacturing knowledge repository and deals with information mining and KPI management.
Mobile Middleware	Incorporates services that enable the development of mobile-based applications (e.g., smartphone/tablet/wearable apps) for the data-driven factory worker.

As envisioned from the project preparation stage and specified in the GA, OPTIMAI bears strong similarities to SITAM in that it utilises a SOA of its own that makes use of a middleware solution to connect IoT infrastructure with Artificial Intelligence (AI) pipelines meant to both support and complement the human worker toward a concrete human-in-the-loop (HITL) implementation for smart manufacturing. SITAM designers purport that the RA described in [11] provides significant advantages over the likes of RAMI 4.0, IIRA, etc., primarily because it introduces the HITL paradigm more concretely in the smart factory systems pipeline. Therefore, similar advantages can be expected to be gained in the OPTIMAI implementation. As the OPTIMAI architecture aligns to RAMI 4.0 more profoundly, potential contributions to the relevant standards bodies, e.g., the Standardization Council Industrie 4.0 (SCI 4.0)[5], the International Electrotechnical Commission

[5] https://www.sci40.com/english/

(IEC)[6] and the International Organization for Standardization (ISO) can be monitored over the lifetime of the project.

2.2.2.2 LASIM Smart Factory

The LASIM Smart Factory (LASFA) RA [12] is a model described by members of the Laboratory for handling, assembly and pneumatics (LASIM) of the Faculty of Mechanical Engineering of the University of Ljubljana, Slovenia[7]. Its purpose is to assist in the planning of smart factory environments by adapting the RAMI 4.0 cubic model into a two-dimensional representation aimed at clarifying abstract concepts present in the current RAMI 4.0 incarnation. LASFA's core advantage compared to RAMI 4.0, as purported by its authors, is its clear identification of the correlation between a production hall (i.e., one or more production lines/cells, warehouses, manual workplaces) and its *digital twins* (a virtual representation of the real system in digital space - DT). A DT can be defined for each element of the production hall, as well as the production hall itself, each managed by a digital agent and a local cloud. The DT is primarily used to power visualisation applications that enable human operators to monitor and provide feedback on manufacturing processes at various stages of the production described by the RAMI 4.0 Layers axis (from 'Asset', i.e., managing factory resources, to 'Business', i.e., strategic decision-making at organisational level).

Digital twinning and visualisation are key elements in the OPTIMAI reference implementation approach, thus pinpointing a high correlation between the LASFA placement proposition for DTs and digital agents and the ones defined for OPTIMAI processes. As LASFA can be considered more an elaboration of RAMI 4.0 for a particular use case rather than a uniform representation toward a common understanding (as is OPTIMAI), OPTIMAI can capitalise on LASFA toward defining proper placement of its own functional elements, while maintaining alignment to the RAMI 4.0 common perspective and leveraging on potential advantages derived from the more concrete implementation that such an elaboration will yield.

2.3 OPTIMAI reference implementation model

The OPTIMAI preliminary framework for implementation was based on the conceptual architecture which is depicted in Figure 6. This model represents the results of a preliminary exercise in the architecture design approach described in Section 3, which was carried out as early as the project preparation period (prior to the signing of the GA, when OPTIMAI was conceptualised as a proposal for the DT-FOF-11-2020 H2020 Call Topic). As a result, a thorough reference implementation of the OPTIMAI smart manufacturing solution has been formulated in the form of a conceptual model, and is elaborated in detail in Annex 1 (part A) of the OPTIMAI GA (Description of Action – DoA).

The core technological functional blocks that comprise this early conceptual architecture for the OPTIMAI solution have since been refined, and will be described in more detail in Section 4. Therefore, the contents described in the GA and shown in Figure 6 are treated in the context of

[6] https://www.iec.ch/homepage
[7] https://web.fs.uni-lj.si/lasim/

this deliverable as a project-specific RA, which underpins the vision and objectives of the project and its use cases. This RA was meant to be used as guidelines for identifying the necessary functional components, and has been updated by means of a second iteration of the processes described in the following Section. In addition, functional blocks and solutions were aligned to prominent RAs for I4.0 (as part of the technology exploration step) to ensure that OPTIMAI can be understood, and shares a common vision with other stakeholders in the I4.0 initiative.

For all future project purposes, the OPTIMAI architecture as defined in the present document takes precedence over the definition specified in the GA.

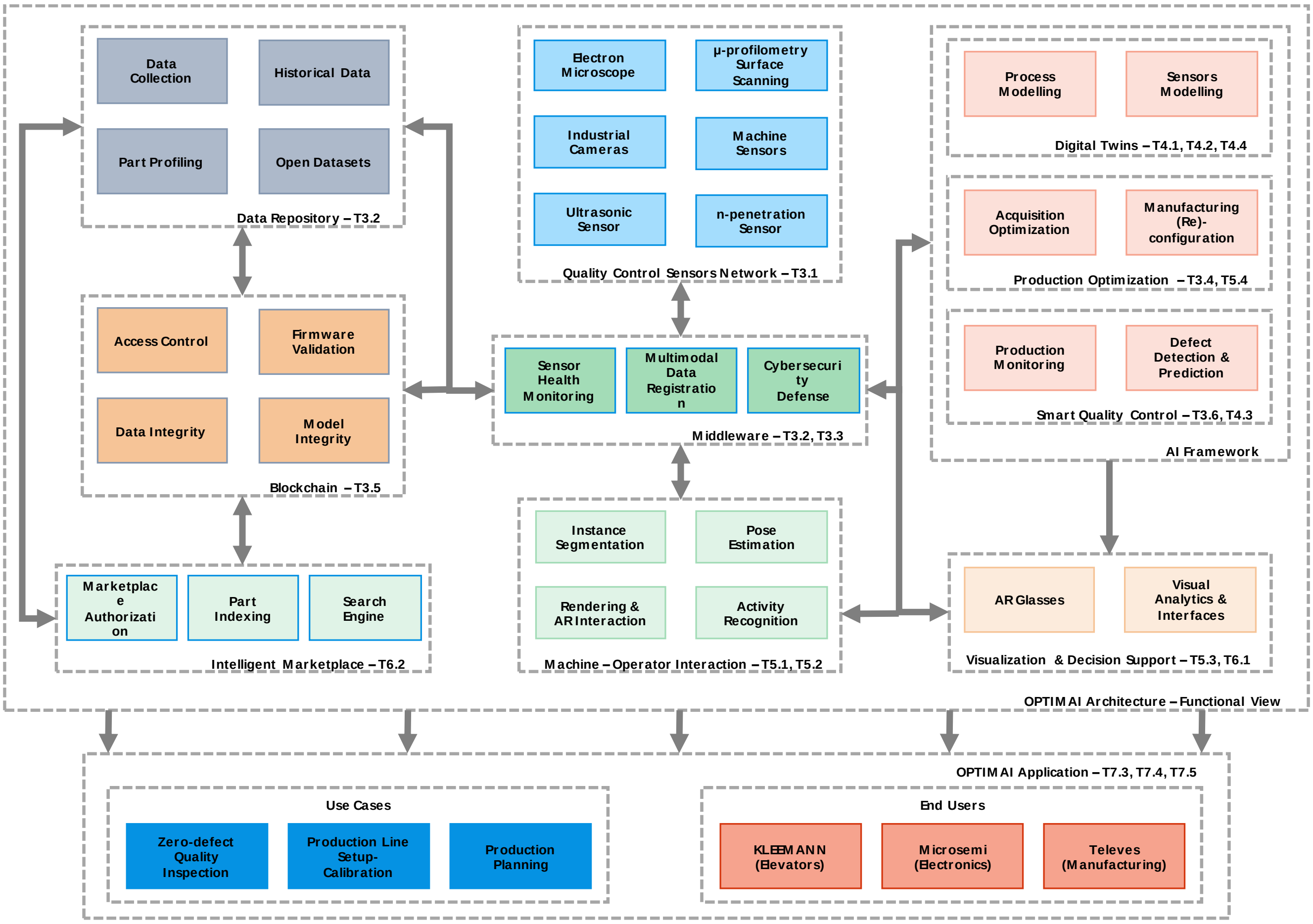

Figure 6: OPTIMAI conceptual architecture (deprecated). Source: GA.

3 Architecture design approach

In this section the approach followed in the framework of Task T2.3 “System specifications and architecture” will be presented with the aim of specifying the reported initial version of the OPTIMAI architecture. Two iterations of this procedure (M1-M12, including the project preparation period, and M12-M18) will take place over the course of the project lifetime. The iterations will revolve around the improvement and development of the architecture with the aim of meeting the identified user requirements (Deliverable D2.1 “User and ethics and legal requirements - 1st version”) and pivotal aspects of the project use cases (UCs, Deliverable D2.6 “OPTIMAI use cases definition”). The approach involves three steps: i) technology exploration; 2) top-down design; and iii) bottom-up refinement. The sequence of implemented actions over the course of a single iteration of the architecture design approach is depicted in Figure 7.

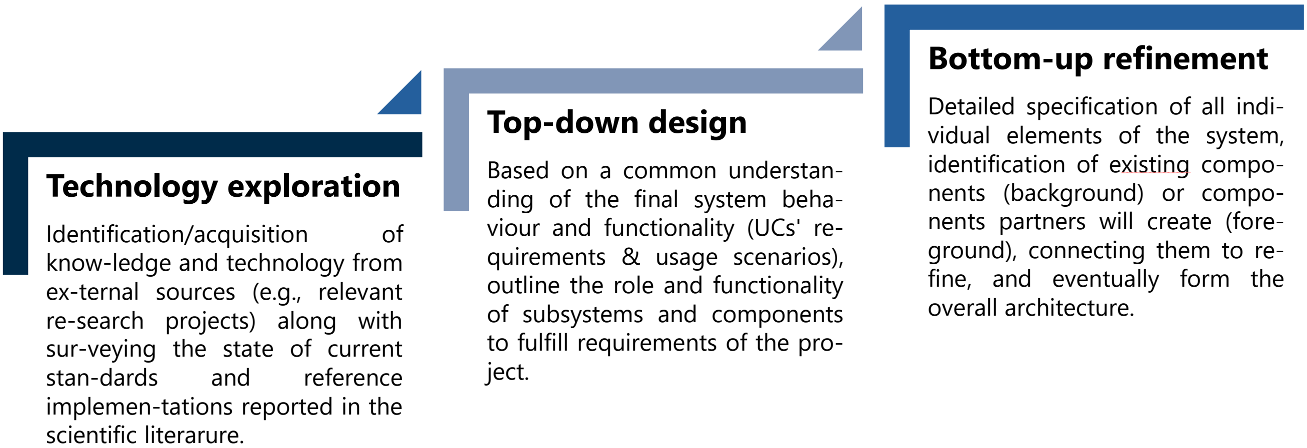

Figure 7: Architectural design approach within OPTIMAI

Technology exploration is identified as the first step in the architecture definition process. It encompasses a procedure where technologies and architectural models relevant to the OPTIMAI goals and vision are identified and studied, in order to generate inputs from relevant results reported in the frameworks of other national and international (preferably, EU-funded) Research & Development (R&D) activities, and distil them into the OPTIMAI architecture design. An early technology exploration process was conducted during the project preparation phase, while a more in-depth exercise was undertaken during Months 1-6 of the OPTIMAI project in the context of Task 2.2: “State of the art analysis, existing and past research initiatives”, whose results are reported in deliverable D2.3: “State of the art survey”. The process was complemented with the survey of RA models presented in Section 2 of this deliverable. Hence, only the activities carried out in the context of the top-down design and bottom-up refinement steps of the process will be described in the following sections.

3.1 Top-down design

Top-down design is the process of dividing the system into the elements it is composed of, leading to the identification of any-level subsystems and base components. The purpose of this phase is to gain an understanding of the main function of the components and how they will communicate with each other. A preliminary top-down design process was performed during the proposal preparation stage, resulting in an initial, conceptual specification of the OPTIMAI architecture reported in the GA and discussed in the previous Section (Figure 6).

During the top-down phase (a process taking place between M1-M6), and using the architecture specification in the GA as a reference model, a total of 36 **components** (functional blocks that cannot be divided into smaller elements) were identified for the OPTIMAI I4.0 solution. The components were abstractly described and presented to project partners as needing further discussion in the subsequent bottom-up phase (see next Section). The 36 components as emerged from the architecture specification procedure for the elapsed period are shown in Table 8.

The Table elements presented in bold constitute the larger **subsystems** composed of the smaller, individual components. A component, identified as an entity that cannot be further broken down, is assigned a component serial number from 001-036. During the first six months of the project, a 1st architecture workshop teleconference was held during which the top-down approach was implemented with inputs from the Task-participating partners. The aim of the workshop was to identify and address issues with relation to the evolving architecture. An asterisk (*) is used to indicate that a component was identified during the workshop, and is not mentioned in the GA RA, while a revision superscript (^{r}) is used to indicate a component with a revised name and role (with the previous corresponding component in brackets).

Table 8: List of identified main and total functional blocks in the OPTIMAI architecture reference model

Component	Component No.	Sub-component(s)	Partner(s) responsible	Implementation Example (Product, or document Section)
Quality Control Sensors Network	001	3D Scanner*	EVT	• AIC Saturn 3D Sensor • EyeScan AT 3D sensor model
	002	3D Area Sensor*	EVT	• Wenglor ShapeDrive MLAS/MLBS 3D Sensors Families
	003	3D Time of Flight (ToF) Camera*	EVT	• LUCID Helios2 ToF Camera

	004	3D Light Detection And Ranging (LI-DAR)*	EVT	• [Percipio XYZ FM811-GIX-E1]
	005	3D Triangulation Sensor*	EVT	• [EyeScan ST 3D Sensor]
	006	2D Camera*	EVT	• [AIC Merkur Smart Line Scan Camera]
	007	1D Camera*	EVT	• [EyeScan ZLS & ZM "Jupiter" smart cameras]
	008	Air Quality Sensor*	FINT	• [FINoT Indoor Air Quality node]
Edge Computing Modules*	-	**Middleware**	FINT	Section 4.1.3
	009	AI Edge Processing Service module[r] (Acquisition Optimisation)	ENG	Section 4.1.2.1
Middleware	010	Multimodal Data Collection Agent[r] (Multimodal Data Registration)	FINT	Section 4.1.3.1
	011	FINoT Platform*	FINT	Section 4.1.3.2
	012	Middleware Service*	FINT	Section 4.1.3.3
	013	Cybersecurity Defence Module	FINT	Section 4.1.3.4
Cloud Computing Modules	-	**Middleware Cloud Data Repository[r] (Data Repository)**	FINT	Section 4.1.5
	-	**Blockchain**	CERTH	Section 4.1.6
	-	**Operator-Machine Interaction & Decision Support (OMIDES) Back-End***	CERTH	Section 4.1.7

	-	**Intelligent Marketplace Back-End***	FINT	Section 4.1.8
	-	**AI Framework**	UTH CERTH VIS FORTH	Section 4.1.9
Middleware Cloud Data Repository	014	File Storage^r (Data Collection)	FINT	Section 4.1.5.1
	015	Historical Data	FINT	Section 4.1.5.2
	016	Open Datasets	FINT	Section 4.1.5.3
Blockchain	017	Firmware validation Service	CERTH	Section 4.1.6.1
	018	AI Model Integrity Verification Service	CERTH	Section 4.1.6.2
	019	Access Control Service	CERTH	Section 4.1.6.3
	020	Data Integrity Service	CERTH	Section 4.1.6.4
OMIDES Back-End	021	Pose Estimation Service	CERTH	Section 4.1.7.1
	022	Activity Recognition Service	CERTH	Section 4.1.7.2
	023	Instance Segmentation Service	CERTH	Section 4.1.7.3
Intelligent Marketplace Back-End	024	Part Indexing	FINT	Section 4.1.8.1
	025	Marketplace Authorisation	FINT	Section 4.1.8.2
	026	Search Engine	FINT	Section 4.1.8.3
AI Framework	-	**Digital Twinning^r (Digital Twins)**	VIS	Section 4.1.10
	-	**Smart Quality Control**	CERTH UTH	Section 4.1.11

	027	Manufacturing (re-)configuration Service	FORTH	Section 4.1.9.1
Digital Twinning	028	Process Digital Twins[r] (Process Modelling)	VIS	Section 4.1.10.1
	029	Virtualised Sensor Network[r] (Sensors Modelling)	FINT	Section 4.1.10.2
	030	AI Production Planning Simulation Engine*	VIS	Section 4.1.10.3
Smart Quality Control	031	Defect Detection & Prediction Service	CERTH	Section 4.1.11.1
	032	Production Monitoring & Quality Control Service[r] (Production Monitoring)	UTH	Section 4.1.11.2
End-users' Applications*	033	OPTIMAI Intelligent Marketplace Customer Front-end*	FINT	Section 4.1.12.1
	-	**OMIDES Front-End***	CERTH FORTH	Section 4.1.13
	034	Visual Simulation Engine*	VIS	• Visual Components 4.0 (custom/modified)
OMIDES Front-End Augmented Reality (AR) App	035	Rendering and AR Interaction Core Components	FORTH	Section 4.1.13.1
	036	Visual Analytics and Graphical UI	FORTH	Section 4.1.13.2

3.2 Bottom-up refinement

Bottom-up design refers to the process in which the components of a system are specified in detail. During this process units are linked to form a more complex system. For the specification of the OPTIMAI architecture, the bottom-up approach was subsequent to the top-down approach. During this phase, technical partners provided their input with regard to the technologies and software components they are planning to contribute to the project. These may be components that partners bring to the project as part of their background, or they have developed as part of the work carried out in the framework of the project Tasks (foreground). As was the case with the top-down phase, a first bottom-up design approach was implemented during the time the project proposal was written. This process helped select appropriate partners for the OPTIMAI consortium on merit of their technological expertise, to further elaborate on the foreseen technical contributions they would deliver. This exercise offered insight into the architectural components required for the technical realisation of the project.

After the top-down architecture design had been revised, a component template (Table 9) was created and shared using the project NextCloud collaboration platform. The component template served as a means of gathering information about the functional blocks identified in the top-down specification (Table 8), that would eventually form the concrete OPTIMAI architecture. A second architecture workshop was organised during this step (M06-12) of the architecture definition process to gather additional information and insight on components descriptions provided. Along with the role and responsibilities of the components, additional information was gathered, such as foreseen interdependencies with other functional blocks by means of expected input and output (towards specifying the information flow), as well as specific dependencies on software (and hardware) elements, to form a basis of the deployment architectural view. Task leaders responsible for the development/modification/update of each component were asked to revise and complete the template based on this updated information. The outcomes of this exercise constitute the contents reported in Section 4 of this deliverable, as identified in Table 8.

Table 9: Component specification template for bottom-up design process

[Name of new tool / component]	
Main functions	*Describe the main functions of the component and related to OPTIMAI's objectives and milestones.*
Connections/Dependencies/Interfaces	*How the component relates with other components as well as OPTIMAI's tasks and deliverables.*
Software dependencies	*Any dependency the component has on other libraries, frameworks, operating system etc.*
Functional Requirements	*Describe the functionality of the system.*
Non-Functional Requirements	*Describe the characteristics and features of the component.*
Input Parameters	*What is the input of the component (type and format).*

Output Parameters	*What is the output of the component (type and format).*

3.3 Ethical compliance – Compliance through Design

The OPTIMAI project incorporates an Ethics by Design (EbD) and Legal Compliance through Design (LCtD) approach. In the case of the former, it adheres broadly to the guidance offered in the European Commission-published document Ethics By Design and Ethics of Use Approaches for Artificial Intelligence, a document building on the results of the SIENNA and SHERPA projects [13].

The purpose of by and through design approaches is to integrate ethical and legal norms and standards into the fabric of the research project at an early stage (or to embed them in institutional response), from design through to implementation and deployment in actual use cases, also with a view with securing the technological outputs', especially AI tools', value and capacity for compliance into the future [13] [14].

Ethical and legal research and action are built into the project primarily through WP9, the objectives of which are to establish, implement and monitor an ethical and legal framework which comprehensively incorporates ethical, legal and regulatory sources. Supporting compliance with this framework are tasks including regularly scheduled integrated ethical, legal and societal risk assessments beginning with D9.2 (M12, M24, M36) and an enforcement and monitoring strategy to ensure ethical and legal compliance is adequately mainstreamed throughout project activities.

The sources of the ethical, and legal OPTIMAI framework are multifarious and are comprehensively outlined in D9.1. Sources include hard law (e.g., the GDPR and the European Convention on Human Rights), soft law (e.g., *United Nation Guiding Principles on Business and Human Rights (2011)* and *ISO 26000 Guidance Standard on Social Responsibility (ISO 26000)*, and ethical principles (e.g., the Independent High-Level Expert Group on Artificial (AI HLEG) *Intelligence's Ethics Guidelines for Trustworthy AI*).

These norms are translated into ethical and legal requirements governing the design, development and use of OPTIMAI solutions as well as the conduct of research activities. Furthermore, as an output of the project's first risk assessment, additional ethical and legal requirements have been devised in response to the identified ethical, legal, and societal risks, and are presented in D9.2. These ethical and legal requirements are applicable to all OPTIMAI partners who must incorporate them into their work. In their totality, the ethical and legal requirements reflect principles arising from SIENNA, SHERPA and the AI HLEG of [13]:

- Respect for Human Agency
- Privacy and Data Governance
- Fairness
- Individual, Social and Environmental Well-Being
- Transparency

- Accountability and Oversight

As the by and through design approaches also entail frameworks of governance, this is described and expounded upon in D9.5 and will continue to be built upon in further iterations of this deliverable.

With regards to the architecture of OPTIMAI solutions, these ethical and legal requirements fundamentally oblige partners to select appropriate hardware and data collection and transmission methods that properly minimize (including through technical and automated methods of anonymization or pseudonymisation) and secure all potential personal and proprietary data that is collected; that data subjects' rights can be upheld by data controllers and processors responsible for elements of the architecture that carry personal data; that data flow and system operations are well-documented and auditable; that stakeholders (including research participants and factory employees) are adequately informed and consulted about data processing activities and changes to their working environment; and that components of OPTIMAI solution architecture and their interactions cause a minimal adverse impact on the environment.

Partners will continue to collaborate to identify and implement the best possible methods and measures for complying with ethical and legal requirements through continuous specification efforts, and all efforts will be monitored and supported by the ethics and legal team.

4 OPTIMAI Architecture - Functional view

Starting from this Section, the core viewpoints of the OPTIMAI architecture will be specified, leading to a concrete first version description of the envisaged solutions in alignment to important reference standards. Particularly, this Section will present the functional entities comprising the fundamental architectural components and subsystems previously identified in Table 8, elaborating on their foreseen roles and responsibilities within the system overall function, as well as the synergies that they should implement with other components, both internal and external to the OPTIMAI framework. A comprehensive component diagram of the proposed solution is presented in Figure 8.

This architectural view presented follows the approach of segmenting the envisioned IT systems on a vertical axis, thus adopting a layered design that indicates the high-level classification of the different technological components in accordance to their contribution to the key user requirements of the project use cases. The functional architecture stack depicted in Figure 8 is further aimed at highlighting rudimentary principles of the OPTIMAI development over time, such as the properties, relationships and execution environment of the functional elements. Each layer corresponds to a major subsystem driving the flow of information from top to bottom (i.e., from the sensing IoT hardware all the way to the users' equipment). These subsystems/nodes are: (i) the Quality Control Sensors Network; (ii) the Edge Computing Modules; (iii) the Cloud Computing Modules; and (iv) the Users' Applications. Specific alignment to the deployment characteristics of each subsystem are discussed in more detail in Section 6. The purpose of this Section is to instead highlight the analogy of use case requirements to the different functional blocks' responsibilities (i.e., making sure component functionalities are well-represented, and that they can be properly matched against elicited system requirements). The descriptions are provided at a functional level, which omits confronting specific implementation details at architectural level, as they are particular to the development Task delivering each component to the final integrated system.

In addition to the definition of components' behaviours and interrelations, care has been taken to refine the original architecture reference model (discussed in Section 2.3) in accordance to well-defined standards (as discussed in Section 2) so as to support interoperability. Such an alignment enables extension of the OPTIMAI solution in a manner that would enable its replication for use cases outside the ones explicitly defined in the scope of the project, within the context of both I4.0 (via a direct mapping of OPTIMAI to the RAMI 4.0 cubic model) and IIoT (by applying mappings drawn between IIRA and RAMI 4.0) scenarios.

The remainder of this Section is organised as follows: Section 4.1 delivers on the descriptions for the subsystems and components depicted in Figure 8, which are directly derived from the templates handed out to project technical partners during the bottom-up design specification, as discussed in Section 3.2. With the functional perspective in place, Section 4.2 then discusses the alignment of the OPTIMAI architecture with the two major standards-led reference architectures, RAMI 4.0 and IIRA.

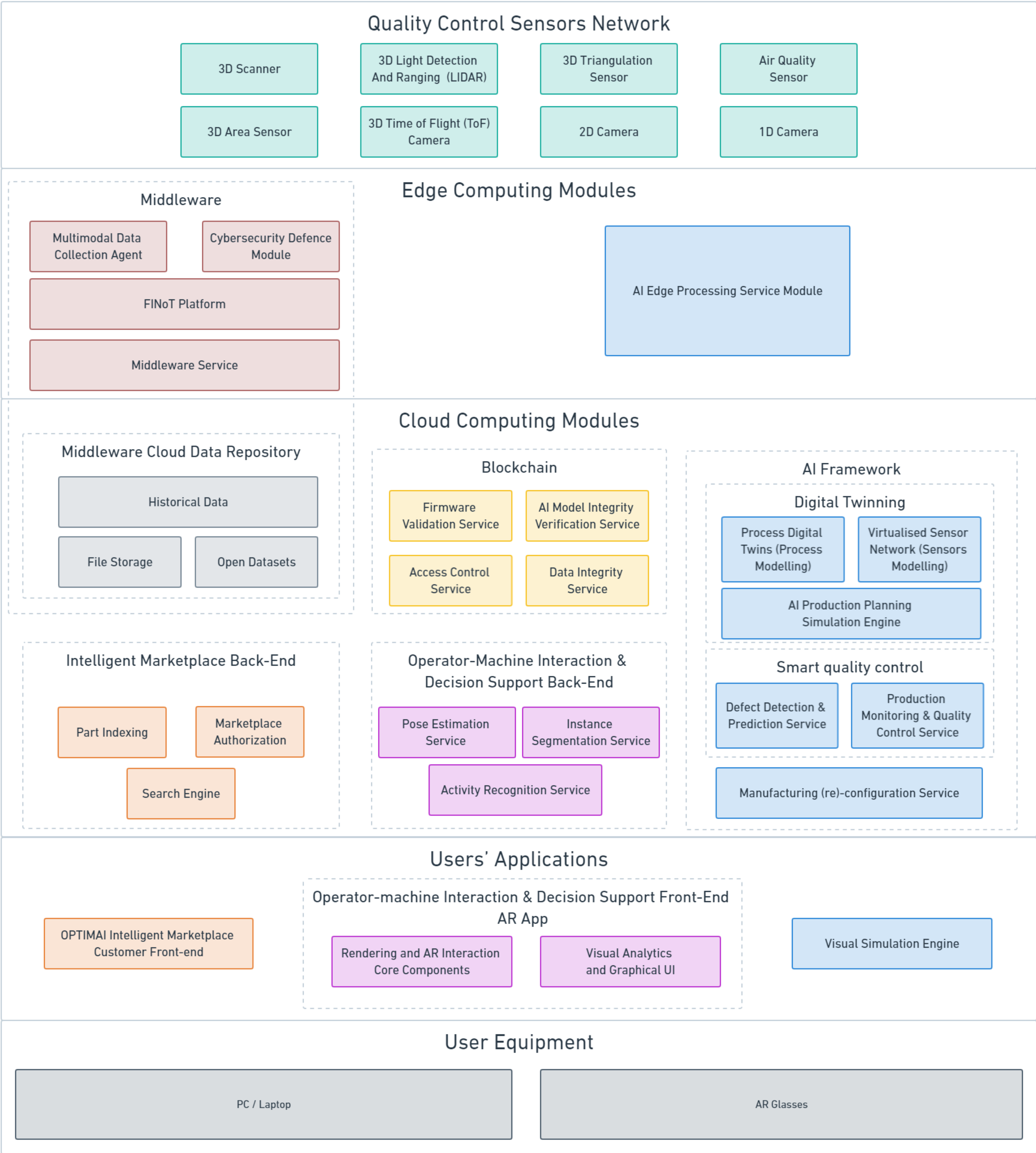

Figure 8: OPTIMAI Functional architecture view – component diagram.

4.1 OPTIMAI functional blocks

The following sub-sections provide a description for each of the functional blocks (both components and larger subsystems) comprising the OPTIMAI system as illustrated in Figure 8. Wherever possible, core system entities (at any-level, i.e., component or subsystem) will be described also in relation to its interdependencies with other OPTIMAI components (thus elaborating on its role in the overall information structure supported by the architecture), as well as the primary interactions formed with other functional elements by means of expected inputs and outputs driving the foreseen runtime behaviour.

4.1.1 Quality Control Sensors Network Subsystem

The Quality Control Sensors (QCS) Network subsystem is comprised of all IoT sensor devices employed for data collection regarding current production parameters. A variety of device types have been identified during the first year of the project in the context of Task 3.1 “Multisensorial data acquisition and actuation network”, and a complete list of devices and specifications will be provisioned with D3.1 due in M16. The following sub-Sections aim at presenting a non-exhaustive list of the different sensor types and families that are considered for use within the OPTIMAI framework for the different use cases of the project. These devices support the higher-level functions (such a AI routines) provided by other components in the architecture by means of specific data acquisition methods and formats (e.g., pose estimation from RGB image data).

The following paragraphs will hence provide more details regarding these key sensory apparatuses that will be utilised within the OPTIMAI architecture for the purposes of feeding vision-based data to the AI algorithms running either at the edge or on the Cloud, thus making the information flow (presented in Section 5) much easier to comprehend at this early stage in the project. These sensors described here will complement the arrays of sensors and actuators that already exist in the pilot site facilities. Hence, it is expected that this subsystem composition will be updated based on the contents of D3.1, with such updates being included in the future version of this deliverable (D2.5). Eventually, all sensor/IoT devices will support communication with the Middleware subsystem based on a standard communication protocols e.g., IEEE 802.15.4, OPC Unified Architecture (OPC UA), OASIS Message Queuing Telemetry Transport (MQTT), etc.

4.1.1.1 001 3D Scanner

One or more 3D Scanner devices will be employed in the QCS Network for the acquisition of 3D Point Clouds, Depth and Intensity image data at different resolutions. Provisioned by EVT, particular product examples that can cover the specifications of this sensor include the AIC Saturn 3D Sensor and the EyeScan AT 3D sensor model.

4.1.1.2 002 3D Area Sensor

One or more 3D Area Sensor devices will be employed in the QCS Network for the acquisition of 3D Point Clouds Depth and Intensity image data based on structured light with different resolutions. Neither the object, nor the 3D sensor is in motion during the scan. Provisioned by EVT and UNIMET, particular product examples that can cover the specifications of this sensor include the Wenglor Shape-Drive MLAS/MLBS 3D Sensors Families, and the UNIMET optical sensors used in 3D scanning technology.

4.1.1.3 003 3D Time of Flight Camera

One or more real-time 3D Time of Flight (ToF) camera sensor devices will be employed in the QCS Network for the acquisition of 3D Point Clouds Depth and Intensity image data, supporting various Field of View settings and as such, different resolutions of 3D point cloud data. Provisioned by EVT, particular product examples that can cover the specifications of this sensor include the LUCID Helios2 ToF Camera sensor.

4.1.1.4 004 3D Light Detection And Ranging

One or more 3D Light Detection And Ranging (LIDAR) camera sensor devices will be employed in the QCS Network for the acquisition of 3D Point Clouds Depth and Intensity image data, supporting various Field of View settings and hence different resolutions of 3D point cloud data. Provisioned by EVT, particular product examples that can cover the specifications of this sensor include the LUCID Helios2 ToF Camera sensor.

4.1.1.5 005 3D Triangulation Sensor

One or more variable smart 3D Triangulation Sensor devices will be employed in the QCS Network with integrated processing chip for delivering edge processing capabilities of the captured image data (e.g., point clouds and intensity images) inside the component itself for performing data evaluation and feature detection with pre-trained neural network models. Provisioned by EVT, particular product examples that can cover the specifications of this sensor include the EyeScan ST 3D laser triangulation sensor ("Saturn").

4.1.1.6 006 2D Camera

One or more variable smart 2D Camera devices will be employed in the QCS Network with integrated processing chip for delivering edge processing capabilities of the captured image data (e.g., Grey and Color Matrix Data) inside the component itself for performing data evaluation and feature detection with pre-trained neural network models. Provisioned by EVT, particular product examples that can cover the specifications of this sensor include the AIC Merkur Smart Line Scan Camera.

4.1.1.7 007 1D Camera

One or more variable smart 1D Camera devices will be employed in the QCS Network with integrated processing chip for delivering edge processing capabilities of the captured image data (e.g., Image Linescan Sensor Data at resolutions within 2 – 6K) inside the component itself for performing data evaluation and feature detection with pre-trained neural network models. Provisioned by EVT, particular product examples that can cover the specifications of this sensor include the EyeScan ZLS & ZM "Jupiter" lines of smart cameras. In addition, cameras under this category may include head-mounted sensors worn by manufacturing line operators as part of the Augmented Reality (AR) glasses hardware solution developed by YBQ.

4.1.1.8 008 Air Quality Sensor

One or more Indoor Air Quality IoT devices will be employed in the QCS Network for monitoring the air quality of the space that the sensor is deployed. The device corresponding to this description will be equipped with several sensors for pertinent data collection, such as Air Temperature, Air Relative Humidity, Total Volatile Organic Compounds (TVOC), and CO_2 emissions. Provisioned by FINT, particular product examples that can cover the specifications of this device include the FINoT Indoor Air Quality node solution.

4.1.2 Edge Computing Modules Subsystem

Edge computing is emerging as a critical enabler in smart manufacturing, enabling the rapid execution of code originally intended for Cloud computing resources on either the devices themselves, or a gateway device or Personal Computer (PC) in close proximity. To capitalise on this emerging paradigm, the OPTIMAI architecture incorporates subsystems and modules for deploying on-the-edge intelligence regarding monitoring and control of specific ("smart") sensors in the QCS Network subsystem (005-007 & 008). This enables OPTIMAI to dynamically manage the various acquisition parameters.

In support of this edge node architecture, the OPTIMAI Edge Computing Modules subsystem defines the following sub-modules:

- the **Middleware subsystem** (Section 4.1.3), which provisions services for: (i) the collection of all sensors' data in real time; (ii) the application of cybersecurity at the acquisition level (as soon as data enters the system); (iii) sensor health monitoring functions; and (iv) coordination of the exchange of information between the edge and cloud modules.
- the **Artificial Intelligence (AI) Edge Processing Service module** (Section 4.1.2.1), which is responsible for operating the AI services on-the-edge, so as to optimize the acquisition, detection and analysis processes.

4.1.2.1 009 AI Edge Processing Service module

The AI Edge Processing Service module is part of the OPTIMAI edge node architecture, and enables the deployment and operation of AI services on-the-edge. Hence, it will directly relate to the AI services developed in OPTIMAI, and will target sensor acquisition optimisation, real-time analytics modules and adaptation of intelligent manufacturing assets. The component will undertake the responsibility of pursuing the following objectives:

- **Deployment of AI services on-the-edge**: Deployment and operation of AI services on-the-edge, which it will support in the form of services to both optimise acquisition routines, as well as applying real-time adaptation for reconfiguring "smart" manufacturing assets.
- **Smart data management and data distribution mechanisms**: Reducing of downstream computation, network traffic and data volume (as deployment of the component will alleviate the need to continuously send constant values to the Cloud services).
- **Correct circulation of data across all OPTIMAI endpoints**: Optimisation of resource usage (e.g., network, energy, storage), while providing guarantees of the actuation on production equipment.
- **Micro-service execution at edge nodes**: Implementation of lightweight operations at edge nodes (e.g., sensors equipped with processing chips, IoT gateways or PCs in close proximity to the sensor devices), while supporting collaboration with Cloud workflow orchestration and monitoring services.

Table 10 summarises the foreseen interdependencies of the AI Edge processing Service module with other OPTIMAI components, while Table 11 presents preliminary examples of expected runtime data inputs and outputs of the component.

Table 10: Inter-dependencies of the AI Edge Processing Service module with other components

Component Name	Inter-dependency description
Middleware subsystem	Device management and data registration for input data.
Smart Quality Control subsystem	Execution of AI models and algorithms for quality control toward zero defect manufacturing.
Manufacturing (re)-configuration service	Provision of quality control results to directly enable re-adjustment of equipment parameters.

Table 11: Input/Output of the AI Edge Processing Service module

I/O Type	Data
Input	Sensorial data (presumably in JSON format, but dependant on the Middleware output format support).
Output	Optimized sensorial data (preferably in JSON format).

4.1.3 Middleware Subsystem

The Middleware subsystem provisions secure interfaces between the various units, components, sensors and subsystems. It supports the network protocols required to exchange control and store data and information needed to facilitate operations and services in an environment with many different networking and system components. As such, the Middleware subsystem incorporates all necessary services required for integrating the various sensor devices in the QCS Network within a unified framework. Aside from data collection, a core purpose of this subsystem is to expose a unified interface to interact with the other OPTIMAI platform components, exposing all necessary functionalities by establishing all necessary API endpoints (based on the Representational State Transfer - REST architecture) towards the interdependent platform components.

This subsystem is based on FINT's commercial *FINoT Platform* solution, albeit extended to implement the necessary functionality dictated by the project objectives. Based on the architecture of the FINoT solution, the Middleware subsystem will provision functions toward: (i) sensors' data acquisition and registration; (ii) facilitating storage, processing and aggregation (on the edge); (iii) incorporate cyber-security defence in the form of authentication/authorisation routines; and (iv) expose functions through a powerful REST API interface.

The following sub-Sections will detail the Middleware components. Table 12 summarises the foreseen interdependencies of the Middleware subsystem components with other OPTIMAI functional entities, while Table 13 presents preliminary examples of expected runtime data inputs and outputs of the component.

Table 12: Inter-dependencies of the Middleware subsystem with other OPTIMAI components

Component Name	Inter-dependency description
QCS Network	Exchanges supporting data acquisition and configuration control toward the sensor devices.
All other OPTIMAI subsystems	Aggregated/registered quality inspection and production data.

Table 13: Input/Output of the Middleware subsystem

I/O Type	Data
Input	The sensor data collected from the area that the sensors deployed.
Input	Status monitoring information collected from the deployed sensor nodes and the IoT agents.
Input	Control command result status.
Input	Incoming network traffic.
Output	Current and historical IoT sensor data.
Output	Control commands to any devices/sensors with actuation capabilities.
Output	Cyber security analysis results in JSON format

4.1.3.1 010 Multimodal Data Collection Agent

Corresponding to the functionalities provisioned by the FINoT IoT Agent, the responsibility of this module is to translate the raw data coming from the QCS Network's devices into context aware entities (Next Generation Service Interfaces - NGSI) and vice versa.

4.1.3.2 011 FINoT Platform

This module is the "heart" of the Middleware subsystem, and corresponds to the FINoT Platform commercial solution (hence treated as a component from OPTIMAI's perspective, despite being further broken down into sub-modules, as described next). It consists of several provisioned services responsible for storing, processing and aggregating the acquired data. On this block, it will also be possible to execute event-driven, real-time operations, using a Complex Event Processing (CEP) engine, or asynchronous tasks through a Task scheduler. The responsibility of this functional block will include: (i) collection, management and access provision to data from different sources by means of a context broker; (ii) Edge-based entity storage for holding all the data required by the context broker; (iii) a CEP engine responsible for executing complex, event-driven tasks over real-time streams of sensor data; and (iv) a Task scheduler, responsible for executing asynchronous tasks on the platform.

4.1.3.3 012 Middleware Service

The basic purpose of the Middleware Service is to creates and provides all the necessary REST APIs endpoints toward external to the Middleware OPTIMAI components, so as to interact with the subsystem's components. Additionally, it provides the authentication and authorization functionalities to secure the platform and the resources, and implement managing of the users in the system. As such, it provides all the necessary tools to authenticate and authorize users to access an application, API, service or any other data resource available in the Middleware subsystem.

4.1.3.4 013 Cybersecurity Defence module

Cybersecurity Defence module facilitates the use of acceleration (by means of a field-programmable gate array - FPGA) for: (i) detecting and analysing cybersecurity threats; (ii) monitoring and managing the AI and cybersecurity algorithms; and (iii) provision the communication/interfacing modules, that will enable the interaction with other external components (central logging server, software agents, etc.).

4.1.4 Cloud Computing Modules Subsystem

The Cloud Computing Modules subsystem contains the OPTIMAI components that are expected to be deployed in a cloud computing environment, due to their foreseen needs in storage and computational power (e.g., for processing-heavy AI and other routines). As such, it houses the following subsystems and modules:

- the **Middleware Cloud Data Repository subsystem** (Section 4.1.5), which acts as the centralised storage point of the entire OPTIMAI system.
- the **Blockchain subsystem** (Section 4.1.6), which is responsible for maintaining a distributed ledger of all critical operations, the employment of smart contracts to automate several production processes, and the provision of data integrity verification mechanisms.
- the **Operator-Machine Interaction & Decision Support (OMIDES) Backend subsystem** (Section 4.1.7), which provides the necessary AI routines for processing of the visual information as obtained from the point of view of an operator wearing Augmented Reality (AR) glasses, in order to conclude on spatial and contextual information regarding the production line as communicated to the system by an "augmented" operator.
- the **Intelligent Marketplace Back-End subsystem** (Section 4.1.8), which will facilitate user-generated data storage and transactions functionality for the envisioned OPTIMAI scrap and AI models marketplace solution.
- the **AI Framework subsystem** (Section 4.1.9), which provides resources for the virtualisation and simulation of the production line, executes smart quality control processes and calculates production optimization parameters based on predictive analytics using the powerful resources available the cloud.

4.1.5 Middleware Cloud Data Repository Subsystem

The Middleware Cloud Data Repository is the centralised storage space for all data circulated in the OPTIMAI system. As the name suggests, its purpose is to support persistence mechanisms for storing sensor Middleware-aggregated data for the purposes of updating the training of AI models. In addition, it will contain historical data archives as well as any open datasets needed for algorithm training. Each of the following modules described in the sub-Sections below will support compatible interfaces for supporting both storage and retrieval of data, by all interconnected OPTIMAI components and subsystems.

Table 14: Inter-dependencies of the Middleware Cloud Data Repository subsystem with other OPTIMAI components

Component Name	Inter-dependency description
Middleware subsystem	Data handling and data registration.
Blockchain Subsystem	Data verification.
Smart Quality Control subsystem	Pose estimation data toward the semantic fusion of quality control information.
AI framework	AI solutions to be verified.

Table 15: Input/Output of the Middleware Cloud Data Repository subsystem

I/O Type	Data
Input	Verified sensorial data (presumably in JSON format, but dependant on the Middleware output format support).
Output	Optimized sensorial, historical data and open datasets (preferably in JSON format).

4.1.5.1 014 File Storage

This module provides an S3-compatible API for file storage in the cloud.

4.1.5.2 015 Historical Data

This module is responsible for managing (storing and retrieving) historical raw and aggregated time series data registered in the FINoT Platform context broker.

4.1.5.3 016 Open Datasets

This block represents a collection of dataset modules for the inclusion of open datasets for defect detection routines in the OPTIMAI system. These datasets can either be partner property (i.e., created during past research activities), or publically open data made available to the scientific community by its creators.

4.1.6 Blockchain Subsystem

The Blockchain subsystem will be responsible for dealing with system objectives regarding the security, privacy, traceability, integrity, compatibility and interoperability of data storage and exchange. In addition, a blockchain-based AI model integrity validation mechanism will be provisioned. The foreseen developed Cloud-based subsystem will be provisioned by implementing a Blockchain-as-a-Service (BaaS) model. In this regard, a blockchain Application Programming Interface (API) will be exposed to enable traceability and validity of every data transaction occurring in the system, whether it be a collection of measurements, a reconfiguration request, or a sensor health check.

This subsystem is implemented to deliver on core security, privacy and data transparency concerns within I4.0 smart manufacturing use cases [15]. The specific objectives pursued by this subsystem are organised in four cloud-based blockchain modules targeted at: (i) firmware validation; (ii) access control; (iii) Data integrity; and (iv) AI Model integrity functions. Because of its role in the OPTIMAI architecture, the Blockchain subsystem supports interdependencies with several other OPTIMAI components, as listed in Table 16.

Table 16: Inter-dependencies of the Blockchain subsystem with other OPTIMAI components

Component Name	Inter-dependency description
Middleware Cloud Data Repository subsystem	Support for sensorial and transactional data exchange.
Middleware subsystem	Device management and data registration for input data
Intelligent Marketplace Back-End subsystem	Verified AI solutions in order to be re-used for other purposes such as testing or refurbishing.
AI framework	AI solutions to be verified

The foreseen input/output parameters for supporting these exchanges are listed in Table 17.

Table 17: Input/Output of the Blockchain subsystem

I/O Type	Data
Input	Sensorial and transactions data.
Output	Records on the distributed ledger of the blockchain framework that will be used.

4.1.6.1 017 Firmware Validation Service

The Firmware Validation Service will be responsible for recording all critical system processes as immutable and verifiable transactions in a blockchain network. This service will also record actions related to firmware versions running on sensors. Regarding the sensors measurements, a side chain, i.e., an external database will be utilized to store the high volume of data.

4.1.6.2 018 AI Model Integrity Verification Service

The AI Model Integrity Verification Service will utilise dedicated blockchain-based integrity verification mechanisms directed at the machine learning models produced by the AI Framework during training of models, so as to ensure application of correct AI models in the prediction of configuration parameters within the production line.

4.1.6.3 019 Access Control Service

The Access Control Service will use smart contracts to automate a variety of processes within the production line, and develop an access control mechanism to prevent unauthorized users from undertaking critical system activities.

4.1.6.4 020 Data Integrity Service

The Data Integrity Service will use blockchain-based data integrity methods to verify the integrity of the software and firmware versions deployed on the sensors and actuators, along with the measurements acquired by the system's sensors, so as to enforce traceability and accountability and thus safeguard the system against outside tampering attempts.

4.1.7 OMIDES Back-End Subsystem

The OMIDES Back-End subsystem encompasses all the different components that power the functionality of the foreseen OPTIMAI AR Application solution for defect detection and notification. Specifically, it includes AI routines that enable detection, interpretation and visualisation of information obtained through visual means by: (i) analysing captured 2D image data toward categorising and estimating the 3D pose of detected objects in the manufacturing line; and (ii) performing activity recognition to understand operators' actions and intentions, thus facilitating gesture-driven interaction of operators by correctly classifying performed gestures and mapping them onto application functionality. The following sub-Section describe the foreseen functionality provisioned by the AI services housed in this subsystem.

4.1.7.1 021 Pose Estimation Service

The Pose Estimation Service aims at estimating the pose of detected objects of interest (e.g., produced parts, production machines, etc.) in the live video frames captured by the AR smart glasses worn by an operator inspecting the line. The service can thus provide personalised information to each user, as it will analyse objects' 3D properties with respect to the position and current viewing angle of the operator. Emphasis is placed on this service implementing real-time algorithms with minimal computational requirements, which may unlock the capacity to selectively execute the pose estimation code on the edge device itself. Toward this end, state-of-the-art pose estimation networks will be tested and further developed in the context of Task 5.1.

Table 18 summarises the interdependencies of the Pose Estimation Service with other OPTIMAI components, while

Table *19* presents the data inputs and outputs of the component.

Table 18: Inter-dependencies of the Pose Estimation Service with other OPTIMAI components

Component Name	Inter-dependency description
OMIDES Front-End	Provision of the estimated object key points, that may be displayed on the user's field of view in the augmented reality environment.
Smart Quality Control subsystem	Pose estimation data toward the semantic fusion of quality control information.

Table 19: Input/Output of the Pose Estimation Service

I/O Type	Data
Input	Video frames originating from an operator's AR glasses.
Output	Estimated keypoints of the objects of interest within the input image (in JSON format).

4.1.7.2 022 Activity Recognition Service

The Activity Recognition Service will facilitate the development of AI methods for activity and gesture recognition, which are required to be able to translate operator hand motion information captured through the AR glasses head-mounted camera into smart application commands for reconfiguring production parameters.

Table 20 summarises the interdependencies of the Activity Recognition Service with other OPTIMAI components, while Table 21 presents the foreseen inputs and outputs of the component.

Table 20: Inter-dependencies of the Activity Recognition Service with other OPTIMAI components

Component Name	Inter-dependency description
OMIDES Front-End	Exchange of gesture data regarding the performed AR interaction, and identified gesture for the purposes of visualisation/visual feedback.
Middleware subsystem	Device management and data registration for input data.
Manufacturing (re-)configuration Service	Provision of gesture data for mappings to a gesture vocabulary and election of the appropriate function for AR-driven actuation.

Table 21: Input/Output of the Activity Recognition Service

I/O Type	Data
Input	Sensor data, specifically from multiple camera sensors (RGB images).
Output	Images with additional layers of information (RGB images), classified images based on classes of task undertaken by an operator.

4.1.7.3 023 Instance Segmentation Service

The Instance Segmentation Service performs semantic segmentation on the live feed camera frames obtained from the AR glasses embedded camera that the operator wears. Objects of interest include produced parts and their constituent modules, along with production machines. Few-shot learning techniques will be employed so that new types of objects from each pilot site can be successfully identified without requiring a large number of samples for training. Future considerations of deployment on the edge will place emphasis on solutions with real time performance and low computational requirements.

Table 22 summarises the interdependencies of the Instance Segmentation Service with other OPTIMAI components, while Table 23 presents the foreseen inputs and outputs of the component.

Table 22: Inter-dependencies of the Instance Segmentation Service with other OPTIMAI components

Component Name	Inter-dependency description
OMIDES Front-End	Provision of segmentation output data used for rendering outlines of detected objects.
Defect Detection & Prediction Service	Provision of segmentation output data for enabling defect detection & prediction methodologies to act directly on image data obtained from the AR glasses.

Table 23: Input/Output of the Instance Segmentation Service

I/O Type	Data
Input	Live feed from AR glasses, or other external 2D cameras.
Output	The segmentation mask indicating production equipment, semi-finished parts and final products.

4.1.8 Intelligent Marketplace Back-End Subsystem

In its entirety, the OPTIMAI Intelligent Marketplace (distinct for architectural purposes into Back-End and Front-End – the latter described in Section 4.1.12.1), aims at helping the manufacturing ecosystem players decrease scrap within their production line. In this direction It will provide:

- services based on the AI algorithms developed in OPTIMAI. This will provide marketplace functionality to third parties that would like to increase their production quality by minimising scrap (using OPTIMAI's capacity to identify potential defective parts, along with the prediction of defects and malfunctions algorithms).
- a marketplace for registered defective parts, which may be useful in the processes undertaken by another organisation (prior or after to the recycling process, e.g., reuse of scrap for lab measurements and material properties studying, etc.).

The subsystem described in this Section incorporates functionality related to storing and managing marketplace listings and transactions. Table 24 summarises the interdependencies of the Intelligent Marketplace Back-End subsystem with other OPTIMAI components, while Table 25 presents the foreseen inputs and outputs of the component.

Table 24: Inter-dependencies of the Intelligent Marketplace Back-End subsystem with other OPTIMAI components

Component Name	Inter-dependency description
OPTIMAI Intelligent Marketplace Customer Front-End	Provision and execution of marketplace-related functionality based on user input.

Defect Detection & Prediction	Defective part unique identification number for part indexing in the marketplace.

Table 25: Input/Output of the Intelligent Marketplace Back-End subsystem

I/O Type	Data
Input	User credentials.
Input	Marketplace Customer Front-End requests (e.g., description of available scrap items, preferences for specific scrap items, queries for available scrap items, requests for purchasing scrap items, etc.).
Input	Part unique identification number.
Output	User authentication results.
Output	Marketplace Customer Front-End request responses (e.g., feedback from buyers, list of available scrap items, buyers' search history, buyers' transaction history, recommended purchases of scrap items, sellers' feedback, etc.).
Output	Indexed Part listing registered to a particular seller for further development.

4.1.8.1 024 Part Indexing

Every part being entered into the OPTIMAI Intelligent Marketplace will be indexed by means of both front-end and backend keywords (i.e., keywords visible to the customer, and keywords used only within the backend). This module will be responsible for associating these keywords to the part unique identification number, optimising search engine functionality and results to customer queries.

4.1.8.2 025 Marketplace Authorisation

This component will aim at implementing a user authentication/authorisation mechanism based on the Single Sign-On (SSO) scheme so as to control access of users using a single set of login credentials. Additionally, this component will implement Role-Based Access Control (RBAC) in order to restrict access to specific functions provisioned by the Intelligent Marketplace Customer Front-End for authorised users only based on their roles (e.g., seller, buyer). Each user is assigned a role which defines the access level of the user for specific functionality The RBAC shall therefore allow high-level management of access to certain API endpoints (and hence, certain functionalities) only to those users that are meant to be able to use this functionality (e.g., only the authorised seller can have access to a part listing seller's UI).

4.1.8.3 026 Search Engine

This module will implement a search algorithm based on the indexed data provisioned by the Part Indexing component. The algorithm will rank results based on relevancy to the query using keywords, with the intent of displaying the most likely items to satisfy the users' needs at the top of the list.

4.1.9 AI Framework Subsystem

The AI Framework subsystem encompasses different subsystems and components that support the overall functionality of the OPTIMAI system regarding the use cases' execution. Hence it incorporates cloud-based, AI-driven solutions, such as:

- the **Digital Twinning subsystem** (Section 4.1.10) enabling the virtualisation of manufacturing systems along with AI-powered simulation of the production process for optimisation analysis.
- the **Smart Quality Control subsystem** (Section 4.1.11) for driving optimization of the production through data-intensive defect detection and prediction routines.
- the **Manufacturing (re-)configuration Service** (Section 4.1.9.1) for the intelligent orchestration of production equipment configuration.

4.1.9.1 027 Manufacturing (re-)configuration Service

The Manufacturing (re)-configuration Service will serve a two-fold purpose:

- **Manual reconfiguration by a human Operator:** To deliver on this functionality the component will implement a context-aware natural interaction interfacing scheme to be used in the production environment with support for a rich, adaptable and adaptive gesture interaction vocabulary, mapped to the specific preferences of the user obtained through a stored profile, as well as the context of use (e.g., equipment). To this end, a context-driven adaptive interaction ontological model will be employed, that will orchestrate (near)real-time re-configuration of equipment towards inferring the most suitable sequence of actions that the user should follow to interact from their position by using the appropriate gestures mapped from the vocabulary as the main mode of interaction. Essentially, the goal will be to provide fast context-aware recommendations to the operator wearing AR glasses, which will aim at anticipating emerging defects in production, and guiding the operator to address them through a re-configuration of machine parameters.
- **Automated reconfiguration:** This functionality will utilise the components' actuation pipeline to directly re-adjust equipment parameters based on quality control results obtained by the Smart Quality Control subsystem. The module will apply the recommended reconfiguration by using an internal agent (based on reinforcement learning principles) and without human intervention. The reward of the agent will be proportional to the quality level achieved, while training will be performed either with live or virtually collected data by means of the Digital Twinning subsystem.

Table 26 summarises the interdependencies of the Manufacturing (re)-configuration Service with other OPTIMAI components, while Table 27 presents the inputs and outputs of the component.

Table 26: Inter-dependencies of the Manufacturing (re-)configuration Service with other components

Component Name	Inter-dependency description
AI Edge Processing Service	Receive quality controls results to directly re-adjust equipment parameters.
Smart Quality Control subsystem	Provision of parameters regarding quality control and production monitoring results.
Activity Recognition Service	Provision of gesture data for mappings to a gesture vocabulary and election of the appropriate function for AR-driven actuation.
OMIDES Front-End	Provision of content regarding sequence of actions and proper gesture mappings that should be presented to an operator.

Table 27: Input/Output of the Manufacturing (re-)configuration Service

I/O Type	Data
Input	Quality control results in a JSON format
Input	Gestures recognized in JSON format
Output	Equipment configuration (JSON format)
Output	Content for visualization through the OMIDES front-end (JSON format)

4.1.10 Digital Twinning Subsystem

The Digital Twinning subsystem encompasses the different components that comprise simulation routines and functionality of a DT environment. It is based on a modified version of Visual Components 4.0 (VC4.0) by VIS, a commercial 3D simulation and visualization solution for discrete manufacturing. The solution allows simulating the entire production system at different levels of granularity, from a simple sensor or actuator to a whole manufacturing system with robotics, automation, logistics, and production flows.

For the purposes of the OPTIMAI architecture, the functionality of the modified VC4.0 is broken down into the back-end (described in this Section) and the front-end (described in Section 4.1.12.2). The components described in the following sub-Section constitute the base for the simulation (the digital model), which is the virtual representation of the physical asset in the virtual simulation space. Table 28 summarises the interdependencies of the Digital Twinning subsystem with other OPTIMAI components, while Table 29 presents its inputs and outputs.

Table 28: Inter-dependencies of the Digital Twinning subsystem with other OPTIMAI components

Component Name	Inter-dependency description
Visual Simulation Engine	Response/request exchanges associated to UI functionality (e.g., creating the virtual environment using the available resources).

Middleware Cloud Data Repository	Datasets used for the DT environment, e.g., production datasets (which include CAD data in different formats, product structure/BOM according to manufacturer, etc.), and production data sets from sensors and equipment.
Middleware subsystem	Actual equipment status/configuration, quality data from sensors.
Smart Quality Control subsystem	Candidate equipment status/configuration (to replace the actual being used), quality data from sensors.

Table 29: Input/Output of the Manufacturing (re-)configuration Service

I/O Type	Data
Input	Quality control results in a JSON format
Input	Gestures recognized in JSON format
Output	Equipment configuration (JSON format)
Output	Content for visualization through the OMIDES front-end (JSON format)

4.1.10.1 028 Process Digital Twin(s)

Process Digital Twin(s) (also referred to as *Process modelling* within VC4.0) refers to a feature provided by VC4.0, which enables fast and easy conceptualization of the manufacturing system that makes up the DT. The datasets used to create the manufacturing system can be production requirements for new production systems, or historical production data for existing production facilities. It deals with the creation of the virtual environment of the equipment and the products to be produced, and the definition of the production tasks (called "processes"). The latter result from the product structure (in the form of a Bill of Materials - BOM), which defines the processes that take place in the creation of the manufactured goods by specifying behaviours and specific details.

4.1.10.2 029 Virtualised Sensor Network

The Virtualised Sensor Network is based on simulation interfaces provisioned by VC4.0 for creating the sensors required to be simulated. It is used to virtualise sensors functionality and enable prediction and estimation of the sensor's future physical readings. These readings will be compared to the actual real values stemming from the physical counterparts of the virtualised sensors. In that way, the difference between the expected/estimated measurements and the real value may be used as an indicative of the physical sensor health besides predicting any near time possible failures in the process.

4.1.10.3 030 AI Production Planning Simulation Engine

The AI Production Planning Simulation Engine enables the analysis of the simulation datasets with the production datasets in order to infer production optimisations aimed at to enhancing productivity.

4.1.11 Smart Quality Control Subsystem

The Smart Quality Control is composed of functional blocks that provision the functionality necessary to perform analysis of the sensory signals and allow for the automatic identification and localization of defective parts. Additionally, it supports AI routines for deriving predictions of upcoming defects through the utilization of DTs, as well as an analysis of past data regarding the quality control measurements. The following sub-Sections details the services that comprise this subsystem.

4.1.11.1 031 Defect Detection & Prediction Service

The Defect Detection & Prediction Service focuses on detection of defective parts, as well as the prediction of upcoming defects through the employment of AI models on-the-edge. Consecutive measurements of previously produced parts are used to extrapolate future measurements, on which the developed defect detection methodologies will be applied for the detection of possible future defects. Several state-of-the-art deep architectures will be examined to support the necessary functionality, such as Deep Residual Networks for defect detection, and Long Short Term Memory (LSTM) and Generative Adversarial Networks (GANs) for defect prediction.

Table 30 summarises the interdependencies of the Defect Detection & Prediction Service component with other OPTIMAI components, while Table 31 presents the data inputs and outputs of the component.

Table 30: Inter-dependencies of the Defect Detection & Prediction Service with other OPTIMAI components

Component Name	Inter-dependency description
Middleware subsystem	Receive the appropriate sensor data.
OMIDES Back-End subsystem	Semantic fusion of quality control information, such as detected defects.
OMIDES Front-End	Information to be propagated to the AR environment.
Blockchain subsystem	Defective part unique identification number for part indexing in the marketplace.

Table 31: Input/Output of the Defect Detection & Prediction Service

I/O Type	Data
Input	• 3D Point Cloud data acquired from laser sensors. • RGB image data acquired from machine vision cameras. • time series data from industrial sensors.
Output	Part state information parameters in JSON format

4.1.11.2 032 AI Production Monitoring & Quality Control Service

The AI Production Monitoring & Quality Control Service is related to OPTIMAI's objective of developing AI methodologies for quality control. More specifically, this component will include the following quality control routines, organised for each use case specified in D2.6:

- KLEE site use case: Hydraulic lift Power Unit Quality Control
  - Check parts used
  - Valve block pressure monitoring
  - Testing
- MTCL site use case: Glue/epoxy diffusion, GPD dispensing system
  - Inspect the circuit
  - Defect Detection
- TVES site use case: Antenna line - defect detection
  - Defect detection

Developed algorithms will be provided in the forms of scripts, connected with proper APIs for receiving and sending data.

Table 32 summarises the interdependencies of AI Production Monitoring & Quality Control Service with other OPTIMAI components, while Table 33 presents the data inputs and outputs of the component.

Table 32: Inter-dependencies of the AI Production Monitoring & Quality Control Service with other components

Component Name	Inter-dependency description
Middleware subsystem	Acquisition of input data through the application of reading requests for transferring a representation of the state.
OMIDES Back-End subsystem	Provision of feedback to the requests of the decision support components.
Digital Twinning subsystem	Prediction of defects in the virtual replica of the manufacturing process

Table 33: Input/Output of the AI Production Monitoring & Quality Control Service

I/O Type	Data
Input	• Digital instances of individual parts: Photos/images, 3D Scans, Point clouds • Digital instances of assemblies. • Sensor readings (time-series).
Output	Defect information.

4.1.12 End-users' Applications

4.1.12.1 033 OPTIMAI Intelligent Marketplace Customer Front-end

The OPTIMAI Intelligent Marketplace Customer Front-End is a web-based marketplace front-end application delivering a rich UI experience for users to browse through scrap item/part listings, user profiles for sellers and buyers, content and feedback pages, guide buyers through the transaction flow etc. It will additionally feature a seller's administrative UI space for enabling authorised sellers access to a management dashboard for their part listings, monitoring capabilities of transactions and buyers' reviews, etc. As has previously been mentioned in Section 4.1.8, this component will communicate with the Intelligent Marketplace Back-End for provisioning of the different services needed to satisfy user requests.

4.1.12.2 034 Visual Simulation Engine

The visual simulation engine is a front-end visualisation client to the Digital Twinning Subsystem. It will be based on the VC4.0 commercial solution by VIS, and is aimed at realistically representing the DT using a rich 3D graphics engine, while also exposing graphical UIs with bindings to the Digital Twinning subsystem components for e.g., creating a virtual factory layout, or defining processes based on tasks. In addition, a statistics UI available in VC 4.0 will be provisioned, enabling a user to monitor the performance of the virtual factory according to the layout configuration created.

4.1.13 OMIDES Front-End

The OMIDES Front-End contains the different applications that will support operators in the manufacturing process by providing AR-enabled features realized through worn OPTIMAI-developed AR glasses (Task 5.3 "Wearable devices for real-time assistance on the production line"). More specifically, it will provide a graphical UI that will be displayed through the binocular smart glasses' lenses, presenting information in the form of intuitive visual analytics with regard to the quality level of the current production processes. The UI will be adaptable, based on users' profiles, and the overall context of use, ensuring that information is only provisioned if it is useful for the user at the given time. In this respect, it will present only the important information regarding a machine defect that could affect the production quality.

4.1.13.1 035 Rendering and AR Interaction Core Components

The Rendering and AR Interaction Core components will focus on facilitating the necessary interaction routines of the end-users with the AR glasses. To achieve that, a rule-based decision making, and integer linear programming component combination will be implemented as a computational approach in order to be able to decide the spatial and temporal placement of the various graphical elements to be displayed to the operator, as well as the level of detail of the information that those elements should present. Additionally, through a separate interaction actuation routine, users will be able to issue context aware gesture-based commands so as to be able to address deficiencies found in the production line.

Table 34 summarises the interdependencies of the Rendering and AR Interaction Core components with other OPTIMAI components, while Table 35 presents the data inputs and outputs of the component.

Table 34: Inter-dependencies of the Rendering and AR Interaction Core Components with other components

Component Name	Inter-dependency description
Visual Analytics and Graphical UI	Forwarding of the configuration for the visualisation of information on the smart glasses.
OMIDES Back-End subsystem	Receive application-pertinent data from each of the OMIDES Back-End components (e.g., gesture detected).
Manufacturing (re)-configuration Service	Facilitate the exchange of data regarding suggestions for configuration and results from the operator gestures as output to the (re)-configuration suggestion.

Table 35: Input/Output of the Rendering and AR Interaction Core Components

I/O Type	Data
Input	Data regarding visualization information to render, as well as notifications, suggestions, that need to be displayed to the operator (in JSON format).
Input	User input as gesture recognized from the vocabulary mapping at the Manufacturing (re-)configuration service.
Output	Rendering configuration to be used for the display of content on the AR glasses.

4.1.13.2 036 Visual Analytics and Graphical User Interface

The Visual Analytics & Graphical UI component aims to provide visual interfaces for the presentation of aggregated analytics results and suggestions in the form of a Heads-Up Display (HUD). The Visual Analytics tool will aim at enriching the AR environment and facilitate human-AI collaboration. This tool will support different visualisation formats, such as graphs, lines, scatters, radars, bars, clock views etc. These will follow the UI organisation rules outputted by the Rendering and AR Interaction Core Components, so as to provide visual decision support during manufacturing operations, that will timely inform workers on anomalies or upcoming defects directly on the HUD, while also adapting to the level of assistance each user will require.

Table 36 summarises the interdependencies of the Visual Analytics and Graphical UI with other OPTIMAI components, while Table 37 presents the data inputs and outputs of the component.

Table 36: Inter-dependencies of the Visual Analytics and Graphical User Interface with other OPTIMAI components

Component Name	Inter-dependency description
Middleware subsystem	Device management and data registration for input data.

Smart Quality Control subsystem	Obtaining and visualizing results from the Smart Quality Control components, algorithms and AI models.
OMIDES Back-End sub-system	Receive and visualize the appropriate data from each component (e.g., pose keyponts, segmentation masks).
Rendering and AR Interaction Core Components	Obtain the configuration for the visualisation of information on the smart glasses

Table 37: Input/Output of the Visual Analytics and Graphical User Interface

I/O Type	Data
Input	Data obtained from the analysis and processing conducted by the AI framework.
Input	Rendering configuration to be used for the display of content on the AR glasses
Output	The output will be diagrams and charts rendered on the AR HUD.

4.2 OPTIMAI alignment to standards-led reference architectures

Taking into account the various RAs existing for describing I4.0 solution architectures as described in Section 2, each can be used as a collection of guiding principles for concretising the design of smart manufacturing systems and applications. Presented in this Section at a functional level, the OPTIMAI components architecture resulting from both design steps and architecture workshops has been refined appropriately from its early rendition reported in the GA to be foremost aligned to the specifications of the RAMI 4.0, allowing us to address aspects, such as communication standards and functions of the various assets, using a common representation/general abstraction of I4.0 elements. A similar approach to the one followed in this Section has been reported in [16].

In this Section, we explore the parallels drawn between the first version of the OPTIMAI architecture and the RAMI 4.0, while also mapping the presented architecture against the IIRA, highlighting in which aspects the architecture maintains compatibility with the IIoT framework established therein. Additional mappings between the proposed architecture and other RAs presented in Section 2 can be developed, but such an exercise is deemed beyond the scope of the current deliverable.

4.2.1 Alignment to RAMI 4.0

RAMI 4.0, as specified in Section 2.2.1, is a cubic map intended to encapsulate the entirety of concepts and elements that make up the modern I4.0 smart factory environment. It delivers no concrete architecture or implementation guidelines, but rather, presents a structured approach to address the particulars of a given smart factory use case.

Interpreting OPTIMAI in the context of RAMI4.0, the foreseen UCs to be evaluated in the three pilot sites (and thus, the defined architecture for the proposed solution to those UCs' requirements) are:

- Reducing the number of quality defects in the production line (“Zero defect quality inspection” – UC1).
- Improving the efficiency of the production line by optimally calibrating machines/robotic cells in a way that decreases stoppages (“Production line setup-calibration” – UC2).
- Optimising the production of the manufacturing line by means of a digital twin where optimal product manufacturing sequence can be calculated for future planning purposes (“Production planning” – UC3).

It becomes apparent that these processes are created inside the ‘Instance’ phase of the Life Cycle & Value Stream axis defined in RAMI 4.0. Particularly, the processes described in the 24 sub-UCs defined in D2.6 fall within the ‘Production’ state, as the actions undertaken to deal with quality issues during manufacturing execution, and formulating knowledge for ideally setting up the manufacturing environment. Hence, with the placement in one of the three axes specified, OPTIMAI can be layered on top of a 2D slice extracted from the cubic map (Figure 9) presenting a layer-and-hierarchy mapping in terms of “Production”.

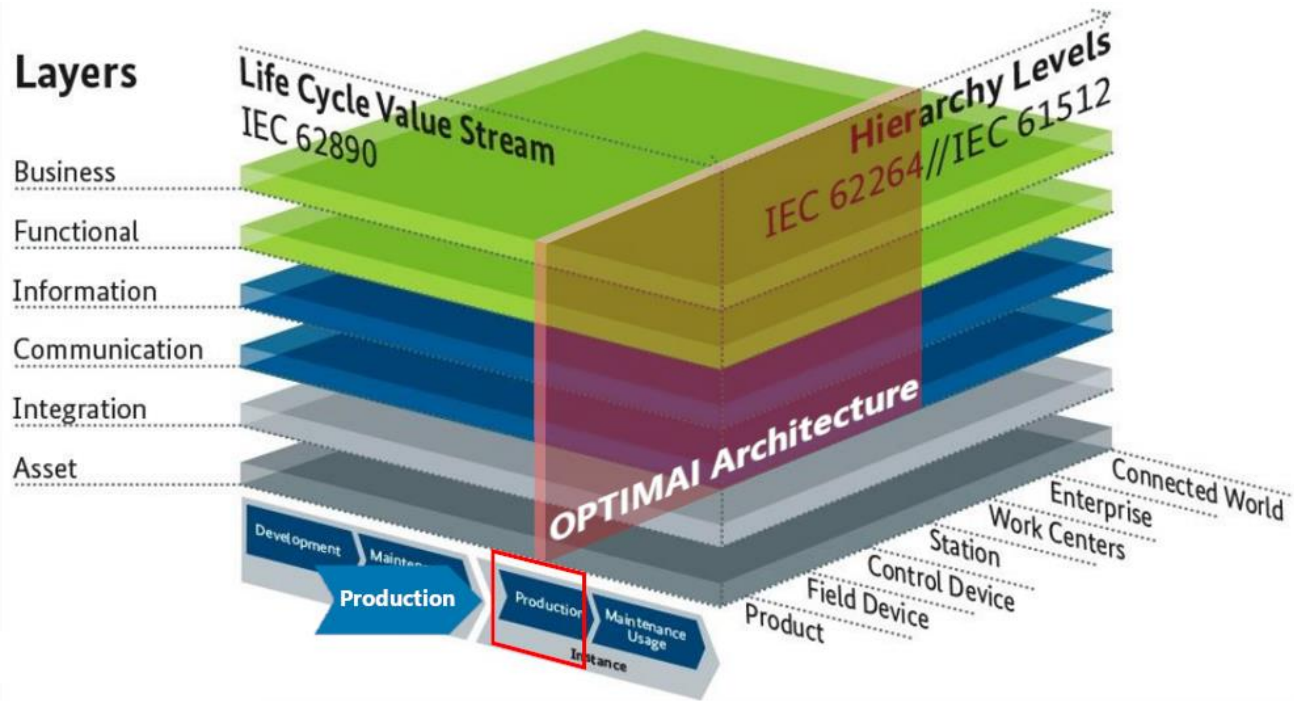

Figure 9: OPTIMAI placement within the RAMI 4.0 cubic model. Adapted from the original Graphic © Plattform Industrie 4.0 and ZVEI, retrieved from [4]

Based on the described operations of the functional components presented in the previous Section, the OPTIMAI architecture is mapped onto the 2D layer-and-hierarchy slice as shown in Figure 10, thus enabling the compatibility with the RAMI 4.0 guiding principles to be better illustrated.

As can be seen, OPTIMAI defines components across all Layers and Hierarchy Levels for the “Production” state of the ‘Instance’ Life Cycle & Value Stream Phase:

- The QCS Network subsystem (functional blocks 001-008) quite logically maps to the ‘Field Device’ and ‘Asset’ intersection, as the components defined therein constitute sensory apparatus installed on the shop floor for collection of data and actuation based on calculated parameters.

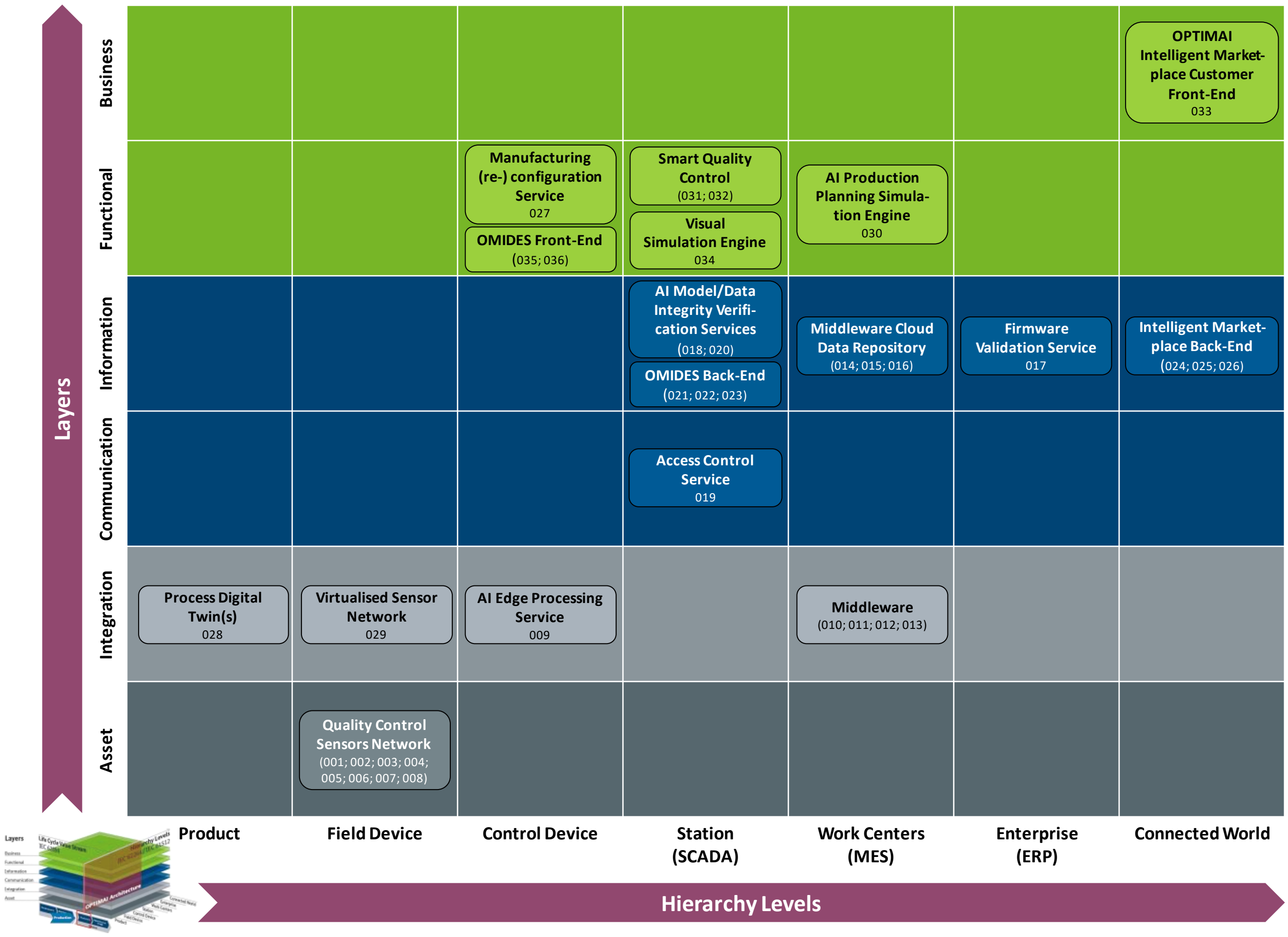

Figure 10: OPTIMAI Layer-and-Hierarchy mapping to RAMI 4.0.

- The AI Edge and Processing Service (component 009) is mapped onto the 'Control Device' Hierarchy Level, as deals with edge-based calculation and delivery of control parameters in an attempt to optimize acquisition and actuation. It further aligns to the 'Integration' RAMI 4.0 Layer, as it facilitates a direct interaction of readings from the Quality Control Sensor Network with edge-based AI models.
- The Middleware subsystem (components 010-013) similarly maps to the 'Integration' Layer as it underpins the communication of the entities in the real world with the higher-level software components of the architecture. Dealing heavily with the exchange of data among functional units, sensors and subsystems, the Middleware components are aligned to the functional areas defined within the responsibility of a MES, e.g., the 'Work Centers' Hierarchy Level, with the added benefit of implementing security mechanisms as early as the data acquisition process (i.e., as soon as data enters the system).
- The Middleware Cloud Data Repository (blocks 014-016) complements the aforementioned functional area of MES regarding collection and storage of process and production data occurring within the 'Work Centers' Hierarchy Level, while mapping to the 'Information' Layer where RAMI 4.0 considers structured data storage.

- As blockchain technology is Components of the Blockchain subsystem are mapped onto RAMI 4.0 according to their provisioned functions. Three of the subcomponents (017, 018 & 020) are naturally defined within the 'Information' Layer, for dealing with the distributed ledger architecture (e.g., a database spread across locations) for storing records (017) and processing of events where integrity of the data and processes (018, 020) plays a crucial role [17]. However, validation of firmware and software installed on sensors and OPTIMAI middleware (017) relates to functional capacities of ERP systems to provide unified and centralised device management (aligning 017 to the 'Enterprise' Hierarchy Level), while checks on data integrity are a crucial component in the data acquisition processes occurring at SCADA systems [18] (placing 018 and 029 within the 'Station' Hierarchy Level). Finally, Access Control functionality refers to a cybersecurity perspective of dealing with security challenges related to attack vectors to communication methods over the network [17], particularly relevant to the data acquisition part of modern SCADA systems. Hence, functional block 019 is mapped at the intersection of the 'Communication' Layer with the 'Station' Hierarchy Level.
- The OMIDES Back-End components (021-023) relate to processing of information for determining both: (i) higher-order information on raw data (e.g., pose and labelling information out of raw image data – 021 & 023); as well as (ii) information on performed gestures by a worker used to facilitate commands during supervision of machines via novel HMI equipment (022). These attributes place the subsystem at the intersection of the 'Information' Layer and the 'Station' Hierarchy Level. The OMIDES Front-End on the other hand (functional blocks 035 & 036), combines with the Manufacturing (re-)configuration Service component (027) to deliver decision-making on the information produced by 022 toward applying automated and manual re configurations of individual machine parameters. These modules are therefore found at the 'Functional' Level, and relate to processes described in the 'Control Device' Hierarchy Level.
- The Intelligent Marketplace components comprised of the Back-End (blocks 024-026) and Customer Front-End application (033), provide a connection between the factory and the outside world (third parties) thus mapping clearly to the 'Connected World' RAMI 4.0 Hierarchy Level extension. It becomes further apparent that block 033 refers to an application designed to influence decisions at the strategic level (such as the acquisition/reuse of scrap produced by another organisation), while the Back-End subsystem deals with recording of transactional information and feedback. They are thus found within the 'Business' and 'Information' Layers respectively.
- Based on the observations made in [12] regarding the vague nature of DTs in the scope of RAMI 4.0, several attempts have been proposed in the scientific literature to facilitate DT architectures conforming to RAMI 4.0 principles. In our own approach we mapped the OPTIMAI Digital Twinning subsystem components (blocks 028-030) in accordance to the Generic Digital Twin Architecture (GDTA) proposed in [19], where clear mappings to the OPTIMAI components can be drawn. Hence, the simulation models of processes (028) and sensor network (029), tasked with the generation of data that emulates the operation of real-world entities (i.e., as if data was gathered by actual equipment on the shop floor) are placed in the 'Integration' Layer, identified as either "Engineering" or "Runtime" data

respectively within the GDTA. Simulation services, such as those provisioned by the AI Production Planning (component 030) and Visual Simulation Engines (block 034) are delivered at the 'Functional' Layer, as they represent instruments for driving decision-making based on availability of information (simulated, in this case). While GDTA does not align its architecture to the Hierarchy Levels dimension of RAMI 4.0, purporting that DTs can be defined for various hierarchical levels, OPTIMAI, being use-case driven, identifies the following mappings to the Hierarchy Levels axis:

  - Process Digital Twin(s) (028) refers to the concept of "Engineering" Data, i.e., topological information about the production plant that should be taken into account as they can affect production parameters (such as the distance a worker has to travel from one area to the next in a manual workstation). Hence, we can define those data as originating in the 'Product' Level, referring to the production facilities and the interdependencies they impose [5].
  - The Virtualised Sensor Network (029) supplements the functionality provisioned by the QCS Network, similarly placing it within the 'Field Device' Level.
  - The AI Production Planning Simulation Engine (030) refers to the overall execution/simulation of processes occurring at the 'Work Centers' Level, as it simulates various functional areas associated to MES' scope.
  - The Visual Simulation Engine (031) represents a rich UI provisioned for the supervisory control part of SCADA systems, which is aimed at enabling monitoring of real-time data generated by the simulation and the direct interaction with the DT devices and sensors through emulated HMI software. Hence, for OPTIMAI purposes, this component is found at the 'Station' Level.

- Finally, Smart Quality Control components (031 & 032) perform the necessary processing checks and deliver decision support toward defect identification and detection events notification based on real-time data generated by other modules in the OPTIMAI architecture. They are hence a crucial component to the 'brain' of the OPTIMAI solution, and are defined at the intersection of 'Functional' Layer and 'Station' Hierarchy Level.

4.2.2 Alignment to IIRA

As has been mentioned in Section 2, IIRA and RAMI 4.0 are significantly similar in their support of SOAs, i.e., decomposition of system functionality into an array of interconnected services. Because of the service-oriented approach followed in the OPTIMAI architecture, as well as its overt alignment to RAMI 4.0 described in the previous Section, by extension OPTIMAI can significantly be parallelised to the IIRA as well. Regarding the IIRA Viewpoints, with respect to the OPTIMAI project, the following are thoroughly defined in the context of OPTIMAI:

- The *Usage Viewpoint* of the OPTIMAI architecture is described in detail in deliverable D2.6, which precedes the present document in specifying the expected usage of the OPTIMAI system. In specifying the use cases in detail prior to the architecture (i.e., taking use case description into account when designing the ICT systems to support each case), OPTIMAI aligns to the principle of IIRA for top Viewpoints to guide the design of the viewpoint directly below.

- The *Functional Viewpoint*, which is reflected in the specification of the OPTIMAI system functional blocks and their correspondences to roles and responsibilities (Section 4.1), related to the functions they are expected to perform to support the use cases. As IIRA and RAMI 4.0 intersect at the IIoT systems for smart manufacturing domain, RA alignment principles are established for the functional mapping of the IIRA and RAMI 4.0 [8], which are iterated in Figure 11. Based on these mappings, correspondences between the OPTIMAI architecture (from an IIoT solution perspective) and the IIRA are shown in Figure 12.

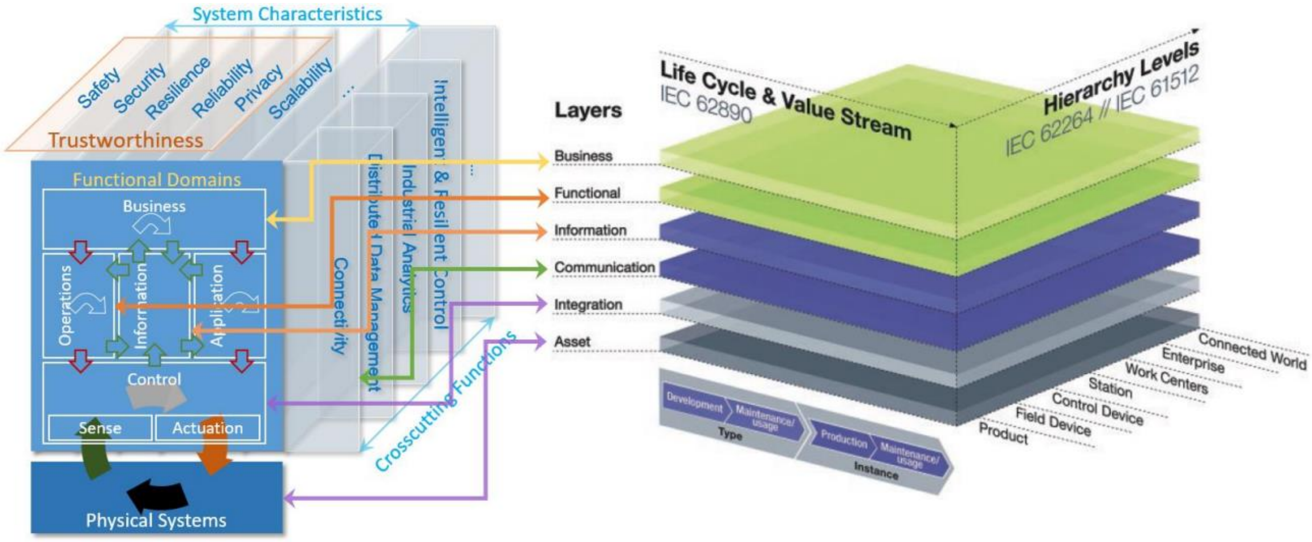

Figure 11: mapping between the IIRA Functional Viewpoint and IT layers established in RAMI 4.0. Source: [8]

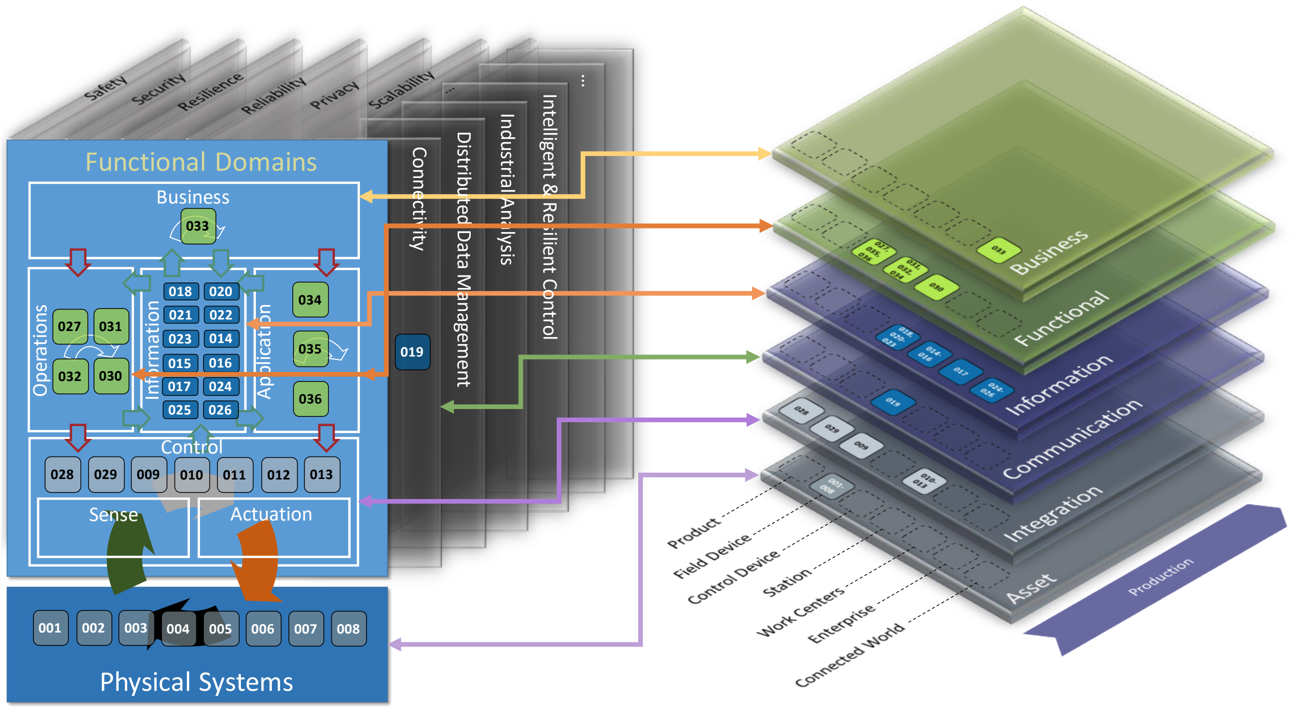

Figure 12: Functional mapping of the OPTIMAI Architecture to IIRA based on the alignment to RAMI 4.0.

More specifically, following the guidelines established in Figure 12, the OPTIMAI architectural components can be mapped to the following IIRA *Functional Domains*, as indicated in the following Table (Table 38):

Table 38: Mapping between IIRA Functional Domains and OPTIMAI functional components.

IIRA Functional Domain	Description	OPTIMAI Functional Blocks
Physical Systems	Despite not being formally established as a Functional Domain in IIRA, Physical systems are directly mapped onto the RAMI 4.0 Assets Layer, and are understood as the physical resources on the factory shop floor.	**001, 002, 003, 004, 005, 006, 007, 008**
Control domain	Includes components, whose functions deal mainly with the control, sensing and actuation on the physical systems. Such functions involve the collection of data from sensors, potential application of logic and eventual execution of actuation commands. Because of this property, within the IIRA, these assets are considered to be in close "proximity to the physical systems they control" [7] (i.e., the Network Edge). Identifies the following functions: • **Sensing:** Read data from sensors. • **Actuation:** Apply configuration to a hardware component. • **Communication:** Connect to external entities. • **Entity abstraction:** Sensor digital representation (digital twin). • **Modelling:** Monitoring of operation. • **Asset management:** Exercises management over the hardware. • **Executor:** Exercises logic to facilitate control (at the edge-level).	<u>Sensing</u> **010** <u>Actuation</u> **010** <u>Communication</u> **012, 013** <u>Entity Abstraction</u> **028, 029** <u>Modelling</u> **009, 011** <u>Asset Management</u> **011** <u>Executor</u> **009, 011**
Operations domain	Exercises monitoring, management and control over the assets in the Control domain. Essentially, this domain deals with operations regarding decision-making based on data capturing, processing and valida-	<u>Provisioning & Deployment</u> **027** <u>Asset Management</u> **027**

	tion. Hence, it is directly mapped to the functions defined within the RAMI 4.0 Functional Layer. Identifies the following functions: • **Provisioning and Deployment:** Configure and deploy/retire assets from the network • **Asset Management:** Intelligently manage assets' processes. • **Monitoring & Diagnostics:** Detection and prediction of issues. • **Prognostics:** Predictive analytics. • **Optimisation:** Improve asset performance.	<u>Monitoring & Diagnostics</u> **031** <u>Prognostics</u> **032** <u>Optimisation</u> **030**
Information domain	Deals with the management, processing, transformation and storage of data. Maps directly to the RAMI 4.0 Information Layer. Identifies the following functions: • **Analytics:** Data modelling and rules' application. • **Data:** Pre-processing and persistence of data.	<u>Analytics</u> **018, 020, 021, 022, 023, 025** <u>Data</u> **014, 015, 016, 017, 024, 026**
Application domain	Deals with functions that support application-specific logic. Within the IIRA it differentiates from the Operations domain by being use-case-specific, yet, its functions (listed below) maintain mapping to the RAMI 4.0 Functional Layer. • **Logic and Rules:** implements specific rules and models tied to a specific use case. • **APIs and UI:** Refers to the interfaces exposed towards other functional entities, including human operators.	<u>Logic & Rules</u> **034** <u>APIs & UI</u> **034, 035, 036**
Business domain	Deals with functions that implement business processes. Maps directly onto the RAMI 4.0 Business Layer.	**033**

Not depicted in the above Table, the RAMI 4.0 Communication Layer (in which OPTIMAI component 019 is defined) is mapped onto the Connectivity Crosscutting Function defined in IIRA, reflecting the need for a particular security function to be implemented across communications of the functional components [7] (in OPTIMAI's case, Access Control).

Finally, some elements related to the *Implementation viewpoint* can be seen in the descriptions of both the technologies' selected technical components (e.g., sensor hardware) and communication schemes (e.g., OPC-UA), while the strategic goals and benefits driving the system implementation and deployment across the three industrial pilot sites (i.e., the IIRA *Business Viewpoint*) are reflected in the Key Performance Indicators (KPIs) elaborated for each case in D2.6.

4.2.3 Key takeaways

The alignment exercise elaborated in this Section delivers a foundation for implementing the OPTIMAI framework in accordance to the RAMI 4.0 and IIRA standardised specifications. Through this process, specific cross-cutting system aspects, such as inter-communication of components, integration and interoperability, can be addressed in a more harmonised manner, applying the principles reported in the documentation of the two RAs. Through the demonstration activities planned further into the project's lifetime (WP7), OPTIMAI further gains the benefit of showcasing a pragmatic implementation of RAMI 4.0 (which is a generalised framework) in the context of zero-defect manufacturing and zero-waste production planning, which can extend to other use cases outside the scope of the project. Therefore, impact of the OPTIMAI solution and its architecture is increased.

5 OPTIMAI Architecture - Information view

The Information view of the OPTIMAI architecture specifies the information flow among the functional blocks defined in Section 4. In this respect, this Section will define synchronisation and synergies formed between individual blocks of the envisioned system, with particular focus at the manner in which information is communicated from one component to another.

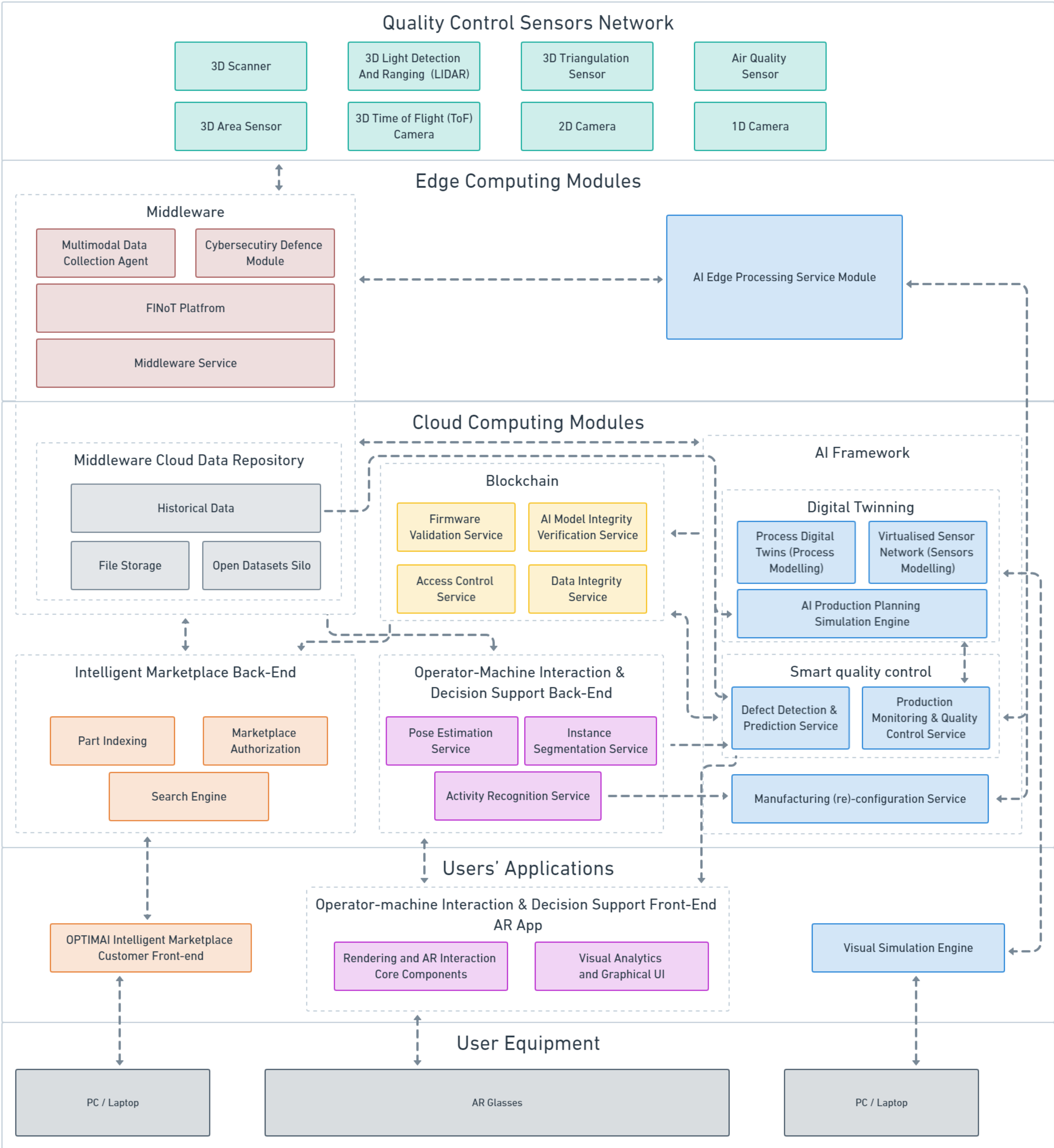

Figure 13: OPTIMAI high-level Information Flow Diagram.

5.1 Overview information flow

Figure 13 presents an overview of the OPTIMAI Information Flow Diagram (IFD) layered on top of the functional architectural viewpoint, hence depicting the manner in which information flows through the system between locations, ether within the same, or between federated computer systems. The goal is to define the manner in which information is exchanged between OPTIMAI functional blocks, as well as with potential external entities (e.g., user equipment). All foreseen data exchanges are represented following the producer-consumer paradigm [20], in which any one of the interdependent functional blocks can assume the role of a producer (of data), a consumer, or both. This paradigm will be useful in the second phase of the architecture definition procedure (M12-M18) for defining concrete interfaces among the identified functional components, which will describe in detail the functions that a "producer" block exposes to its interdependent "consumer". Specification of these interfaces (in terms of message exchange formats, request and response schemas etc.) will be elaborated in the updated version of the present deliverable (D2.5), which will thus describe the complete reference frame for the final system implementation. Activities toward this direction will be undertaken in the context of T2.3.

5.2 Use Case-specific information flow

This section presents several indicative high-level IFDs, aimed at clarifying how information flows through the system so as to enable specific functionality pertinent to the three use case types defined for each industrial pilot site in the project. These IFDs are meant to elaborate on the high level overview of the flow of information presented in the previous paragraph, and provide a simple, yet detailed overview of the overall synergies and interdependent component flows created for individual process delivering on specific use-case functionalities. Each diagram is carefully designed to demonstrate the roles and responsibilities of the different components in the OPTIMAI architecture with respect to the use case in question. We omit to include a description of the UCs themselves, as this information is available in deliverable D2.6.

5.2.1 UC1 - Zero defect quality inspection

Figure 14 presents an illustrative example of the information flowing through the OPTIMAI system during a typical "Zero defect quality inspection" use case (for any one of the three industrial pilot sites), as defined in D2.6. During this particular task, the system is implied to utilise sensors installed in the production line (e.g., 001-008) and notifying a Production Control Engineer (PCE) equipped with AR glasses in case a defect is detected. Hence, the actor is assisted in monitoring and expediting activities toward maximising the amount of non-defective parts or products produced.

The system receives data from the sensors and translates it into context aware entities for further (pre-)processing (**1**). Then, it conducts analyses on the data at the edge through the Middleware subsystem components to form aggregated time series ("important measurement") data, usable within the OPTIMAI system (**2**). It then brokers the data toward the Cloud and invokes data registration routines at the Blockchain to validate integrity of the data (**3**, **4**). If data integrity is verified, the system stores the data in the appropriate Middleware Cloud Data Repository entity (**5**).

The Defect Detection & Prediction Service pulls the data in order to detect whether a part under being manufactured is defective (**6**, **7**). If the part under inspection is found to be indeed defective, the system proceeds to inform the operator through a notification on the OMIDES Front-

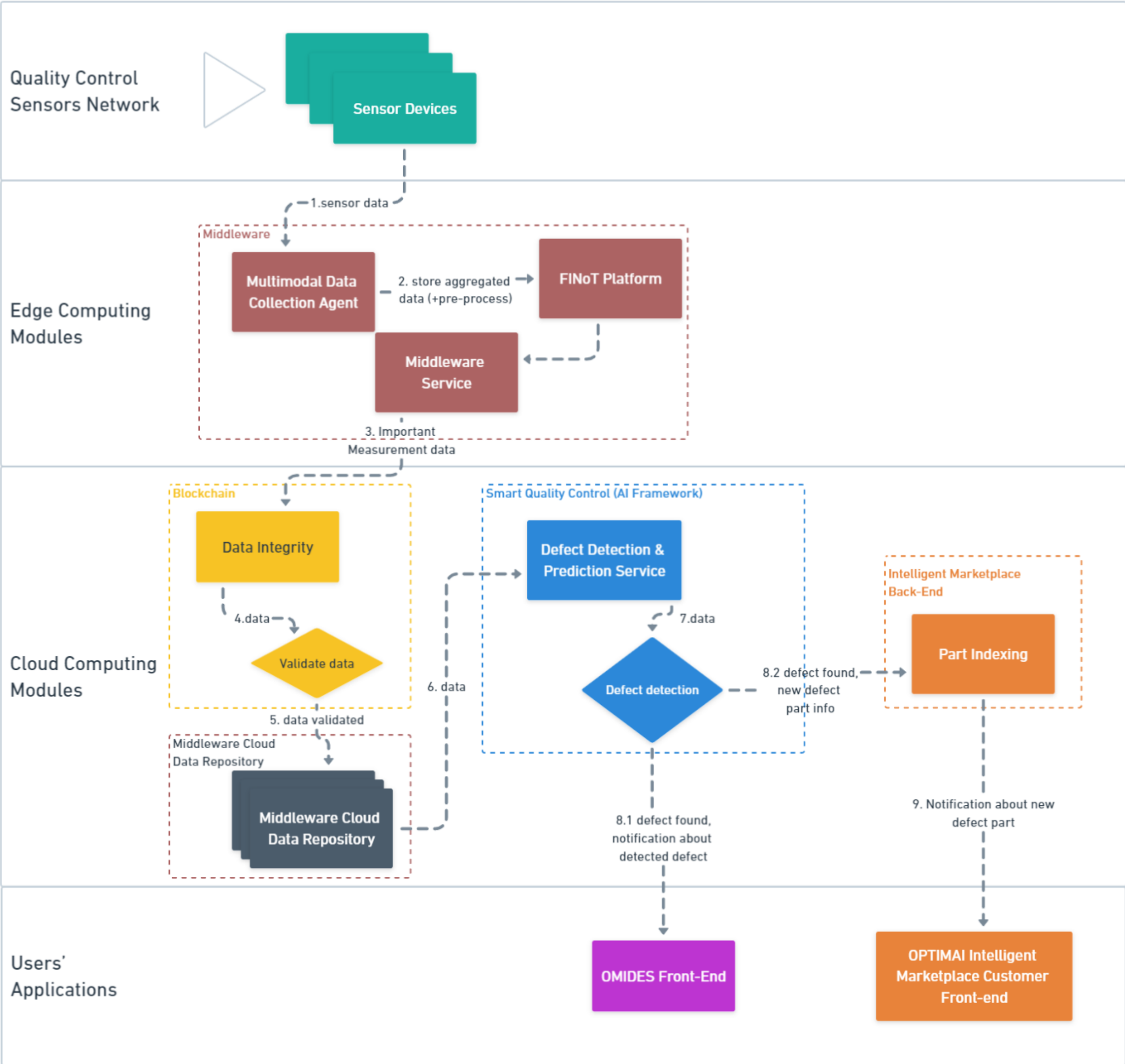

Figure 14: Information Flow Diagram for an indicative "Zero-defect quality inspection" task.

End (**8.1**); and simultaneously registers the defective part on the Intelligent Marketplace Back-End for indexing (**8.2**). The latter block then creates a notification to the Intelligent Marketplace Customer Front-End for further development of the part listing through the administrator's dashboard, so as to make it available to third parties (**9**).

5.2.2 UC2 - Production line setup-calibration

Figure 15 presents an illustrative example of the information flowing through the OPTIMAI system during a typical "Production line setup-calibration" use case (for any one of the three industrial pilot sites), as defined in D2.6. In this particular example, an operator equipped with smart glasses can initiate a check on the possibility of defects being introduced to the manufactured goods using the current configuration at the actor's station, and make informed decisions in decreasing the likelihood of that happening.

During this particular task, the system receives data from the sensors and proceeds to repeat steps **1-5**, as described in the previous Section, toward applying data registration and integrity verification checks as early as the acquisition stage.

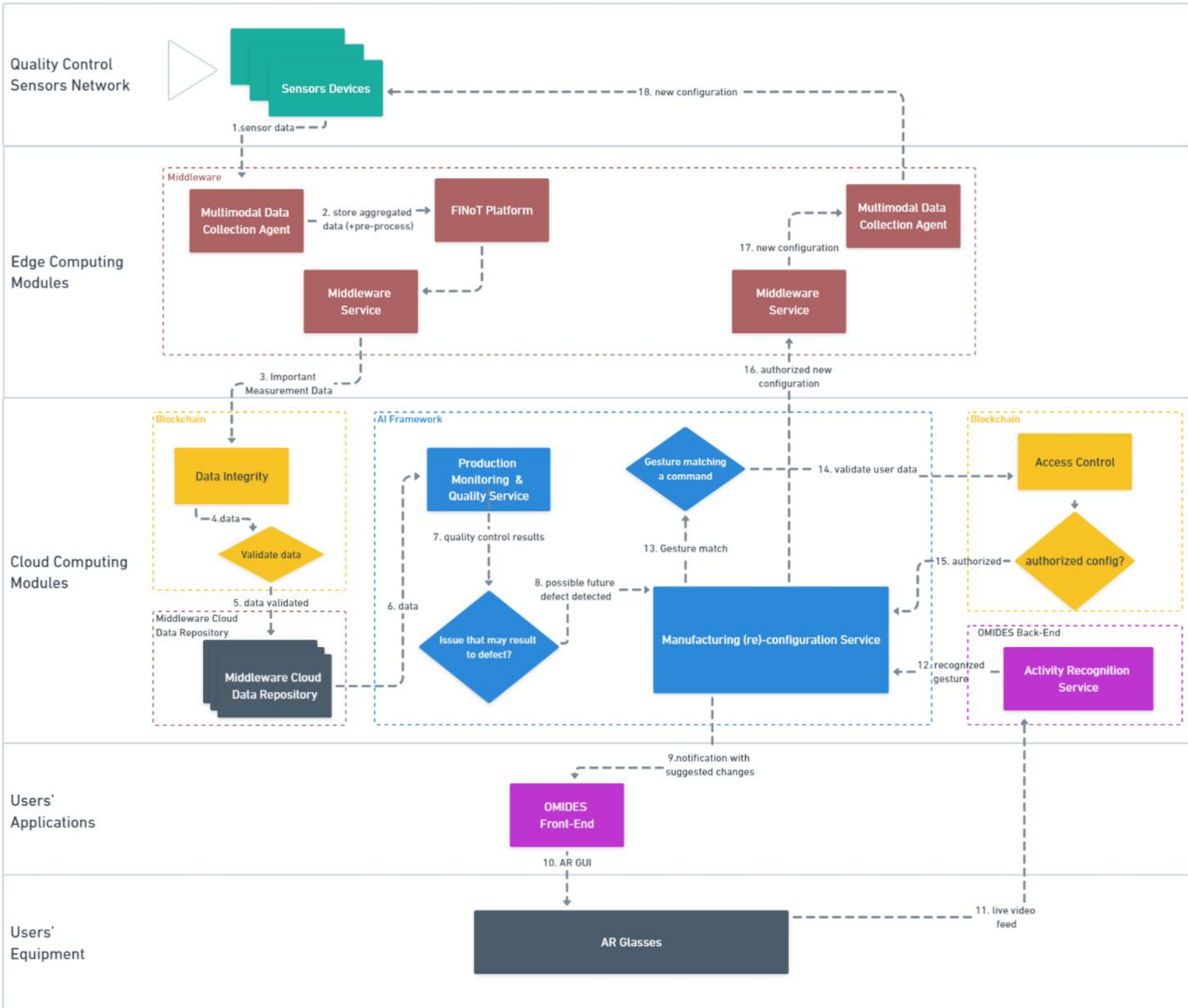

Figure 15: Information Flow Diagram for an indicative "Production line setup-calibration" task.

The Production Monitoring and Quality Service pulls the data in order to determine the likelihood of a defective part being produced using the current environment configuration (**6**, **7**). In case the present configuration is found to be prone to a defective part being manufactured above a tolerable threshold, the Manufacturing (re)configuration Service is invoked on 'manual' mode, to determine the most appropriate re-adjustment of equipment parameters based on the received quality control results (**8**), and forwards suggestions to the operator using the OMIDES Front-End regarding the newly discovered configuration conditions (**9, 10**).

The user will approve of all, or part of the configuration recommendations by interacting with the OMIDES Front-End performing the gestures, with the video frames captured by the AR Glasses being forwarded to the Activity Recognition Service to determine the users' intended actions (**11**). The latter module forwards the gesture recognition results to the Manufacturing (re)configuration Service (**12**), which then maps gestures in accordance to the vocabulary mappings drawn to configuration actuation commands necessary to approve of the reconfiguration (**13**). As an additional security check, the system verifies the authority of the operator in deploying the proposed configuration via the Blockchain Access Control service (**14**), which either approves of, or rejects the configuration if the operator is not cleared for executing actuation (**15**). On approval, the Middleware receives the new configuration through the Middleware Service (**16**), and proceeds to execute actuation by applying the new configuration parameters at the specified sensors/machines (**17**, **18**).

5.2.3 UC3 – Production planning

Figure 16 presents an illustrative example of the information flowing through the OPTIMAI system during a typical "Production planning" use case (for any one of the three industrial pilot sites), as defined in D2.6. In this particular example, a PCE, Logistics Planner or Production Manager is able to experiment with different plant configurations in order to derive important decisions for production optimisation.

During this particular task, the system receives data from the sensors and proceeds to repeat steps **1-5**, as described in Section 5.2.1, toward applying data registration and integrity verification checks as early as the acquisition stage.

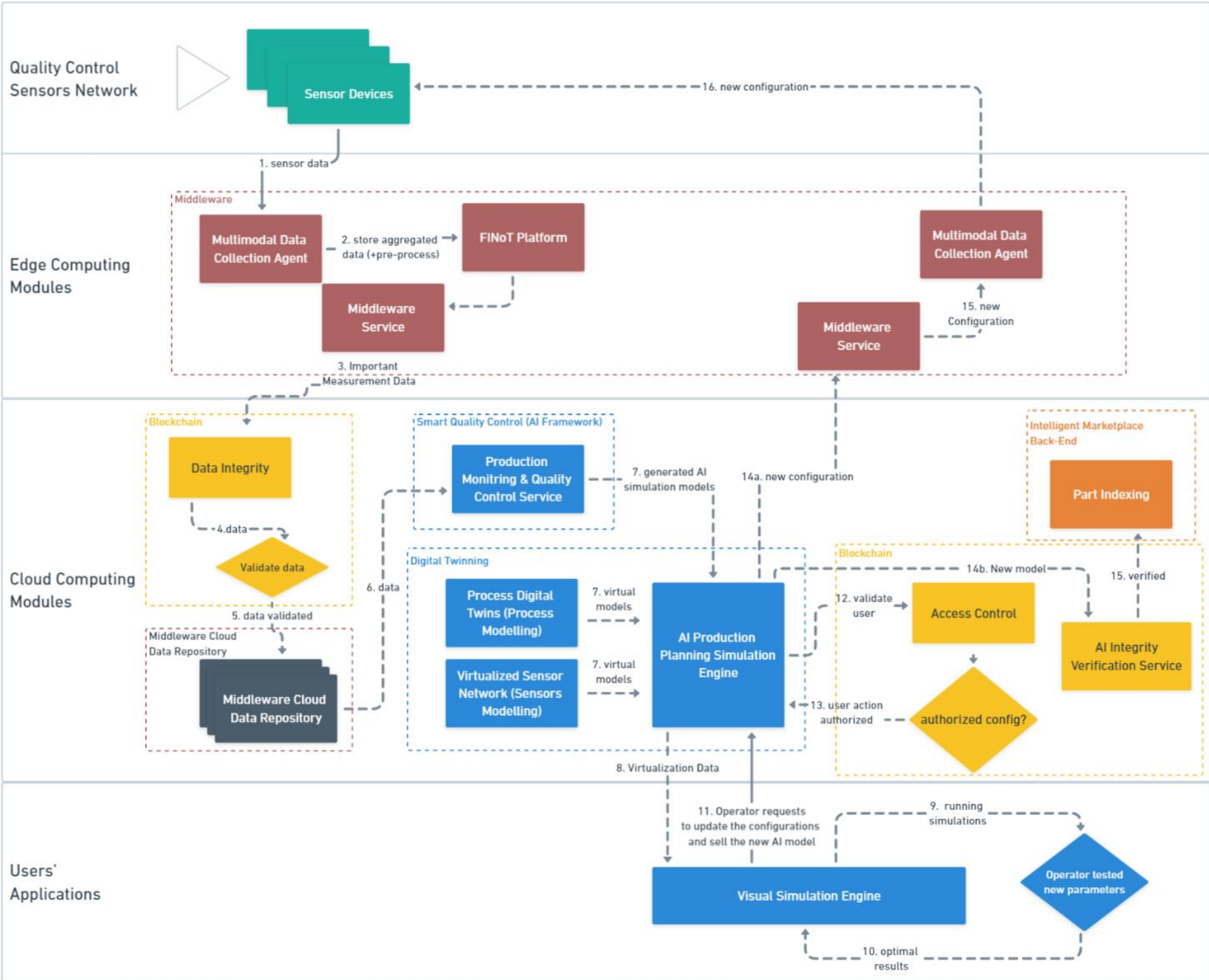

Figure 16: Information Flow Diagram for an indicative "Production planning" task.

An operator proceeds to simulate the current production process by modifying specific configuration parameters of sensors and/or machines to test against the currently applied configuration using the Visual Simulation Engine front-end. To achieve this, within the OPTIMAI system the Production Monitoring and Quality Service is invoked, which pulls the data in order to derive the new AI Model for improved quality control routines (**6**). The Model is fed to the AI Production Planning Simulation Engine, which initiates the simulation using the process model and initiating and receiving readings from the Virtualised Sensor Network (**7**). Simulated processes are executed on the production line's DT infrastructure, generating new data which are displayed to the Visual Simulation Engine (**8**), where the operator can modify in real-time various parameters to determine under which conditions results of the manufacturing line can be optimised (**9**, **10**).

The operator may approve of the new configuration, and simultaneously render the new AI model available for the OPTIMAI Intelligent Marketplace (**11**). Upon approval of the new configuration by the operator, the system applies an additional check (as was the case in steps 14-15 described in the previous Section) so as to verify operator authority in both deploying the proposed configuration and forward the AI Model to the marketplace on behalf of the organisation (**12**, **13**). Upon approval, the system executes the actuation by applying the new configuration parameters at the specified sensors/machines similarly to the processes described in Steps 16-18 of the previous Section (**14a**, **15a**, **16**); and also proceeds to verify the AI Model integrity (**14b**), for making the algorithm available on the Intelligent Marketplace Back-End for indexing (**15b**).

6 OPTIMAI Architecture - Deployment view

Typically, in software system architectures following the IEEE 1471 "Recommended Practice for Architectural Description for Software-Intensive Systems" [21], the 'Deployment' view specifies the physical deployment of the described system's components across physical and virtualised computing resources, together with their hardware and software requirements. Using this viewpoint, the system developers and integrators are in a better position to determine: (i) the number and specifications of the hardware necessary for the execution of particular system components; (ii) the location where these entities will be deployed; and (iii) means to map software blocks to hardware components. The means by which this information is conveyed is through a *deployment diagram*.

The present deliverable's main purpose is to describe the overall functional and information viewpoints so as to provide OPTIMAI developers and integrators a proper set of guidelines for the continued implementation of functionality needed by the independent functional units, alongside the functions these should expose to other blocks and subsystems. As such, the current version of the functional architecture is provided at a purely functional level, i.e., the contents in Sections 4 and 5 of the present deliverable neither imply nor impose a specific implementation as of yet. For this reason, the complete description of the deployment diagram nodes and artefact types (such as executable applications, software libraries, scripts, etc.) that partake in the execution of the system cannot be specified at the present time.

Instead, the current document will discuss foreseen topological considerations in the framework of the Industry 4.0 smart factory framework tangible layers [22]. These constitute:

- **Physical resources.** Constitutes the various kinds of "smart" physical resources, e.g., machinery, products and sensors/actuators, that make up the IIoT network.
- **Industrial network.** Refers to the wireless network deployed for the support of both the intercommunication of physical resources as well as the connection of those resources to the Cloud.
- **Cloud.** Delivers Infrastructure-, Platform- and Software-as-a-Service (IaaS, PaaS, SaaS) solutions by virtualising both computing and storage capabilities that can scale in and out on demand. Within the context of OPTIMAI, this layer encompasses either *private* (i.e., on-premise/internal) or *public* (i.e., over the Internet) Cloud services, depending on strategic (e.g., the business already investing in on-premise data centres), or security concerns.
- **Supervisory control terminals.** Includes all end-user terminals (e.g., PCs, laptops, smart surfaces, head-mounted displays, etc.) that can access processing results and storage data housed within the Cloud, and display it to an end-user via UI/HMI solutions.

For the purposes of OPTIMAI deployment, we proceed to extend this framework by introducing an **Edge** layer between the Physical resources and the Industrial network layers, reflecting the more recently introduced capacities afforded by the edge computing paradigm.

Based on this framework, the proposed topological and deployment architecture of OPTIMAI is illustrated in Figure 17.

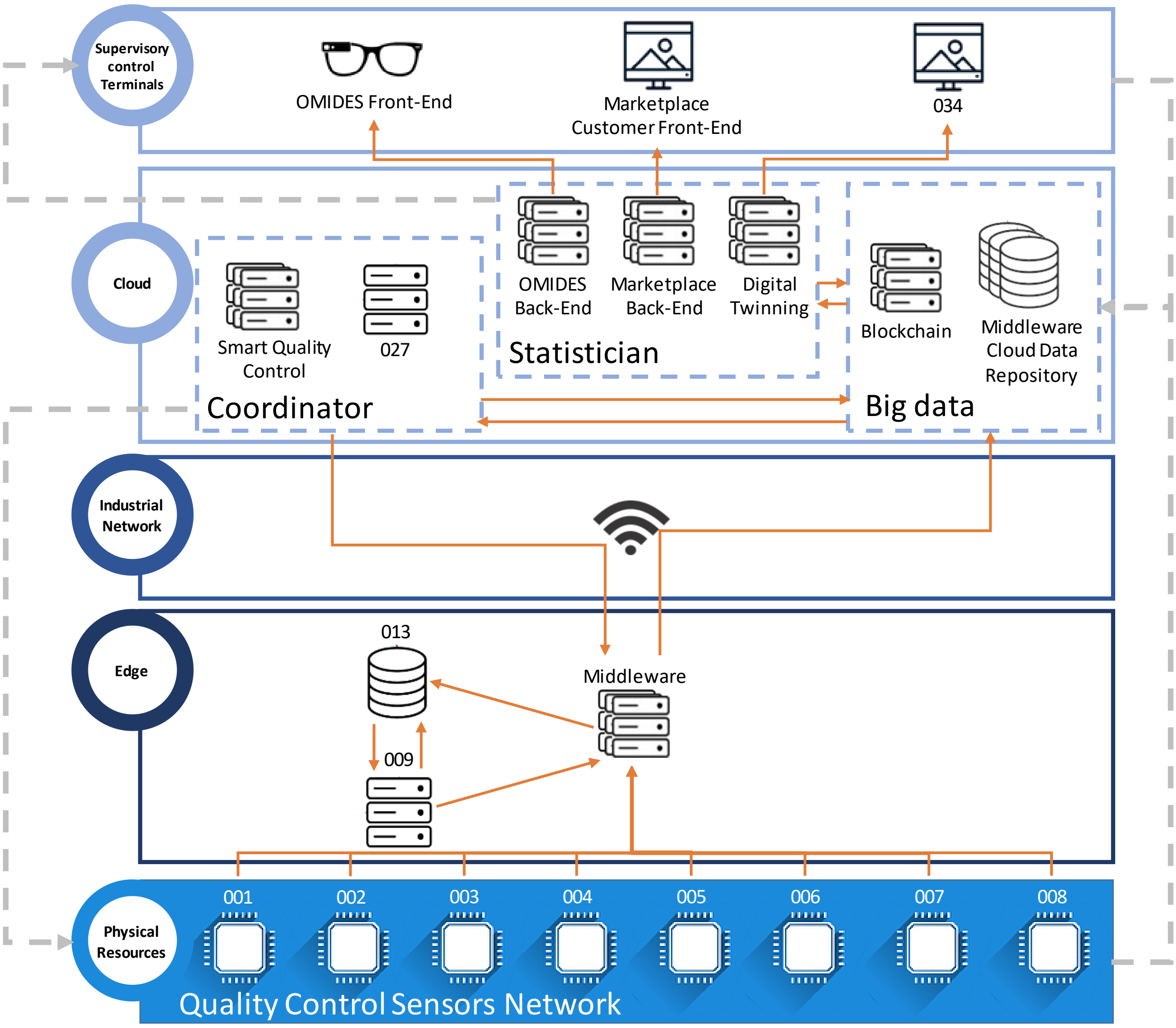

Figure 17: Topological view of OPTIMAI architecture through the smart factory framework perspective.

The depicted view organises components in the different smart factory framework layers, and their relation to the operational mechanism dual loop closed system [23]. The first loop considers elements that are involved in the coordination and feedback provided at the Cloud-level toward reconfiguring assets found in the Physical Resources Layer ("Coordinator"), while the second regards data visualisation and manipulation loop manifested between the Cloud components engaged in statistical analysis ("Statistician"), and the supervisory terminal applications. Big data storage provisioned on the Cloud plays a central role in each loop, facilitating both sensing and actuation, as well as control manipulation processes in the smart factory framework.

The contents in this Section aim at delivering an initial perspective for the foreseen deployment of the system components. D2.5 will update on this view by delivering the full deployment diagram, complete with hardware specifications and software dependencies for artefact deployment.

7 Conclusions

In this deliverable, an overview of the first version of the OPTIMAI architecture was presented, complete with the detailed specification of all of its functional elements, subsystems and main interaction rules and principles. The present document is considered to complement the contents of D2.3 by resenting a thorough overview of the I4.0 RAs that aim to guide the specification of interoperable systems for smart manufacturing use cases.

Building on the knowledge created by the technology exploration activities, along with the acquired knowledge of the use cases and user requirements, the OPTIMAI architecture was specified through a top-down approach (for breaking down the overall system into its most basic of components, identifying pertinent subsystems and their functionalities along the way), followed by a bottom-up specification of each eventually identified basal component. In total, 36 core elements are reported in this version of the architecture, comprising a total of 13 subsystems. Interdependencies between identified components and subsystems have been recorded so as to guide the implementation of the respective modules by the development teams dispersed across the technical Work Packages. Part of the architecture's evolution from the project preparation stage, is the approach undertaken at mapping the components to the RAMI 4.0, which further hints at specific principles and guidelines regarding component implementation along with integration aspects.

The document further specifies the foreseen flow of information through the system's components by indicating the aspect of data movement for indicative scenarios in each of the three use case types defined for the OPTIMAI project. This allows us to efficiently visualize how the provisioned services will communicate information within the confines of the use case, as well as provide useful sketches for the currently envisioned interactions between core functional blocks in the system design.

Finally, this document presented an early view on system deployment to enable the considerations of specific provisions (e.g., real-time processing requirements of some algorithms intended to be deployed as close to the physical resources as possible) at a sufficiently early stage in the project lifetime.

The presented first version of the OPTIMAI architecture has been defined in the period spanning M1 and M12 of the project, and has been the outcome of careful consideration given to the inputs collected from the technical work packages (*i.e.*, Work Packages 2 through 6), the requirements from D2.1 and the UCs from D2.6.

An updated (final) version of this architecture will document all introduced revisions and modifications proposed to components' descriptions, along with updated details regarding the presented architectural viewpoints (Functional Viewpoint, Deployment viewpoint and Information viewpoint).

References

[1] DIN, DKE. (2020). German Standardization Roadmap Industrie 4.0 Version 4.0. *DIN and DKE Roadmap*. URL: https://www.din.de/en/innovation-and-research/industry-4-0/german-standardization-roadmap-on-industry-4-0-77392

[2] Monostori, L. (2014). Cyber-physical production systems: Roots, expectations and R&D challenges. *Procedia CIRP, 17*, 9-13.

[3] Nakagawa, E. Y., Antonino, P. O., Schnicke, F., Capilla, R., Kuhn, T., & Liggesmeyer, P. (2021). Industry 4.0 reference architectures: State of the art and future trends. *Computers & Industrial Engineering, 156*, 107241.

[4] Plattform Industrie 4.0 (2018). *Reference Architectural Model Industrie 4.0 (RAMI4.0) - An Introduction*. URL: https://www.plattform-i40.de/IP/Redaktion/EN/Downloads/Publikation/rami40-an-introduction.html

[5] Pisching, M. A., Pessoa, M. A., Junqueira, F., dos Santos Filho, D. J., & Miyagi, P. E. (2018). An architecture based on RAMI 4.0 to discover equipment to process operations required by products. *Computers & Industrial Engineering, 125*, 574-591.

[6] Baptista, L. F., & Barata, J. (2021). Piloting Industry 4.0 in SMEs with RAMI 4.0: an enterprise architecture approach. *Procedia Computer Science, 192*, 2826-2835.

[7] Industrial Internet Consortium. (2019). *The Industrial Internet of Things Volume G1: Reference Architecture. Version 1.9 June 19, 2019*. URL: https://www.iiconsortium.org/pdf/IIRA-v1.9.pdf.

[8] Lin, S-W, Murphy, B., Clauer, E., Loewen, U., Neubert, R., Bachmann, G., Pai, M., & Hankel, M. (2017). *Architecture Alignment and Interoperability: An Industrial Internet Consortium and Plattform Industrie 4.0 Joint Whitepaper* [White paper]. Industrial Internet Consortium. URL: https://www.iiconsortium.org/pdf/JTG2_Whitepaper_final_20171205.pdf

[9] Industrial Value Chain Initiative. (2016). *Industrial Value Chain Reference Architecture (IVRA)*. URL: https://docs.iv-i.org/doc_161208_Industrial_Value_Chain_Reference_Architecture.pdf

[10] IBM (2021). Industry 4.0 architecture for manufacturing. IBM Cloud Architecture Center. URL: https://www.ibm.com/cloud/architecture/files/industry-40-architecture-pdf-template.pdf

[11] Kassner, L., Gröger, C., Königsberger, J., Hoos, E., Kiefer, C., Weber, C., Silcher, S., & Mitschang, B. (2016). The Stuttgart IT architecture for manufacturing. In *International Conference on Enterprise Information Systems* (pp. 53-80). Springer, Cham.

[12] Resman, M., Pipan, M., Šimic, M., & Herakovič, N. (2019). A new architecture model for smart manufacturing: A performance analysis and comparison with the RAMI 4.0 reference model. *Advances in Production Engineering & Management, 14*(2), 153-165.

[13] Dainow, B., & Brey, P. (2021). *Ethics By Design and Ethics of Use Approaches for Artificial Intelligence*. Retrieved from https://ec.europa.eu/info/funding-tenders/opportunities/docs/2021-2027/horizon/guidance/ethics-by-design-and-ethics-of-use-approaches-for-artificial-intelligence_he_en.pdf

[14] Hashmi, M., Casanovas, P., & de Koker, L. (2018). Legal Compliance Through Design: Preliminary Results. In *Proceedings of the 2nd Workshop on Technologies for Regulatory Compliance* (pp. 59–72).

[15] Javaid, M., Haleem, A., Singh, R. P., Khan, S., & Suman, R. (2021). Blockchain technology applications for Industry 4.0: A literature-based review. *Blockchain: Research and Applications*, 100027.

[16] Park, H. S., & Febriani, R. A. (2019). Modelling a Platform for Smart Manufacturing System. *Procedia Manufacturing, 38*, 1660-1667.

[17] Elkhawas, A. I., & Azer, M. A. (2018, December). Security perspective in rami 4.0. In *2018 13th International Conference on Computer Engineering and Systems (ICCES)* (pp. 151-156). IEEE.

[18] Hossain, M. T., Badsha, S., & Shen, H. (2020, September). PoRCH: A novel consensus mechanism for blockchain-enabled future SCADA systems in smart grids and industry 4.0. In *2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)* (pp. 1-7). IEEE.

[19] Steindl, G., Stagl, M., Kasper, L., Kastner, W., & Hofmann, R. (2020). Generic digital twin architecture for industrial energy systems. *Applied Sciences, 10*(24), 8903.

[20] Jeffay, K. (1993). The real-time producer/consumer paradigm: A paradigm for the construction of efficient, predictable real-time systems. In *Proceedings of the 1993 ACM/SIGAPP symposium on Applied computing: states of the art and practice* (pp. 796-804).

[21] IEEE. (2000). IEEE Recommended Practice for Architectural Description for Software-Intensive Systems (1471-2000). IEEE. URL: https://ieeexplore.ieee.org/document/875998

[22] Wang, S., Wan, J., Li, D., & Zhang, C. (2016). Implementing smart factory of industrie 4.0: an outlook. *International journal of distributed sensor networks, 12*(1), 3159805.

[23] Wang, S., Wan, J., Zhang, D., Li, D., & Zhang, C. (2016). Towards smart factory for industry 4.0: a self-organized multi-agent system with big data based feedback and coordination. *Computer networks, 101*, 158-168.