HORIZON 2020

D6.1 A reliable software and hardware architecture for integrated situation awareness in land- and sea-borders

Augmented Reality Enriched Situation awareness for Border security
ARESIBO – GA 833805

Deliverable Information

Deliverable Number: D6.1
Date of Issue: 15/01/2021
Document Reference: N/A
Version Number: 1.0

Nature of Deliverable: Report

Work Package: #6

Dissemination Level of Deliverable: Public

Author(s): NKUA, ADS, MARINHA, VIASAT, VTT, CERTH, CMRE, IES, TEKEVER, HMOD, MST, ROBOTNIK, UBIMAX, ADMES

Keywords: ARESIBO Architecture; Design Principles; System modules; System interfaces; Information workflow;

Abstract: This public deliverable summarizes the sub-modules of the ARESIBO components, their interfaces and the information flows among them to set-up a reliable architecture capable of supporting the border- and coast-guards' needs.

Document History

Date	Version	Remarks
18.02.2020	0.1	**Deliverable outline and Table of Contents**
23.03.2020	0.2	**First inputs integrated**
01.05.2020	0.3	**Second set of inputs integrated**
06.06.2020	0.4	**Data architecture**
18.09.2020	0.5	**Conceptual and Logical architecture**
17.11.2020	0.6	**Revisions on inputs are integrated**
28.11.2020	0.7	**Missing inputs are integrated**
05.01.2021	0.8	**Prefinal version ready for review**
12.01.2021	0.9	**Comments addressed**
15.01.2021	1.0	**Final version**

Document Authors

Entity	Contributors
NKUA	**Vassilis Papataxiarhis**
NKUA	**Sofia Kostakonti**
NKUA	**Dimitrios Zampouras**
NKUA	**Michael Loukeris**
NKUA	**Michael Tsiroukis**
NKUA	**Kostas Kyriakos**
NKUA	**Stathes Hadjiefthymiades**
ADS	**Philippe Chrobocinski**
ADS	**Nicolas Raggueneau**
MARINHA	**Vitor Conceição**
VIASAT	**Rayco Perez Cadahia**
VIASAT	**Davor Todorovski**
VTT	**Kaj Helin**
CERTH	**Ilias Koulalis**
CERTH	**George Prountzos**
CERTH	**Savvas Apostolidis**
CERTH	**Thanasis Kapoutsis**
CMRE	**Luca Berretta**
CMRE	**Sasha Godfrey**
IES	**Massimo Cristaldi**
TEK	**António Brás Monteiro**
TEK	**Pedro Petiz**
HMOD	**Andreas Tsigopoulos**
MST	**Fernando Bittencourt**
ROB	**Marc Bosch**
UBIMAX	**Pranav Menon**
ADMES	**Antonis Mousterakis**

Table of Contents

List of Tables

List of Figures

List of Acronyms

Acronym	Meaning
AHRS	Attitude and Heading Reference System
ANDROMEDA	An Enhanced Common Information Sharing Environment for Border Command, Control and Coordination Systems
AR	Augmented Reality
ARESIBO	Augmented Reality Enriched Situation awareness for Border security
BODEGA	Proactive enhancement of human performance in border security
C2	Command and Control
CAMELOT	C2 Advanced Multi-domain Environment and Live Observation Technologies
CISE	Common Information Sharing Environment
CML	COMPASS Modelling Language
CONOPS	Concept of operations
COMPASS	Comprehensive Modelling for Advanced Systems of Systems
DDDA	Dynamic Data Driven Assimilation
DSS	Decision Support System
EUCISE-2020	EUropean testbed for the maritime Common Information Sharing Environment in the 2020 perspective
EUROSUR	European Border Surveillance System
GCS	Ground Control Station
GSM	Global System for Mobile Communications
HLA RTI	High Level Architecture Run-Time Infrastructure
HOMER	Holistic Optical Metrology for Aero-Elastic Research
IDIRA	Interoperability of data and procedures in large-scale multinational disaster response actions
IoT	Internet of Things
ISTAR	Intelligence, surveillance, target acquisition, and reconnaissance
JSON	JavaScript Object Notation
KBS	Knowledge Base service
KPI	Key Performance Indicator
MANET	Mobile Ad Hoc Network
MARISA	MARitime Integrated Surveillance Awareness
MR	Mixed Reality
OCEAN2020	Open Cooperation for European mAritime awareNess
PERSEUS	Policy-oriented marine Environmental Research for the Southern European Seas
PKI	Public Key Infrastructure
RANGER	RAdars for loNG distance maritime surveillancE and SaR operations
RAWFIE	Road-, Air-, and Water- based Future Internet Experimentation
RHIB	Rubber Hull inflatable boat
ROBORDER	Autonomous swarm of heterogeneous robots for border surveillance
RTP	Real-time Transport Protocol
SoS	System of Systems
SoTa	State of the Art
TENSOR	Retrieval and Analysis of Heterogeneous Online Content for Terrorist Activity Recognition
TLS	Transport Layer Security
UAS	Unmanned Aerial System

UAV	Unmanned Aerial Vehicles
UCS	Unmanned Control Segment
UDP	User Datagram Protocol
UGV	Unmanned Ground Vehicles
UML	Unified Modelling Language
USV	Unmanned Sea-surface Vehicle
UUV	Unmanned Underwater Vehicle
VisRec	Visual Recognition
VR	Virtual Reality
W/B	White Balance
WEKIT	Wearable Experience for Knowledge Intensive Training

1 Executive summary

ARESIBO comprises a highly interconnected system of various and independent modules that feed each other with the acquired information/data concerning surveillance and monitoring tasks in border territories. The proposed solution involves three main pillars of processing: (i) a complete configuration at tactical and execution level to optimise the synergies between humans and sensors, (ii) multiple modules for enhancing the understanding and filtering of the acquired data and (iii) C2 level capabilities for enhanced event reporting. In this document, we summarise the overall architecture of ARESIBO along with the different components, their features and the basic interconnections.

2 Introduction

This document defines the high-level architecture of the ARESIBO system and the basic design principles. The rest of the document is organised as follows:

Section 3 provides a brief overview of relevant projects and systems focusing on border security as well as considering the three technological pillars of ARESIBO (i.e., Augmented Reality, Augmented Intelligence, Augmented Communication and Sensing). It also intends to position the project within the overall frame of EU-funded projects considering relevant tools and features offered by other EU-funded projects. As a complementary note, in Annex A, the reader may also find for each system component a quick summary of the starting points considering the development activities that are taking place in the project.

Section 4 provides the reader with different views of the system design focusing on the logical, data and physical views/perspectives of the architecture. The basic design principles are analysed considering (i) the user requirements as they are described in project Deliverable D2.1 "User requirements and Cognitive issues", and (ii) the mapping between the user requirements and the technical specifications as it is reflected in the project Deliverable D1.8 "Requirements Traceability Matrix V2".

Section 5 gives a more detailed description of each one of the modules that comprise the system highlighting for each module aspects such as a brief technical description, input/output, correlation with other components and responsibilities among partners. We categorise the several modules across the three technology pillars of the project and, whenever suitable, we also provide UML diagrams to emphasize on certain aspects (e.g., package diagrams are used to indicate interconnections between sub-modules, sequence diagrams are used to show flow of information considering a specific scenario/use-case etc.).

Section 6 provides a preliminary version of the message topics that have been designed to support the upcoming developments.

Finally, Section 7 concludes the deliverable summarising the major outcomes achieved and presented in the document.

3 Positioning the ARESIBO project

The goal of this section is to position ARESIBO compared to existing initiatives and highlight its unique features. To achieve this objective, we provide first an overview of the basic capabilities of the system. The we briefly describe the most relevant initiatives and systems focusing on the domain of border security as well as the ARESIBO technology pillars. As a next step, we identify the unique features of ARESIBO compared to the mentioned initiatives and we position the system within the border security ecosystem.

3.1 Overview of the ARESIBO basic capabilities

Several recent initiatives and legal texts foster an increased collaboration between Border/Coast Guards and other agencies and organisations that own and create data pertinent to the border surveillance and control activities. The flow of data made available to the border guards' organizations increases significantly with a wide diversification of the types of information. To manage this important increase of data without having to increase drastically the staff needed to process it, new system capabilities have to be developed on top of the legacy systems. These new capabilities should enhance the situation awareness (better understanding of past and present events and a better assessment of the risks) while they make the best use of Augmented Reality (AR) technologies to reduce the workload of the operators. The latter is achieved by displaying the useful information optimally and by offering real time interaction between the teams in the field and the tactical Command and Control (C2) centres (respectively Coastal Stations for Maritime borders) for the mission preparation and optimization. These new capabilities need to be supported by versatile and trusted wireless communication solutions to warrant a permanent access from the teams in the field to the private "cloud" that host the knowledge. The system can thus enable telebriefing and mission updates without the need to come back to the C2 and transfer the data from team to team.

ARESIBO aims to cover the aforementioned needs by developing and customising a set of state of the art border surveillance/control system tools and functionalities for land and maritime borders, integrating and leveraging the results from previous/on-going projects including: intuitive non-disruptive Augmented Reality tools for improved user perception for different types of actors at tactical and operational level; Hybrid Communication Network for remote areas support; a private cloud with Augmented Intelligence capabilities; sensor fusion and risk analysis tools, UxV mission planning, swarming of drones and sensing optimisation, integration of diverse Surveillance Platforms; Interoperability Layer for data and interfaces; Decision-support tools; Simulation tools and a Serious Game for user training; a semantic layer for automated reasoning. ARESIBO considers the use of all types of unmanned systems, namely Unmanned Ground Vehicles (UGV), Unmanned Aerial Vehicles (UAVs), Unmanned Underwater Vehicles (UUVs), and Unmanned Surface Vehicles (USVs) that can meet the operational requirements of different border security missions and have the potential to lead and autonomously accomplish difficult and dangerous operations, limiting man losses and operational costs significantly.

The tools considered in the project are built around its three technological pillars:

Augmented Communication and Sensing. The first pillar of ARESIBO aims to create a baseline reflecting what the present (now and short term) situation could be (regardless of financial constraints) and implement a first version of the technologies that will be used to enhance situation awareness. This pillar considers a set of technologies and tools that will be developed for unmanned platforms, sensors and sensing optimisation based on active sensing, visual recognition, hybrid intelligence networks, voice/video communication and cyber-security.

Augmented Intelligence. The second pillar of ARESIBO, at mid-term horizon, is to improve situation awareness by enhancing the understanding of the situation through adapted processing of sensor data, correlation between heterogeneous data and information and creation of knowledge through machine learning techniques and forecasting for risk analysis, decision-support for optimised use of resources (e.g., unmanned platforms), a simulation engine for user training, and mission planning tools.

Augmented Reality. The third pillar of the project targets to implement an integrated Augmented Reality (AR) capability around the private “cloud” for both tactical C2 centres and field units. In this context, ARESIBO investigates the use of cutting edge AR devices (both monocular and binocular) for all the three types of actors considered in the project as they are described in D2.1 “User requirements and Cognitive issues” (i.e., C2 operator, Tactical Commander, Field Officer). Among other things, this pillar will develop AR-based cross-layer communication for typical activities in border security operations (e.g., ordering, reporting, etc.) as well as a Serious Game for user training.

Figure 1 summarises the different ARESIBO capabilities per technological pillar.

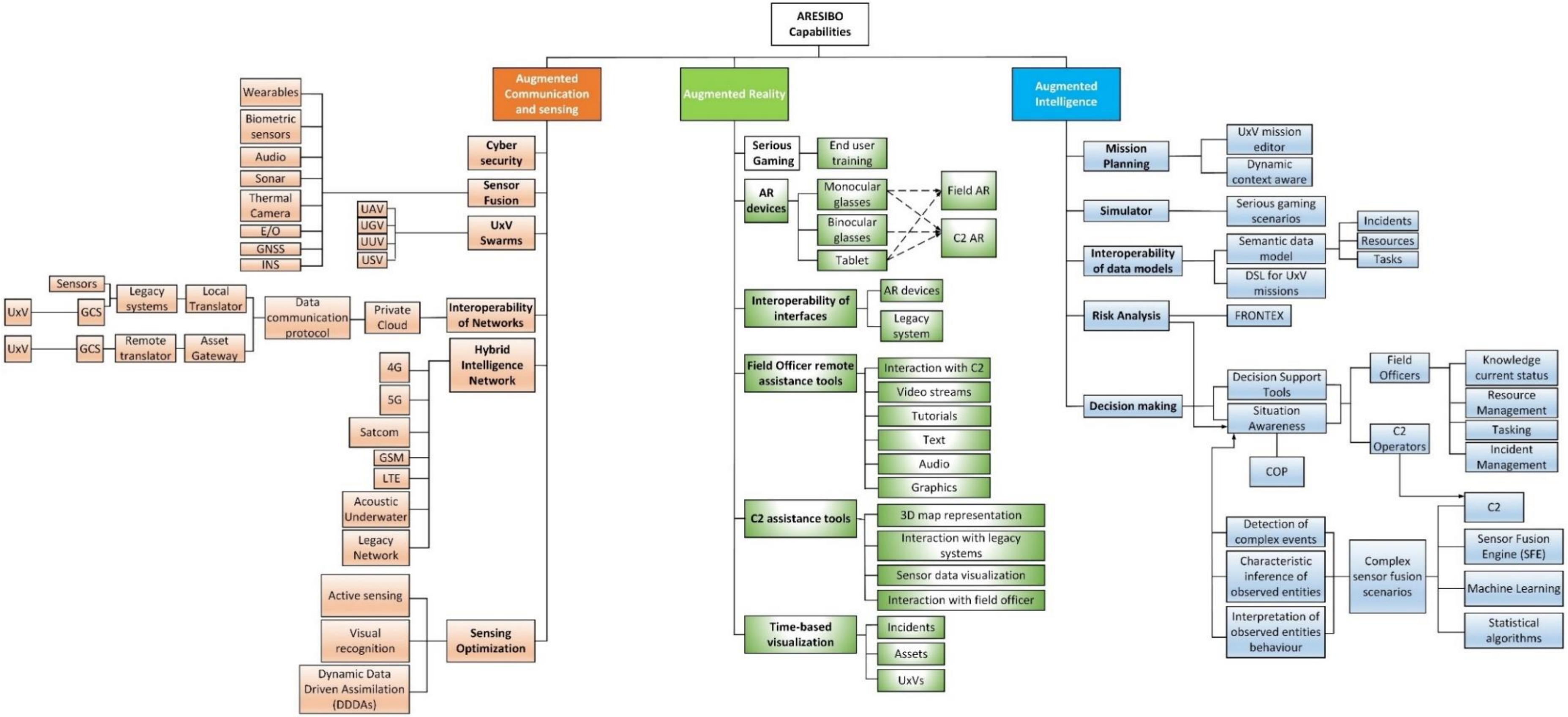

Figure 1: ARESIBO basic capabilities

3.2 Initiatives relevant to ARESIBO

In this section, a brief overview of projects and systems that are relevant to ARESIBO is given. Most of these architectures are funded under the Horizon 2020 (H2020), the 7th Framework Programme for Research (FP7) and the Preparatory Action for Defence Research work programmes.

Road-, Air-, and Water- based Future Internet Experimentation) (RAWFIE). This project produces and implements an experimentation platform, that integrates numerous test beds for Internet of Things or IoT devices. The platform enables experimenters to monitor communications, navigate unmanned vehicles in all the environments (ground, maritime and aerial) via satellite and data collection from devices. A specific message bus is implemented to connect the various components such as a frontend interface to program, plan missions and monitor experiments with components that are controlling the validity of many aspects of the missions that are planned (the testbed selected, data analysis, experiment validation and launching services). Connected via the message bus are also the testbed monitors and managers as well as each UxV's information status and onboard sensor data collection. The middleware developed by NKUA in RAWFIE could be the starting point for the integration environment that will be used in ARESIBO. Also, the mission planning tools of RAWFIE that were developed for the representation of robotic experiments can be abstracted in a way to model UxV missions for border security. Also, parts of RAWFIE swarm operation systems for UxVs correspond to the collective intelligence for swarming robots and optimized human-robot collaboration for ARESIBO.

Comprehensive Modelling for Advanced Systems of Systems (COMPASS). This project focuses on big challenging architectures in the form of System of Systems (SoS) and the development of tools and frameworks that will allow the creation, analysis and maintenance of such systems. The project extends a System Modelling Language and offers a formal notation that describes specific SoS aspects, named COMPASS Modelling Language (CML). The projects outcome acts as a link between underlying semantics, such as transfer protocols etc., and a System Modelling Language with the added CML extensions and components to describe a complex SoS architecture. It is possible to find some connection to the future developments of the ARESIBO project with respect to the unmanned platforms that may be considered as an SoS inside the architecture.

MARitime Integrated Surveillance Awareness (MARISA). This project offers a toolkit developed to help EU and National Authorities to achieve maritime safety, environmental protection, fishery control and maritime border protection of oceans, seas and coasts. Raw data is collected from various and heterogeneous sources to improve the maritime situational awareness, support staff and trainees at sea. Also, collaboration across different border agencies is intended by creating a digital dynamic eco-system of users and technology providers. Focusing on the integration of data that stem from various data sources and the System of Systems concept in mind, the MARISA project architecture consists of 3 operation layers: the observation layer for the data collection and monitoring; the analysis layer, for data analysis; and the decision-making layer that offers possible decision outcomes, predictions and risk assessments. Possible connection between the developed toolkit in the MARISA project and ARESIBO could be the concept of operations between the two projects and the definition metrics and KPIS described in ARESIBO with the evaluation metrics used in MARISA.

Policy-oriented marine Environmental Research for the Southern European Seas (PERSEUS). PERSEUS is a project Funded by the EU under FP7 Theme "Oceans of Tomorrow". It aims to offer better handling of policies mainly regarding the Southern Europe seas. Tools are developed for different policymakers to monitor various phenomena from activities that lead to the degradation of the marine ecosystem. Human activities, such as over-fishing and sea pollution are monitored as well as the real extent of changes to marine life from climate changes and other environmental causes. The toolset designed and developed for this project, enables scientists to obtain findings that lead to accurate knowledge of marine disasters. The toolset consists of the merge of sciences for comprehensive predictions of long-term effects on societies, a scenario based Adaptive Policy Framework, a tool for visualizing marine life thread models, the construction of specific designed vessels and the creation of the Citizen Scientist program to encourage people to contribute to scientific research.

EUropean testbed for the maritime Common Information Sharing Environment in the 2020 perspective (EUCISE-2020). EUCISE 2020 is a Security Research project of the European Seventh Framework Program under the FP-SEC projects and offers an environment for Information Sharing purposes between various maritime authorities in the European States. The design of the information flows can be described as a centralized monitoring tool that monitors various activities at sea, e.g. fishing activities and other surveillance data. The collected data can then be transmitted to various endpoints, like border control agencies or customs agents, etc. Specific protocols have been designed for managing large information sets for achieving information-centric decision making. The CISE data model is one of the inputs considered for the specification of the ARESIBO data architecture.

Autonomous swarm of heterogeneous robots for border surveillance (ROBORDER). The ROBORDER project [1] has the goal to develop a border surveillance system that will be functionally autonomous with use of unmanned robots. Sensors mounted on surface, eater, underwater, aerial and ground vehicles feed the system with real time data. Data Fusion algorithms as well as prediction mechanisms and visual feeds gathered from the unmanned vehicles will assist border authorities to monitor illegal border and coastal activities and marine pollution events and disasters. The system gathers data from different data sources and handles heterogeneous data streams to assist authorities and border officers to take the optimum course of action. The architecture of the project consists of three layers: the operations layer, the autonomous resource task coordinator and the layer regarding the commanding of the robotic missions. Maritime and land operations are controlled and monitored from control rooms with use of radar networks. The controls center then coordinates staff and authorities to resolve any border security issues. Possible connections between the work accomplished in ROBORDER and the ARESIBO future developments include the Augmented Reality tools for the remote control of UxVs by the C2 operators and the simulation environment. From a data perspective, parts of ROBORDER data models correspond to the representation of UxV resources providing a starting point for the ARESIBO data model as it was presented in D4.1.

Wearable Experience for Knowledge Intensive Training (WEKIT) is a Horizon 2020 research project that aims to develop and test the use of smart Wearable Technology in industrial training. With use of the Augmented Reality technologies learning content and technical documentation are projected to the trainee making the training procedures more efficient. The main use cases are the training of aircraft maintenance personnel, bioimaging and astronaut training. One of the devices or wearables is the Microsoft HoloLens device which is head mounted. Many functionalities such as the design of the projected information or the interaction with the devices is studied and developed. Concepts and best practices followed for the AR-visualization System and AR-based Experience Re-enactment and Learning System may prove useful to ARESIBO, and in particular, the AR interfaces and tools for C2 C2 as well as the Serious Games for user training.

Proactive enhancement of human performance in border security (BODEGA). The project's goal was to create a system that will specify the traveller processes, the procedures, the border controller work with use of smart border control systems that make use of biometric features. It automates gates surveillance, control and self-service systems to ensure safe travel for travellers and border security. Evaluation of already existing systems has been studied and used to propose new decision support systems and their design as well as Innovative gaming and e-learning solutions for border control and security.

Extending scope of CISE to meet surveillance needs at the border (ANDROMEDA). ANDROMEDA is a Horizon 2020 project that aims to extend CISE for land operations. Aim of the project is to extend the capabilities of the CISE Maritime model to land scenarios. A specific Land Surveillance Information Exchange system has been created for this purpose along with the implementation of Command and Control systems, data fusion functionalities and Decision Support mechanisms. Along with reports and Legal, Ethical and Societal Aspects of the project and the usage of the developed technology a new e-CISE Data Model description has been described to manage the data collection and usage parallel to the CISE architecture and data model. The design of the project consists of many components and functionalities. Main aspects and components are the integrated C2 capabilities, data fusion and analytics services, and the connection of various human and machine interfaces with developed CISE adaptors. Possible connection between the work accomplished in the ANDROMEDA project and the ARESIBO developments may include the data model developed in the scope of WP4.

RAdars for loNG distance maritime surveillancE and SaR operations (RANGER). RANGER is a surveillance platform that makes use of existing radar technology and combines it with features and platforms to achieve early warning and tracking of sea vessels. With use of data collected from other platforms like EUROSUR and with use of existing frameworks like CISE, an early detection system is built. A radar system has been also created along with the needed interoperability and communication features that are needed. Specific use cases and scenarios have been created to assess and test the platform. Many of the used or studied platforms for this project are also used for border and coast protection. Border security scenarios and pilot use cases found in RANGER may be connected to the development of demonstration scenarios based on CONOPS and the definition of metrics and KPIs in ARESIBO.

Open Cooperation for European mAritime awareNess (OCEAN2020). It is a European Defense Agency project with the goal of enhancing the situational awareness in maritime environments. This is achieved with the integration of legacy and new technologies and the use of unmanned vehicles aiming to combine various existing EU surveillance systems and organizations that offer various technology solutions and frameworks. To achieve this communication and data models are created and designed, integration of UxSs, data fusion from sensors and training tools for enhanced results and operability. CONOPS developed in OCEAN2020 can provide valuable input for the ARESIBO CONOPS and especially for the maritime operations as well as the sea surveillance scenarios.

Holistic Optical Metrology for Aero-Elastic Research (HOMER). This is a project that aims to implement tools to better understand the flow features of components of aircrafts. Due to the great difficulty to accurately understand and measure the physics in various aircraft components, HOMER proposes a specific metrology to contribute to the understanding of flow features and load balancing affecting the aerodynamics of aircrafts. The proposed metrology also implements optical tools to visualize the various flows and aerodynamic behavior in air-vehicle designs. Therefore, it is possible to relate and connect features of implemented visual recognition tools used to recognize a structure with the sensing optimization for visual object recognition.

C2 Advanced Multi-domain Environment and Live Observation Technologies (CAMELOT). This project aims to create and implement a command and control standard to different border security authorities to add new technologies and features to already used command centers. For this to be possible, a new standard is proposed and described that will enable the integration of new assets and tools to existing command and control architectures. Tools and frameworks are implemented to enable the control of new components in C2 centers like unmanned vehicles and surveillance systems that will fit to existing interfaces and architectures. The components to be delivered are divided into blocks and consist of mission planning modules, automatic tasking and control, data visualization of service modules, sensing and detection, data management and analysis and communication and networking. Possible connections between the two projects are the data model, the network architecture, the data communication protocols for field operations and the cybersecurity mechanisms.

Retrieval and Analysis of Heterogeneous Online Content for Terrorist Activity Recognition (TENSOR). TENSOR is a project with the goal to ensure the safety of people against terrorist acts and recruitment. It is intended to be used by law enforcement agencies across the EU offering functionalities for early detection and prevention of terrorist acts by filtering internet content possibly published by terrorist groups or individuals. Many functionalities have been designed in order to filter, search, crawl and gather online content.

Interoperability of data and procedures in large-scale multinational disaster response actions (IDIRA). In this project the main goal was to design and implement a framework that would allow emergency agencies to integrate their capabilities. To achieve this, a mobile integrated command and control structure has been designed and developed. It is a service integration project adopting a service-oriented architecture. The project focused on civil protection agencies and was based on a number of service components including sensor fusion, data interoperability and interoperability of network protocols.

Figure 2 presents the abovementioned initiatives and potential connection points with the ARESIBO developments.

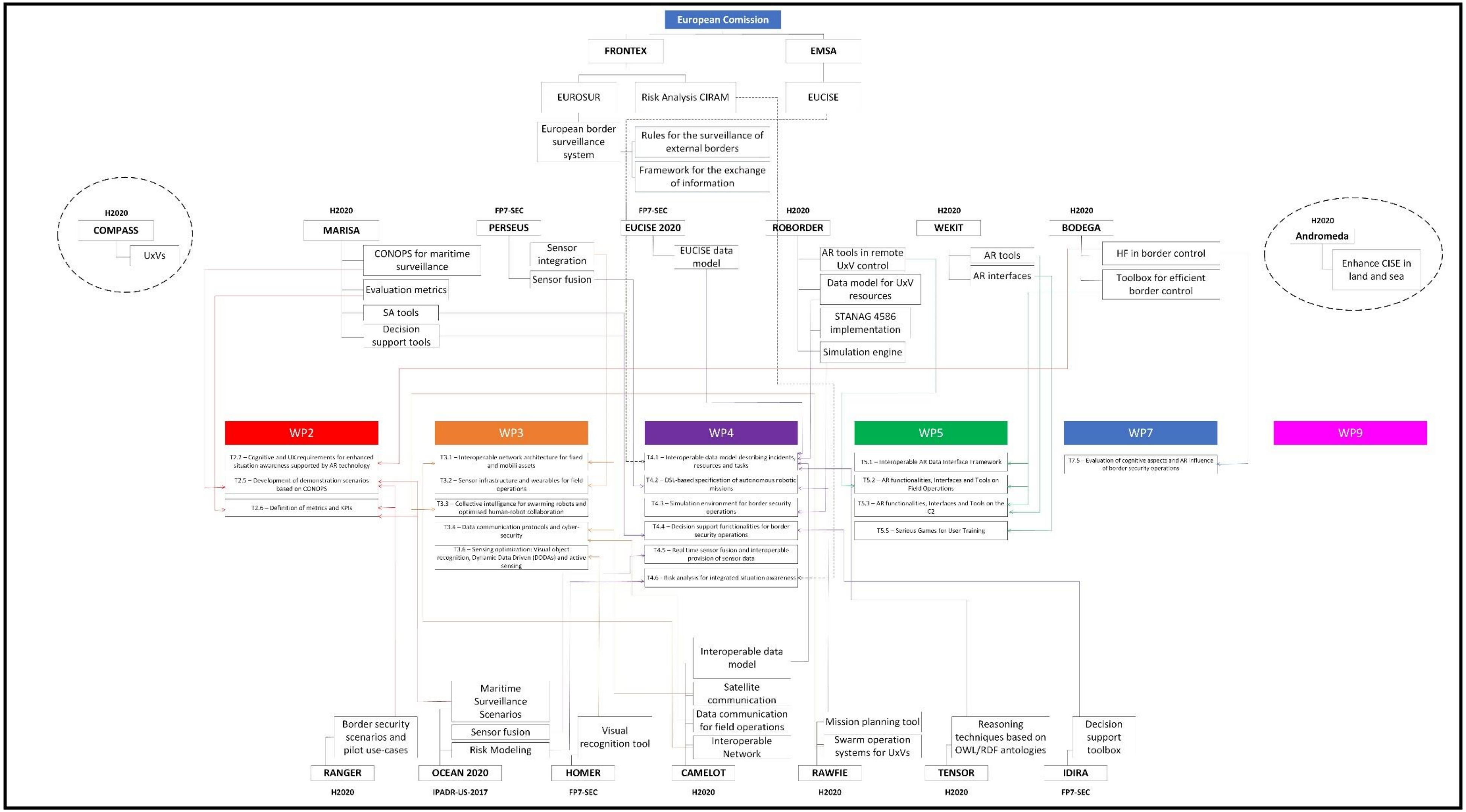

Figure 2: Relevant projects and connection with ARESIBO

3.3 Positioning of the ARESIBO project

Compared to the aforementioned initiatives, ARESIBO goes beyond the creation of yet another IT architecture to support border security operations by making the platform flexible enough to connect with and complement existing legacy systems and by developing tools interoperable in terms of data exchange. ARESIBO does not aim to build a new C2 environment from scratch; instead the project targets to build a set of tools that will seamlessly interact and will supplement existing C2 environments and legacy systems by both consuming/providing data.

In particular, the project differs from existing efforts in the following means:

- ARESIBO builds its tools having the Augmented Reality technologies always in the heart of its developments. It innovates by thoroughly investigating the use of such a technology such as AR for border surveillance in both tactical and operational levels. Also, three different types of actors are considered (i.e., Field Officer, Tactical Commander, C2 operator), each one with different needs, priorities, challenges to be tackled and limitations. For instance, a C2 operator has permanent access to resources such as power supply and network. On the other hand, a Field Officer must deal with little energy resources and limited connectivity. On the other hand, a C2 operator that supervises the operations, needs to have access to tools with capability to visualise all the different types of information upon a map, while the Field Officer that operates in the field needs access only to data that will be really meaningful during a field operation (such as alerts). ARESIBO builds AR tools for all the three types of considered actors.
- The project will integrate, test and evaluate the use of different types of AR hardware (monocular AR, binocular AR, AR tablets) in the different types and different levels of operations and actors. Obviously, people who are field actors cannot carry a heavy headset for a long time; they need a lightweight headset that can be easily carried. On the other hand, for a C2 operator it might be critical to wear a binocular device that offers optimized visualization of data in order to have a better and more accurate understanding of the information.
- From a data model perspective, ARESIBO decided to adopt UCS 3.4 architecture as a suggestion made from the end users of the platform. ARESIBO is adopting and implementing an extended version of UCS3.4 data model. The project follows the exact structure of messages described by the UCS3.4 standard and, whenever needed, it extends the messages with new fields (e.g., for risk alerts, a separate field is used to describe the CIRAM details about threat/impact/vulnerability levels).

- Combination of both real-time detections (visual recognition, sensor fusion, reasoning) and future predictions (risk analysis). Especially, the ARESIBO sensor fusion tool allows for the easy creation and deployment of fusion scenarios that
- Optimised swarming of heterogeneous robots. The project supports the orchestration and optimization of combined UxV missions composed by different types of unmanned platforms. Apart from the typical UxV missions based on the determination of the waypoints the assets should follow, ARESIBO also deals with a second type of missions where the operator defines only a region of interest and a group of unmanned vehicles that will be responsible to optimally cover it based on their characteristics.
- A direct connection between the simulation engine and the serious game is a novelty for the project since not only it allows the user to specify a set of training scenarios for border security operations, but also to test them in a realistic environment as the one offered by the serious game.
- The adoption of CIRAM terminology for the representation and labelling of tactical and operational risks.
- ARESIBO considers scenarios for three different types of environments (land, maritime, and mixed environments) and for three different types of actors in the tactical and operational layers (C2 Operator, Tactical Commander, Field Officer).
- System deployment and testing. The platform is planned to be deployed, tested and validated for long periods in real-world environments in two countries (Finland, Greece).
- KPIs extensible approach. ARESIBO specifies a generic approach for the establishment of Key Performance Indicators (KPIs) that is broken down to a quantitative and qualitative analysis of metrics. Also, specific KPIs are specified for Augmented Reality User Experience and Cognitive Capability.

The above characteristics of the project differentiate the project from existing initiatives and give ARESIBO a unique character within the border security ecosystem by increasing its potential and the expectations for what concerns the project outcomes. The next sections of this document intend to describe the design principles of the ARESIBO architecture and a description of all the project components.

4 High-Level ARESIBO Architecture

This section presents the different views of the ARESIBO architecture before we go into details for each component separately in the next sections. First, we give a conceptual and logical view of the system. Next, we describe the ARESIBO architecture from a data perspective. Finally, the overall physical topology of the system is discussed.

4.1 Conceptual and Logical Architecture

This section briefly describes the logical modules of the ARESIBO system. From a conceptual point of view, ARESIBO can be considered as a stack consisting of the following layers (bottom-up) as it is depicted in Figure 3:

ARESIBO Hardware. In this bottom layer of the ARESIBO stack, we have all the different hardware components of ARESIBO. Among other things, this layer includes the different Augmented Reality devices of the project (i.e., monocular & binocular glasses, tablets), the unmanned platforms we consider as part of the project (i.e., UAV, UGV, UUV, USV), the sensors and any kind of communication infrastructure, the cameras (conventional, thermal, Infrared) and any other potential type of equipment that is part of the system.

Interoperability layer. This layer refers to any data standards that are adopted by the system in order to facilitate its connection with external systems (e.g., legacy systems, C2). Indicatively, in ARESIBO, we implement an extended version of UCS3.4 architecture for the management of UxV and we support a series of standards in the different modules of the system (e.g., HLA for the simulator, CIRAM terminology for modelling operational and tactical risks, OGC standards for sensors and CISE data model).

Storage layer. This refers to the persistency layer of ARESIBO. The project adopts a combination of conventional storage architectures (e.g., RDBMS for data that do not change frequently) and message-oriented middleware (e.g., Kafka message queues and middleware) to facilitate data exchange and storage. The data may refer to any kind of operational data (e.g., ordering and tasking, management of resources, etc.), sensor data, alerts and warnings produced by the system, video data, or even semantic data for the ARESIBO ontology network.

Software. This layer refers to the software services and the applications that are implemented in the context of the project. All the software modules of ARESIBO should participate in this layer.

Application layer. The upper layer of the logical architecture refers to the set of application scenarios that the system will be able to support. In our case, the Project Use-Cases that will be used to validate system performance are definitely the starting point for this layer.

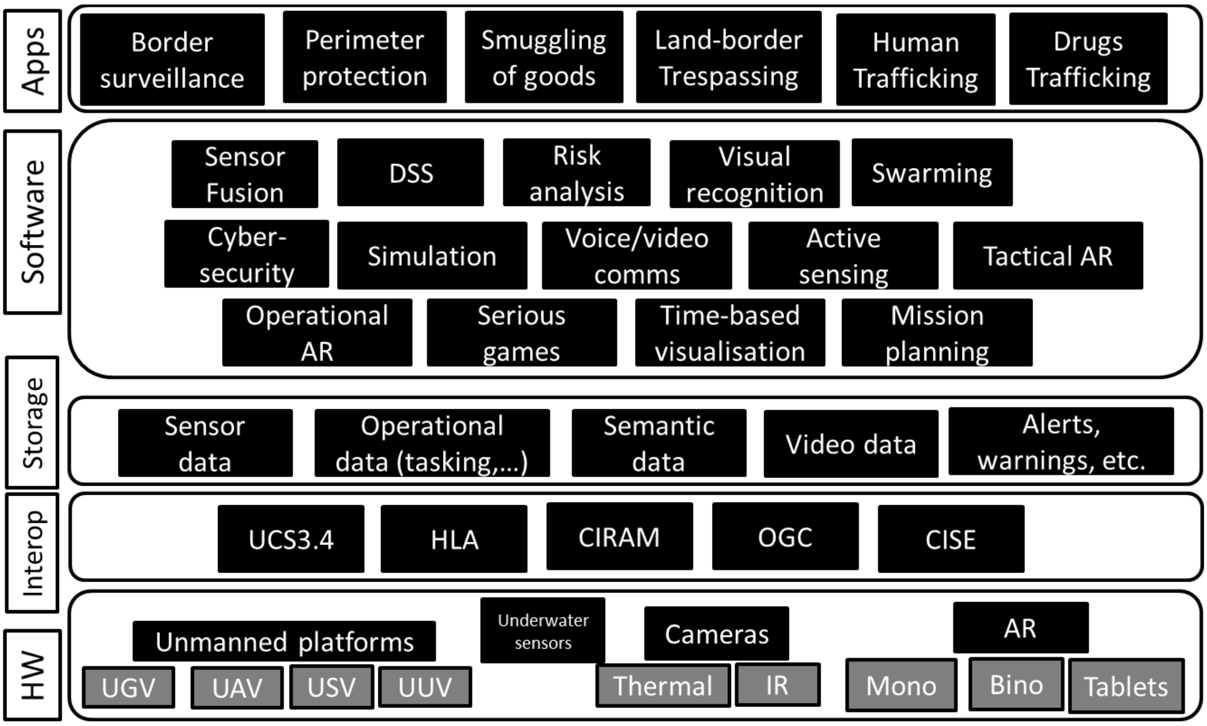

Figure 3: Logical components of ARESIBO

Considering the outcomes presented in Deliverables D2.1 "User requirements and cognitive issues V1" and D2.8 "CONOPS analysis, demonstration scenarios and evaluation metrics V1", as a next step, we discuss the high-level system architecture of ARESIBO. In Figure 4, we present the different components of the system, their classification considering the technology pillars, their interconnections, their correlation with both ARESIBO actors and external systems, and a preliminary assessment of responsibilities between the project partners for leading the implementation activities of each module.

Following a top-down approach, the following parts of the architecture are met:

The ARESIBO Actors (yellow box). ARESIBO considers three types of actors as they were described in WP2 activities and outcomes:

- *C2 Operators* that may operate the different tools within the C2
- *Tactical Commanders* that are the on-scene commanders when an incident takes place.
- *Field Officers* that constitute the patrol units and the coastguards that act in the field.

The three technology pillars of ARESIBO and the technical components of each pillar:

Augmented Reality pillar (cyan box). This pillar builds AR tools for all the three types of Actors considered by the system. The following components belong to the Augmented Reality pillar:

- *AR support for Field Officer* (UBI). The AR applications that will be built for the Field Officers.
- *AR Support for Tactical Commander – Binocular device* (VTT). The AR applications that will be built for the Tactical Commanders on MS HoloLens 2 device.
- *AR Support for Tactical Commander – Tablet* (VTT). The AR applications that will be built for the Tactical Commanders on Tablets.
- *AR Support for C2 Operator* (ADS). The AR applications that will be built for the C2 Operators on MS HoloLens 2 device.
- *Time-based visualization* (VTT). The time-based projection component for MS HoloLens 2 device.
- *Serious Game* (CMRE). A Serious Game for user training.
- *Data communication gateway for AR* (UBI). A data connector for the AR devices in ARESIBO.

Augmented Intelligence pillar (green box). This pillar develops services and tools for the ARESIBO private cloud. The following components belong to the Augmented Intelligence pillar:

- *UxV Mission Editor* (NKUA). A tool with a GUI for designing missions consisting of autonomous vehicles of all types.
- *Reasoning service* (CERTH - MKLab). Back-end service that combines existing data in order to deduce new and more accurate knowledge. It is based on the ARESIBO semantic models.
- *Decision Support Tool* (IES). A tool for providing recommendations on the optimized use of resources that will facilitate decision making.
- *Simulator* (CMRE). A simulation engine for user training. It will feed the ARESIBO Serious Game with data.
- *Sensor Fusion* (NKUA). It provides real-time detections based on multi-level correlation of sensor data.
- *Risk Analysis* (NKUA). It implements a series of predictive models to estimate risks (e.g., future position of an unidentified moving object). It uses the CIRAM terminology to label tactical and operational risks.

Augmented Communication and Sensing pillar (light orange box). This pillar designs and implements the ARESIBO network, the management of the UxVs and the sensing elements. The following components belong to the Augmented Communication and Sensing pillar:

- *Visual Recognition* (CERTH – MKLab). A set of visual detection algorithms based on deep learning.
- *Data communication and cyber-security* (TEKEVER). Secure data communication solutions and protocols in ARESIBO.
- *Swarm Intelligence and Human-Robot collaboration* (CERTH – ConvCAO). Swarm intelligence schemes for optimized orchestration of missions consisting of autonomous unmanned platforms.
- *Network components and Island Gateway* (VIASAT). The ARESIBO network infrastructure for land and maritime environments.
- *Active sensing* (CERTH – ConvCAO). A software solution for the optimized use of sensors (e.g., control, activation, de-activation).
- *Voice and Video communication* (VIASAT). A solution to provide network connectivity between the on-field operators and the corresponding C2 center to support voice and video applications.
- *Resource control* (CERTH – ConvCAO). A back-end optimizer of UxV mission plans.
- *UUS/USV* (MST). The unmanned underwater and sea-surface platforms of ARESIBO.
- *UAV* (TEKEVER). The unmanned aerial platforms of ARESIBO.
- *UGV* (ROBOTNIK). The unmanned ground platforms of ARESIBO.
- *UxV sensors* (MST). The sensors that will be used in ARESIBO; mostly the ones attached to the unmanned platforms.
- *Underwater sensors* (ADMES). Hydrophones and acoustic sensors for underwater surveillance.

ARESIBO Knowledge Base (orange box). This component contains the ARESIBO ontology network which is a set of interconnected semantic models. Within the ARESIBO Knowledge Base, the following semantic models have been designed:

- *ARESIBO Core ontology* (CERTH – MKLab). It is connected with other semantic models for data integration purposes. It is used by the Reasoning service of WP4.
- *CIRAM ontology* (NKUA). A semantic model representing the terminology offered by CIRAM. It is connected with the ARESIBO Core ontology and it is used by the Risk Analysis module to label operational and tactical risks.

Persistency Layer (light grey box). It is the ARESIBO Storage. First it includes a Communication Middleware based on message queues (Kafka middleware). This middleware acts as an intermediate message bus that routes the messages from the sender(s) to the recipient(s) efficiently. The message follows a JSON format following and extending, whenever this is needed, the UCS3.4 data model. In addition, we use a no-SQL database to push all the messages that pass through Kafka into a doc-oriented database.

C2 Visualisation (blue box). ARESIBO will make use of a stripped-down version of C2 environment to demonstrate interoperability between ARESIBO and external components. The basic use of C2 functionalities will be achieved through the AR C2 application.

External interoperability layer (dark grey box) – Connectors with legacy systems (red box). ARESIBO will validate its technical outcomes by checking the capability of the system to communicate with external and legacy systems. This layer provides a set of connectors for the external systems that are considered in the context of the project.

Section 5 that can be found later in this document provides more details for each one of the ARESIBO discrete components that were discussed above.

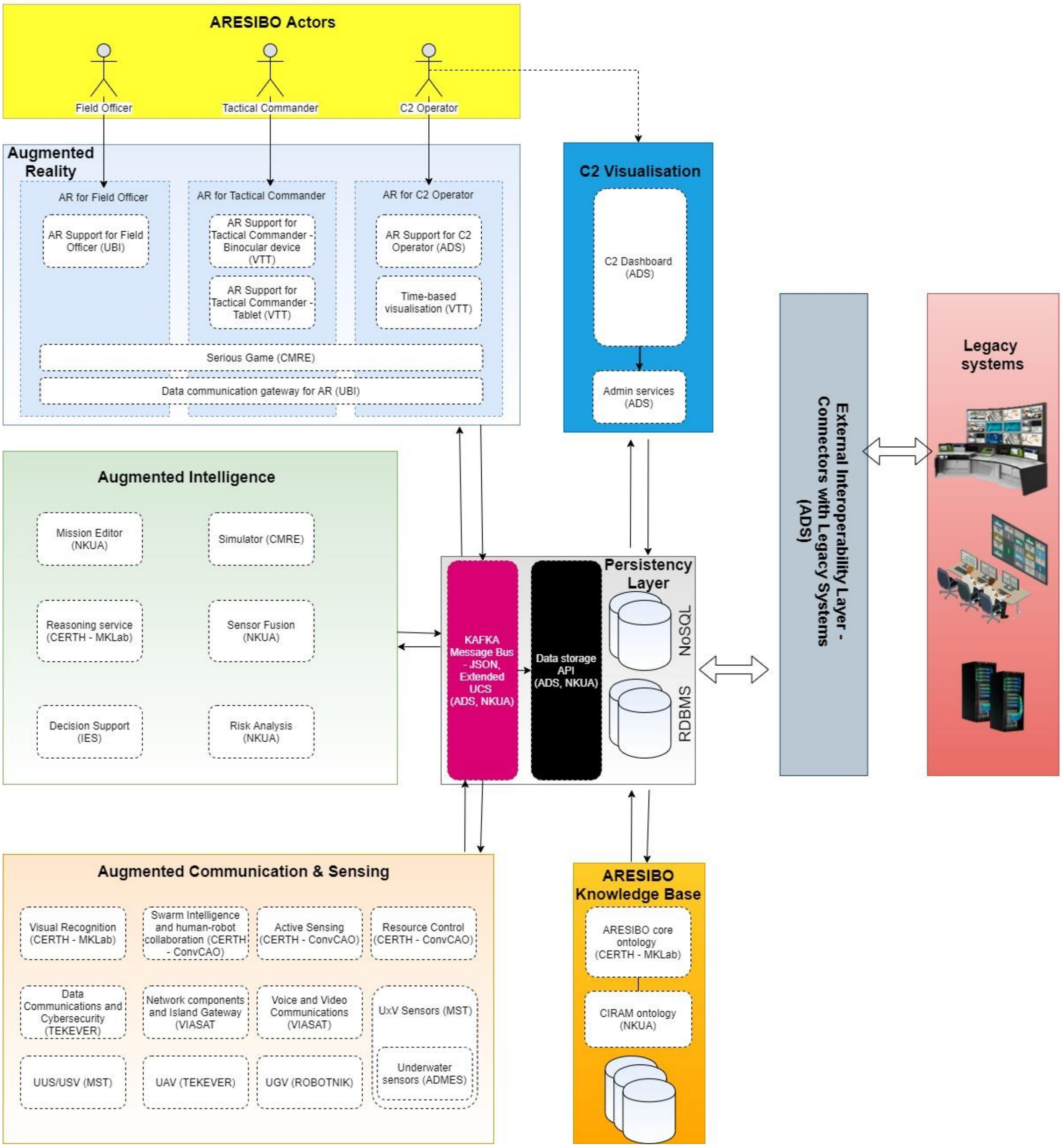

Figure 4: High-level system architecture of ARESIBO

4.2 Data Architecture

The ARESIBO system is an interconnected platform that accumulates, processes and distributes surveillance information from multiple sources to various modules with the aim of optimizing the synergies between humans and sensors, enhancing the understanding of the acquired data and providing an advanced event-reporting functionality. Achievement of these tasks requires the data-source integration framework to be developed upon one interoperability layer, which consists of a number of software modules that have to collaborate, coordinate, create and transfer knowledge for improved situational awareness. This interoperability layer relies on the development of standardized data structures which are part of the information transactions and services and comprise the ARESIBO data model ensuring multidirectional data flow effectively and efficiently.

The data architecture is built upon that data model described in D4.1, which is the core of the system's communication amongst the software components and can be shown in Figure 5.

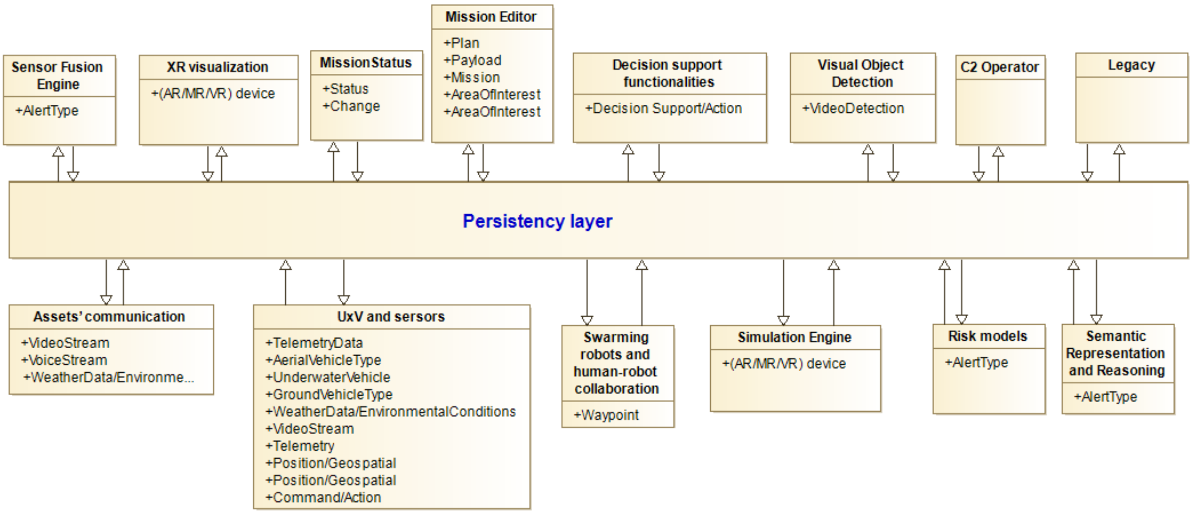

Figure 5: High-level representation of the ARESIBO data structure

Finally, Figure 6 presents a preliminary step of the project towards the identification of the exact messages and dataflows that should be supported by the system. This was the starting point the project used in the beginning of the first integration cycle. It was also the motivation for the technical partners to perform an exhaustive search on the existing standards supported by the end users in order to support those actions. Based on the outcome of this examination, we identified the UCS3.4 messages that could prove useful to ARESIBO in order to facilitate all the different types of required communication.

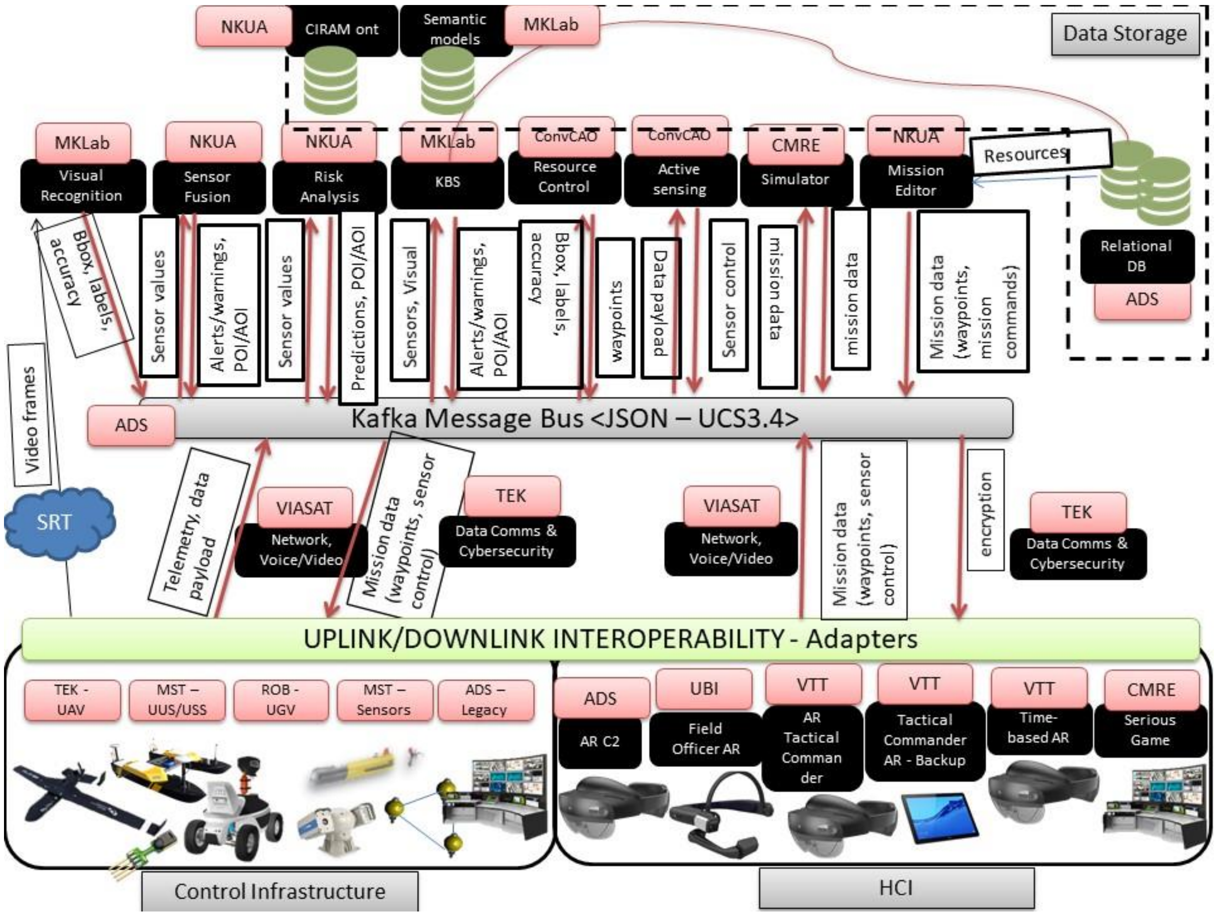

Figure 6: Basic dataflows between components

4.3 Physical Architecture

One of the main goals of ARESIBO is to establish reliable and secure connectivity between the field units, field commanders (operational level) and the C2 centers (tactical level). The overall ARESIBO communication solution is based on a secure hybrid network approach that will provide high availability in remote areas outside of the coverage area of traditional communication networks or areas where these networks have been disrupted due to a disaster. The hybrid network will provide the necessary availability and data rates within the latency constraints in order to support the large set of interactions between the field units and C2 centers in time-critical missions.

4.3.1 Network architecture description

This section presents a detailed description of the ARESIBO network architecture with Figure 7 presenting a high-level overview of the three main segments:

- **Radio network** presents the last-mile network that connects the field units and UxVs to the communication hub.

- **Viasat Land Vehicle** that acts as the backhaul to the radio network, i.e. the communication node which connects the radio network to the C2 center. The Viasat Land Vehicle has three main functionalities:
    - **Communication hub** - The main goal of this nomadic communication hub is to provide network coverage to areas where the ordinary land communication systems are not available due to lack of coverage, unreliability or disruption.
    - **On-board camera** – The vehicle is mounted with an IP camera that streams a real time video and can be controlled both by an operator inside the vehicle and the C2 center.
    - **Ground Control Station** – The vehicle can host an operator of the camera, as well as an officer who has situational awareness and can coordinate and provide support for the on-field units.
- **Command and Control center** is the unit that has a global view and control of all the components of the ARESIBO project.

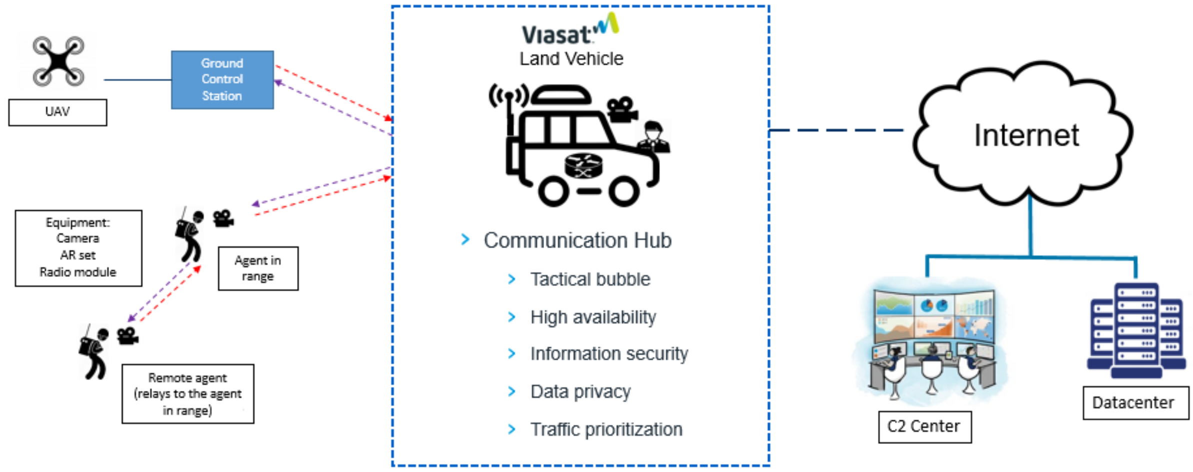

Figure 7: ARESIBO Network Architecture schema

In the following paragraphs the main components of the ARESIBO network architecture will be described in detail explaining indicative use-cases for land, maritime and mixed environments.

5 Components overview

This section intends to collect a separate description for each technical module of the ARESIBO system. This description, among other things, summarises also the following items for each module:

- Supported functionalities;
- Brief technical description;
- Expected inputs and outputs;
- Interconnection with other modules;
- Candidate implementation technologies;
- Tasks where the development will take place;
- Responsibilities of partners.

Whenever suitable, the description of the modules is complemented by UML diagrams to describe the sub-components of the technical module, the functional use of the module and/or its interconnections with the rest of the system.
Based on the functional use, each module is categorised into either one of the three technological pillars of the project (i.e., Augmented Communication and Sensing, Augmented Intelligence, Augmented Reality) or the integration pillar that contains cross-pillar modules.

5.1 Augmented Communication and Sensing Components

5.1.1 Unmanned Ground Vehicles

Component name	**Unmanned Ground Vehicles (UGV)**
Responsible partner	Robotnik
Provided functionalities	UGV's are robots with autonomous navigation capabilities in outdoor environments. The main objective of the UGVs is to patrol the border while carrying a set of sensors to provide live feedback to ARESIBO.
Technical Description	UGV are robots equipped with wheels and a set of sensors for autonomous navigation (GPS, Cameras, Lidars). Two

	different UGV from Robotnik are envisaged to be used in ARESIBO: RB-Car and Summit XL. RB-Car is a small-car sized (3x2x3m) vehicle capable of carrying a big payload of sensors and actuators, with total travel distance up to 40Km. Summit XL is a small vehicle (0.7x0.6x0.4m) with a limited payload, and a total travel distance up to 5Km. Each UGV communicates with a Ground Control Station (GCS) that provides monitoring and teleoperation and also acts as a bridge between the UGV and external systems.
Expected input	Missions composed of a set of waypoints that each UGV has to navigate to autonomously and commands for teleoperating the vehicle itself and its payload.
Expected output	Current pose of each UGV in GPS coordinates and live stream of sensor data.
When it is used	UGV will be used to complete the missions defined by ARESIBO.
Relation to other components	Consumes missions from the Resource Controller. Produces UGV Status (including UGV state, position, etc.) and MissionStatus for the ARESIBO Dashboard and live streams of sensor data for Detection Systems.
Candidate technologies to be used	Robotnik RB-Car, Robotnik Summit XL.
Relevant Tasks	Task 6.2 – Software and hardware integration at component-level Task 6.3 – Full-system integration, customisation and operational testing

Comments	Communication between each UGV and its GCS is crucial. Two communication systems are available: 4G connection (preferred but limited in land borders) or Long-Range Wifi (requires direct line-of-sight between UGV and GCS).

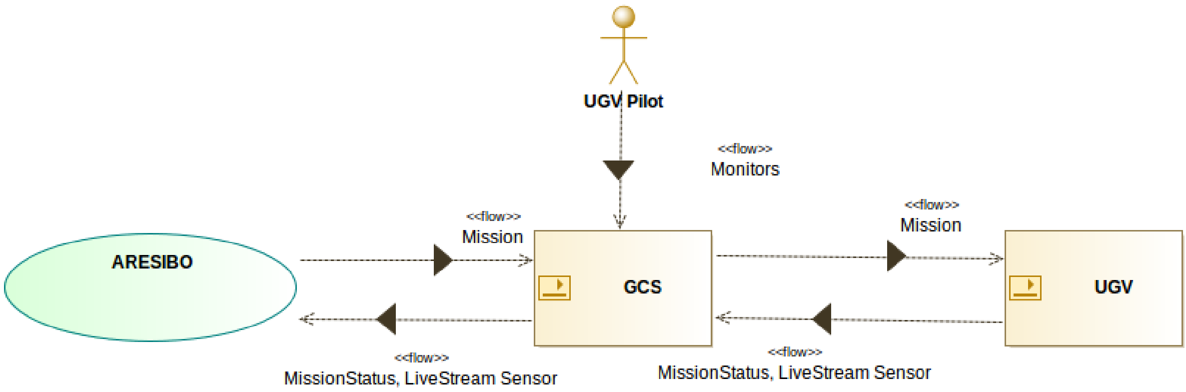

Figure 8: Package diagram for ARESIBO UGV

5.1.2 Unmanned Aerial Vehicles

Component name	**Unmanned Aerial Vehicles (UAV)**
Responsible partner	TEKEVER-ASDS
Provided functionalities	TEKEVER AR3 is a shipborne UAS designed to support multiple types of maritime and land-based missions, including ISTAR, pollution monitoring, infrastructure surveillance and communications support operations
Technical Description	The AR3 Net Ray is a small Unmanned Aerial System (UAS) with medium endurance, specially designed for maritime and coastal surveillance missions, but also supporting different types of land-based operations. Due to its communications range of up to 80 Km, as well as its reduced logistics footprint (short set up time and easily installable launch and recovery

	system), the AR3 platform meets all the requirements to support vessel-based operations. In fact, this platform is launched using a highly mobile catapult system and can be recovered by parachute and airbags (standard recovery option for land-based operations) or with a net system (specifically designed for maritime operations), making it extremely easy to assemble, operate and store.
Expected input	Internal to the system: telecommands such as flight surfaces, commands, thrust commands, payload control signals and system messages; Provided by the operator (external via GCS): waypoints (point and click on map or coordinates Lat/Long/Alt); circuits; areas of interest for payload coverage.
Expected output	Payload data (format depends on the type of payload): cameras provide videos and imagery; Telemetry data and system diagnosis data (including altitude, position, heading, attitude, thrust, airspeed, ground speed, battery levels, etc.).
When it is used	UAVs will be used during the live-demo exercises defined by ARESIBO.
Relation to other components	Consumes mission's data. Disseminate UAV video and sensor data; UAV flight data; system status.
Candidate technologies to be used	TEKEVER AR3 Net Ray
Relevant Tasks	Task 6.2 – Software and hardware integration at component-level

	Task 6.3 – Full-system integration, customisation and operational testing T3.4 Data communication protocols and cyber-security
Comments	N/A

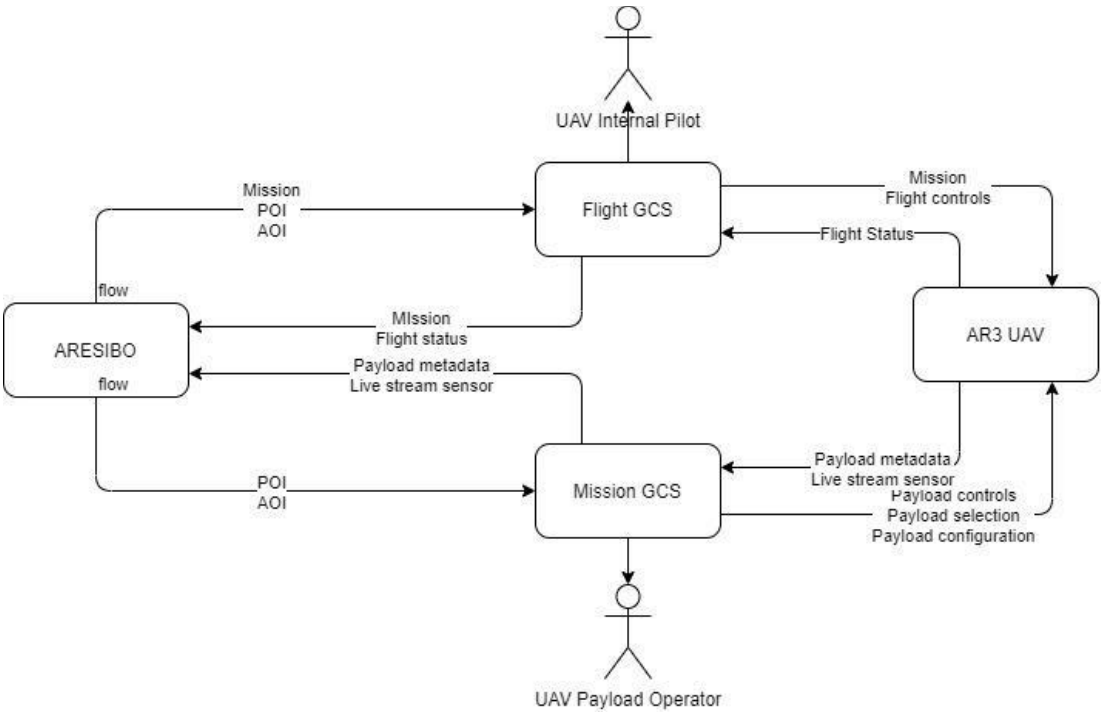

Figure 9: Package Diagram for ARESIBO UAV

5.1.3 Unmanned Sea-surface Vehicle and Unmanned Underwater Vehicle

In this section, we present a summary of the communication structure of UUV and USV vehicles. The communication of these vehicles is based on the Manta Gateway that provide WIFi network for all components to access and be able to communicate.

The driver of the vehicle will have a computer connected to the network and with Neptus running. Neptus is a software used to create plans and monitor the execution and status of vehicles.

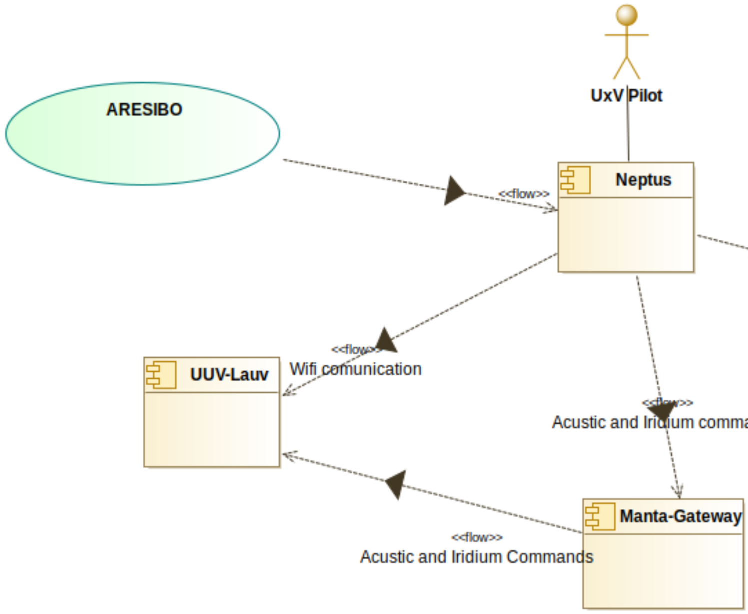

Figure 10: Package diagram for ARESIBO USV

An example of behaviour during a mission sent by ARESIBO, validated by the pilot and performed by the vehicle, is described in the chart below.

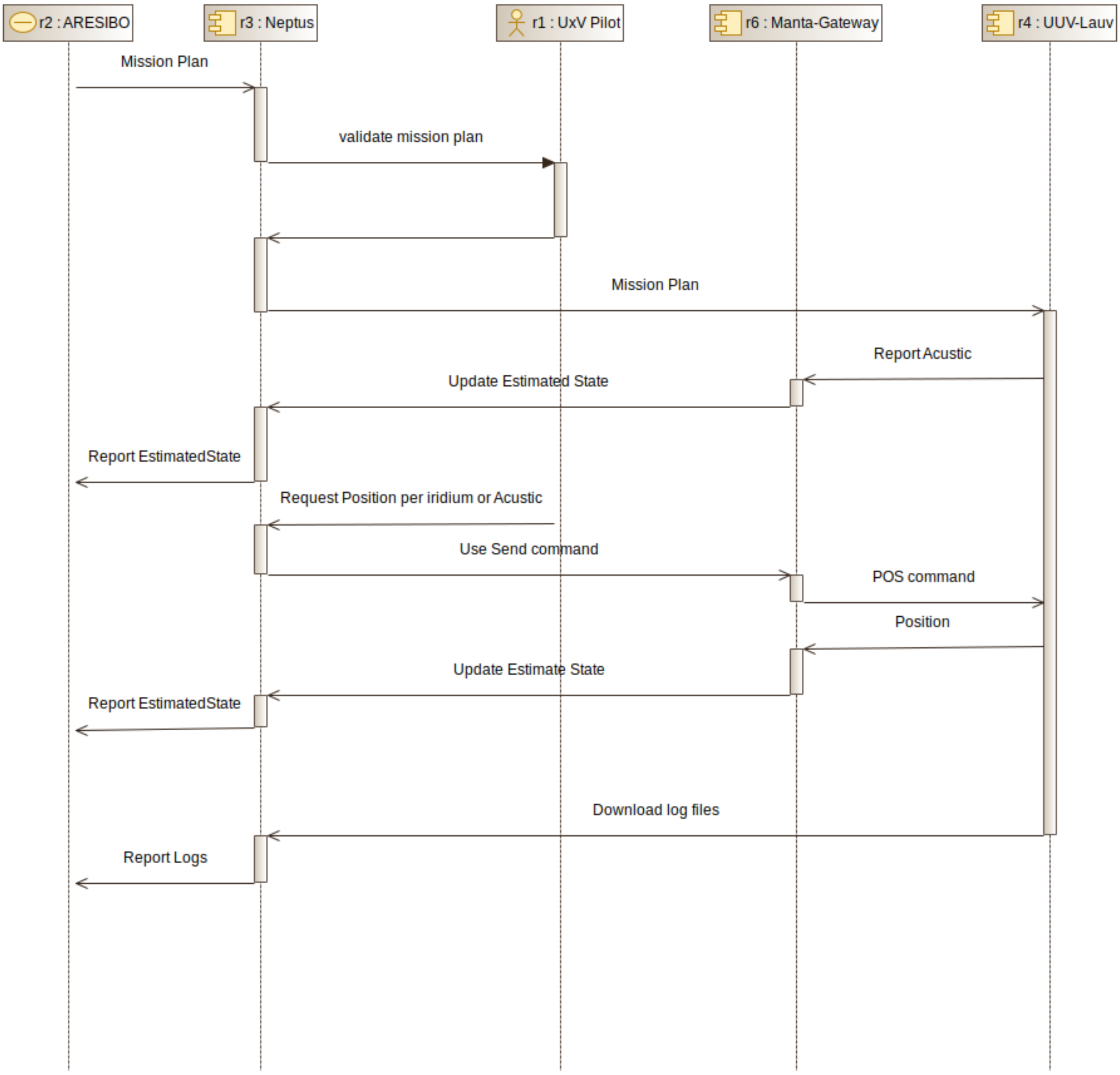

Figure 11: Sequence diagram for ARESIBO USV

Component name	**Unmanned Surface Vehicles (USV)**
Responsible partner	OceanScan –MST
Provided functionalities	USV are robots with autonomous navigation with the ability to navigate in surface of water.
Technical Description	USV is equipped with navigation sensors (GPS) for autonomous navigation, communication system (Wifi, GSM). USV communicates with the Manta-Gateway via WI-FI

	connection. In cases of operation outside the WIFI range it is possible to send commands to the UUV per SMS
Expected input	The mission plan to be executed containing the set of maneuvers to be performed with position and payload used.
Expected output	At the end of the plan execution, the logs containing the executed trajectory.
When it is used	USVs will be used during the live-demo exercises defined by ARESIBO.
Relation to other components	Consumes missions from the Dashboard.
Candidate technologies to be used	IMC Toolchain, Dune, Neptus
Relevant Tasks	Task 6.2 – Software and hardware integration at component-level Task 6.3 – Full-system integration, customisation and operational testing
Comments	It is important to validate the plan sent by the Resource Controller by a USV Operator who is at the place where the USV will be used and can validate the need for adjustments in relation to local conditions. In addition to monitoring the execution of the mission to be aware of the local marine traffic.

Component name	**Unmanned Underwater Vehicles (UUV)**

Responsible partner	OceanScan –MST
Provided functionalities	UUV are robots with autonomous navigation with the ability to navigate under water up to 100m deep. The main objective of UUV is to scan and locate objects on the seabed.
Technical Description	UUV is equipped with navigation sensors (GPS, AHRS) for autonomous navigation, communication system (Wifi, GSM, Iridium) and Side-scan Sonar to analyze the seabed. It also has a black and white camera to film the seabed at times when the quality of the water allows the capture of images.On the surface the UUV communicates with the Manta-Gateway via WI-FI connection and when submerged it reports its position via the acoustic modem. In cases of operation outside the WIFI range it is possible to send commands to the UUV per SMS or Iridium.It has a set of internal batteries that allow an estimated operating time of 4 hours.
Expected input	The mission plan to be executed containing the set of maneuvers to be performed with position and payload used.
Expected output	At the end of the plan execution, the logs containing the executed trajectory, altitude in relation to the seabed, depth are sent. In addition to the data collected by the sensors.
When it is used	When it is necessary to locate a medium or large object on the seabed.
Relation to other components	Consumes missions from the Dashboard.

Candidate technologies to be used	LAUV, IMC Toolchain, Dune, Neptus
Relevant Tasks	Task 6.2 – Software and hardware integration at component-level Task 6.3 – Full-system integration, customisation and operational testing
Comments	When the UUV is submerged, we have no communication to send data in real time. Data can only be sent at the end of the plan's execution. It is important to validate the plan sent by the Dashboard by a UUV Operator who is at the place where the UUV will be used and can validate the need for adjustments in relation to local conditions. In addition to monitoring the execution of the mission to be aware of the local marine traffic.

5.1.4 UxV Sensors

Component name	**W/B Camera for UUV**
Responsible partner	OceanScan-MST
Provided functionalities	Camera providing images in the visible spectre.
Technical Description	The black and white camera allows you to have images of objects in bed in good conditions.
Expected input	Start and stop recording
Expected output	Video mjpeg of what was recorded by the camera.

When it is used	Used to confirm that the point of interest detected by the Side-Scan Sonar is the object you are looking for.
Relation to other components	UUV
Candidate technologies to be used	Genie Nano GigE, MJPEG
Relevant Tasks	Task 3.2 Sensor infrastructure and wearables for field operations
Comments	It is not possible to send images in real time due to the low communication capacity of the submerged UUV.

Component name	**Side-Scan Sonar for UUV**
Responsible partner	OceanScan-MST
Provided functionalities	Provide sonar data to seabed
Technical Description	The black and white camera allows you to have images of objects in bed in good conditions.
Expected input	Start and stop scanning
Expected output	Recorded sonar data
When it is used	Used to map the seabed and detect possible points of interest that may be the object being sought.

Relation to other components	UUV
Candidate technologies to be used	Genie Nano GigE, MJPEG
Relevant Tasks	Task 3.2 Sensor infrastructure and wearables for field operations
Comments	It is not possible to send data in real time due to the low communication capacity of the submerged UUV.

Component name	**Communication Link for UUV/USV**
Responsible partner	OceanScan-MST
Provided functionalities	It allows the communication between the UUV / USV with the Manta gateway and the UxV pilot computer.
Technical Description	WIFI communication network using the 2.4GHz frequency, with maximum range in optimal conditions of 400 meters
Expected input	None.
Expected output	None.
When it is used	During the entire UUV / USV operation.

Relation to other components	Allows communication between USV / UUV and the pilot's computer
Candidate technologies to be used	Long-Range WIFI.
Relevant Tasks	Task 3.2 Sensor infrastructure and wearables for field operations
Comments	Attention to the detail that the USV / UUV and the pilot's computer must be at most 400 meters away from the Manta Gateway in order to be able to use the communication and control of the USV / UUV. Outside this range, the vehicle will perform what was previously ordered until receiving new commands.

Component name	**GPS for UGV**
Responsible partner	Robotnik
Provided functionalities	GPS providing the position of the UGV in a world reference frame.
Technical Description	GPS is a system that is capable of providing the position of a device using satellite trinagulation. The precision depends on the quality of the receivers and the number of satellites visible.
Expected input	None.

Expected output	Position of UGV in a world reference frame
When it is used	It is used by the UGV to navigate between the waypoints during the execution of a mission.
Relation to other components	Provides the position of the UGV to the Dashboard.
Candidate technologies to be used	Trimble MB-Two, UBLOX ZED-F9P
Relevant Tasks	Task 6.2 – Software and hardware integration at component-level Task 6.3 – Full-system integration, customisation and operational testing
Comments	High precision GPS is required to navigate, preferable RTK precision

Component name	**3DLidar for UGV**
Responsible partner	Robotnik
Provided functionalities	Lidar sensor providing range data
Technical Description	Range data is essential for sensing the environment, to navigate in a safe manner while avoiding obstacles.
Expected input	None

Expected output	Livestream of range data.
When it is used	It is used in the missions of the UGV.
Relation to other components	Produces a live stream of range data that is fed to the navigation system of the UGV.
Candidate technologies to be used	RSLidar16, Benewake CE30C.
Relevant Tasks	Task 6.2 – Software and hardware integration at component-level Task 6.3 – Full-system integration, customisation and operational testing
Comments	Camera must be mounted on a pantilt to allow a fine positioning.

Component name	**Communication Link for UGV**
Responsible partner	Robotnik
Provided functionalities	UGV must be connected to a GCS (Ground Control Station) to be teleoperated, monitored and receive missions from ARESIBO systems.
Technical Description	A reliable communication link must be stablished between UGV and its GCS, while keeping good performance in terms of stability, bandwidth and latency.

Expected input	None.
Expected output	None.
When it is used	It is used to operate the UGV, and during the execution of the missions
Relation to other components	Provides communication between the UGV and the GCS, transmitting the data produced in the UGV (such as live stream of data) to the GCS, and from GCS to ARESIBO.
Candidate technologies to be used	4G Connection, Long-Range WIFI.
Relevant Tasks	Task 6.2 – Software and hardware integration at component-level Task 6.3 – Full-system integration, customisation and operational testing
Comments	Communication is crucial for the correct operation of the UGV

Component name	**RGB Camera for UGV**
Responsible partner	Robotnik
Provided functionalities	Camera providing images in the visible spectre.
Technical Description	RGB cameras are cameras that measure light in the visible wavelength, thus allowing to measure the heat emitted by an

	object. They are used in surveillance operations to analyze the scene and detect intruders or unusual objects.
Expected input	Heading orientation, so camera can be pointed in an specific direction.
Expected output	Livestream of RGB images.
When it is used	It is used in the missions of the UGV.
Relation to other components	Produces a livestream of images that is fed to the Detection systems.
Candidate technologies to be used	Axis M5525-E
Relevant Tasks	Task 3.6 Sensing optimisation: Visual object recognition, Dynamic Data Driven Assimilation (DDDAs) and active sensing
Comments	Camera must be mounted on a pantilt to allow a fine positioning.

Component name	**Thermal Camera for UGV**
Responsible partner	Robotnik
Provided functionalities	Camera providing images in the infrared spectre to sense heat-emitting bodies.

Technical Description	Thermal cameras are cameras that measure light in the infrared wavelength, thus allowing to measure the heat emitted by an object. They are used in border surveillance to measure heat emitting bodies, as they can operate in conditions of low ambient light conditions.
Expected input	Heading orientation, so camera can be pointed in an specific direction.
Expected output	Livestream of thermal images.
When it is used	It is used in the missions of the UGV.
Relation to other components	Produces a livestream of images that is fed to the Detection systems.
Candidate technologies to be used	Axis Q1942, Axis PTT99A11
Relevant Tasks	Task 3.6 Sensing optimisation: Visual object recognition, Dynamic Data Driven Assimilation (DDDAs) and active sensing.
Comments	Camera must be mounted on a pantilt to allow a fine positioning.

5.1.5 Underwater Surveillance Sensors

Component name	**Underwater Surveillance Sensors**
Responsible partner	ADMES

Provided functionalities	The installation of hydrophones will provide monitoring of underwater and surface noise. The system it will be capable of recording data from surface ships that travelling at very low speed (approx. 10kts) and from underwater vehicles inside a 2 nautical miles diameter.
Technical Description	The hydrophones will be installed on buoys and they will operate independently. They will provide data to Manta Gateway for further transmission or analysis to control room. It will be design for long term operation with solar panels and batteries.
Expected input	The installation will provide data in frequencies from 1 to 60kHz on maximum depth of 900m. The noise of surface and subsurface vessels will be captured (in a range of 2 nautical miles) and sent wirelessly to a data logger.
Expected output	Vessel noise monitor and detection
When it is used	The equipment will be installed at Skaramagkas region in Athens for PUC3 requirements.
Relation to other components	Data Storage API KAFKA system
Candidate technologies to be used	Java, Javascript/ HTML, Matlab
Relevant Tasks	Task 2.2 Sensor infrastructure and wearables for field operations Task 3.5 Real time sensor fusion and interoperale provision of sensor data

	Task 6.3 Large – Scale demonstrator for integrated situation awareness in land borders
Comments	N/A

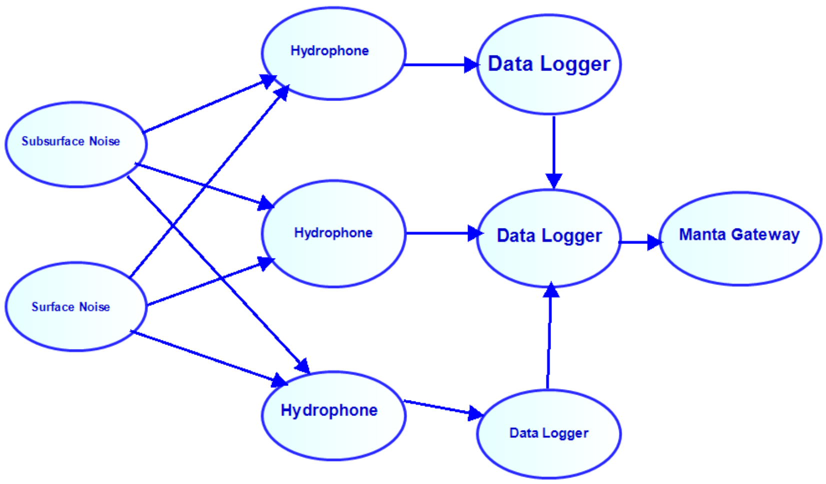

Figure 12: Connections between Sensors and sub-modules

5.1.6 Network component and ARESIBO Island Gateway

5.1.6.1 Radio Network

The main goal of the radio network is to provide connectivity to field units (first responders) and UxVs in remote operation areas which are out of coverage of traditional communication networks. Furthermore, this radio network has to provide high range connectivity to stationary and highly mobile units with low latency for time-critical missions.

This is accomplished by deploying a self-forming/self-healing Mobile Ad Hoc Network (MANET), which provides mission critical video, voice, situational awareness, and data sharing without depending on an established infrastructure. The MANET is established between TrellisWare radios which are available in full-featured handheld (equipped on the first responders) and vehicular configurations, as well as small form factor modules for easy integration into unmanned systems (equipped on the UAVs).

5.1.6.1.1 Trellisware MANET

In addition to the audio channels, each radio module supports H.264 video streams with 8 Mbps IP throughput per channel. Depending on the model of the radio terminal and the resolution of the video streams, the MANET can support multiple video streams simultaneously. Furthermore, the Trellisware radio terminal can act as a relay to other radios in range, thus creating a mesh radio network with a large range (20 km LOS per network hop).

The radio modules (equipped on first responders and UAVs) will send or relay all the data streams to the communication hub where the traffic will be routed to the C2 center.

The Trellisware tactical mobile ad-hoc network has the following features:

- Not dependent on a fixed infrastructure - no towers needed.
- No central control points, programmed routes or tables, access points, or directional antennas (as needed for cellular or Wi-Fi networks).
- No restrictions on topology.
- No restrictions on the number of radios to host in a single network.
- No special setup required to connect to a computer or existing network.
- Every radio is a receiver, transmitter, and relay – all in one.
- Each radio directly communicates with other radios for all network traffic.
- Reliable communications with low latency and low overhead.

An important feature of the Trellisware MANET is that the entire MANET increases in robustness, area of coverage, and path diversity as more radios are deployed.

Figure 13 - Trellisware MANET in an urban environment

The Trellisware tactical mobile ad-hoc network provides the following advantages:

Network Benefits

- Self-forming, Self-healing - Infrastructure-less Mobile Ad-hoc Networking
- Scalable - Large and small network capability, 200+ node network in actual user deployment
- Fast re-entry - Less than one second
- Transparent IP routing - IP devices are plug and play

Services of Network

- Voice, Data, PLI - Simultaneous voice, IP data, GPS, gateway, video encoding, external devices
- Cellular quality voice - 12 channels, AMR5.9 (GSM), or MELPe
- Video - Capable of multiple simultaneous video streaming (H.264, MJPEG)

- Data rate - 8 Mbps IP throughput
- Multi-hop - Up to 8 hops
- Mobility - Instantaneous network anywhere at any time - (vehicle to
- ground to air)
- Harsh RF Environments - Urban, ship, building, tunnels, etc.

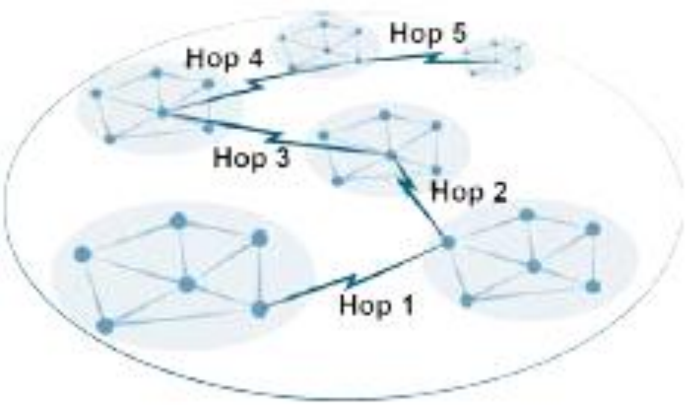

Figure 14 - Trellisware MANET with multiple hops

The radio terminal used in the ARESIBO project is the Trellisware "Cub" with the functional diagram and specifications shown in Figure 15 and Figure 16, respectively.

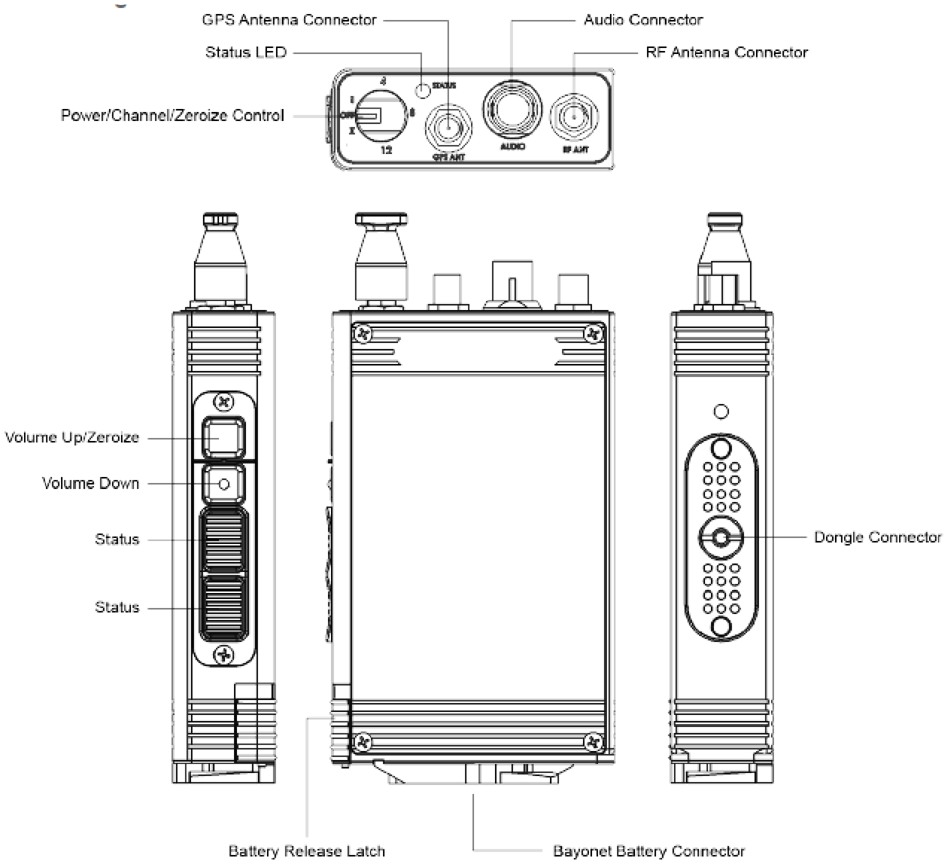

Figure 15: Trellisware Cub diagram

Features	Specifications
Transmit Power	2 Watt peak
Frequency Range	1775-1815 MHz, 2200-2250 MHz
Hop Reach	26 miles per hop (up to 8 hops)
Security	AES-256, remote disable, RSA-2048 and SHA-256 authentication
Size (w/o accessories)	4"(H) x 2.5" (W) x 0.9" (D)
Weight	10 oz (R/T only)
Throughput	Up to 8 Mbps
Environmental Compliance	MIL-STD-810F with 2 meter immersion (R/T)
Host Interface	Ethernet, USB, Wi-Fi, BNC analog video, trigger-in/out, power out
Connectors	LEMO audio, SMA RF and GPS antennas, 24-pin for dongles
Input Power	12-16 V DC
Video/Audio Encoding	MJPEG or H.264 video, AAC audio, AMR 5.9 or MELPe for PTT voice
Audio Channels (and latency)	Up to 12 independent channels (3 hop <250 ms, 8 hop <400 ms)
Net Entry Time	Less than 1 second
Battery Life/Power Conservation	1+ week continuous idle, 30+ days Wake on Wireless, 10+ months Wake on Trigger (on dual BA 5590's)
Occupied Bandwidth	20 MHz, tunable down to 4 MHz
IP Support	Unicast, multicast, broadcast

Figure 16: Trellisware Cub specifications

Figure 17 - Trellisware Cub terminal

5.1.6.2 Communication hub

The communication hub connects the radio network to the C2 center by using two different networks: traditional LTE network and Satellite network. This allows the communication hub to use the LTE network with lower latency when the vehicle is in LTE coverage area and switch to using the Viasat satellite network in cases when there is no LTE coverage, which is very common for the use cases in the ARESIBO project. This will ensure high reliability and availability even in rough terrains and changing conditions. Figure 18 illustrates a detailed schema of the ARESIBO network architecture.

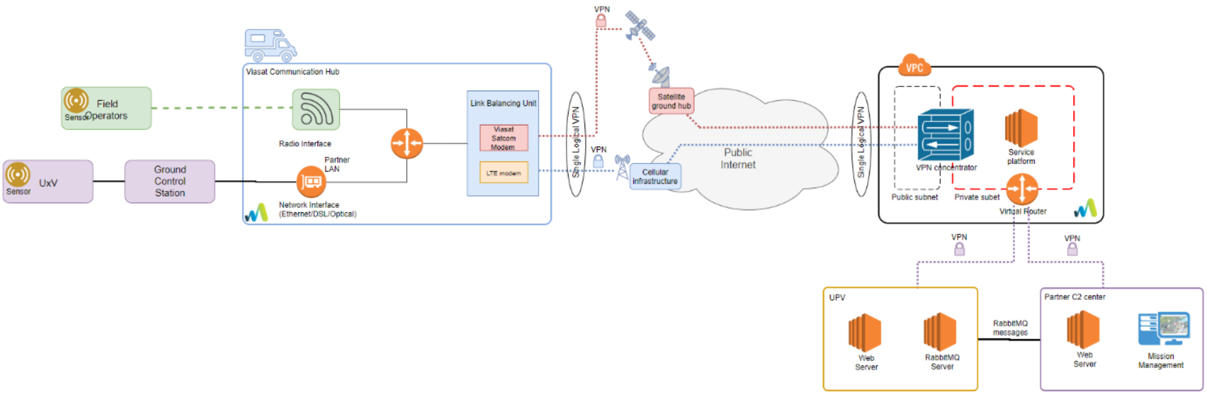

Figure 18 - Detailed network architecture diagram

5.1.6.2.1 Link failover

In order to ensure reliability and availability in different network scenarios, two separate VPN tunnels (one over LTE and the other over the Satellite link) are established between the Link Balancing Unit on-board the Viasat communication hub (vehicle) and the VPN concentrator hosted in private cloud which terminates the VPN tunnels. The Link Balancing Unit and VPN concentrator manage the VPN tunnels to provide Active/Standby failover between the two links (LTE and Satellite). The VPN tunnels act as a single logical VPN tunnel by performing packet-based fail-over between the links.

The communication hub can establish the satellite link by utilizing the Dawson “Dawson SC Zero 70 KA-SAT” fully automated system or the self-pointing Viasat “KaLMA” satellite antenna.

Figure 19 - Viasat Land Vehicle (side)

5.1.6.2.2 Dawson satellite antenna system

The Dawson SC Antenna is a fully automatic flyaway antenna system that operates on the satellite Eutelsat KA-SAT as well as other Viasat networks around the world. The full system specifications are presented in Figure 21.

Figure 20 - Dawson SC Zero 70 KA-SAT antenna system

Section	Item	Specification
Power requirement	**Prime power** **Power consumption**	220 V AC (12 V DC optional) Approximately 100W
Physical	**Antenna box** **Equipment box** **Positioner box**	850 H x 820 W x 180 D (mm) Wt 14 Kg 240 H x 560 W x 620 L (mm) Wt 19 Kg 410 H x 360 W x 1120 L (mm) Wt 34 Kg
Operation	**Set up / Pointing time** **Pointing system**	<4 minute typical Fully automatic Manual over ride and emergency stow
Environmental	**Temperature (use)** **Wind speed** **Rain**	Minus 40 to Plus 60 C 70 Kph use 160 Kph stow (with ballast) Fully waterproof in use
Construction	**Antenna system** **Reflector** **Base**	Aluminium and stainless steel ViaSat OEM part Stainless steel tripod
Positioner	**Elevation range** **Azimuth range** **Polarisation range**	10 - 90 Degrees fully motorised 200 degrees +/- fully motorised Automatic switching of polarity
Performance	**BUC size** **Data rate TX** **Data rate RX**	Standard ViaSat E TRIA KA-SAT services supported KA-SAT services supported
RF	**Antenna**	Eutelsat KaSat
	Frequency	KaBand
	Compatibility	Eutelsat KA-SAT
	Antenna size **Feed**	75 cm nominal Circular Polarised KaBand

Figure 21 - Dawson SC Zero 70 KA-SAT specifications

5.1.6.2.3 KaLMA satellite antenna

The KaLMA antenna is a self-pointing Ka-band satellite antenna mounted on the communication hub (Viasat Land Vehicle). This antenna is used to establish a link to the satellite, thus creating the backhaul link through the Viasat core satellite network.

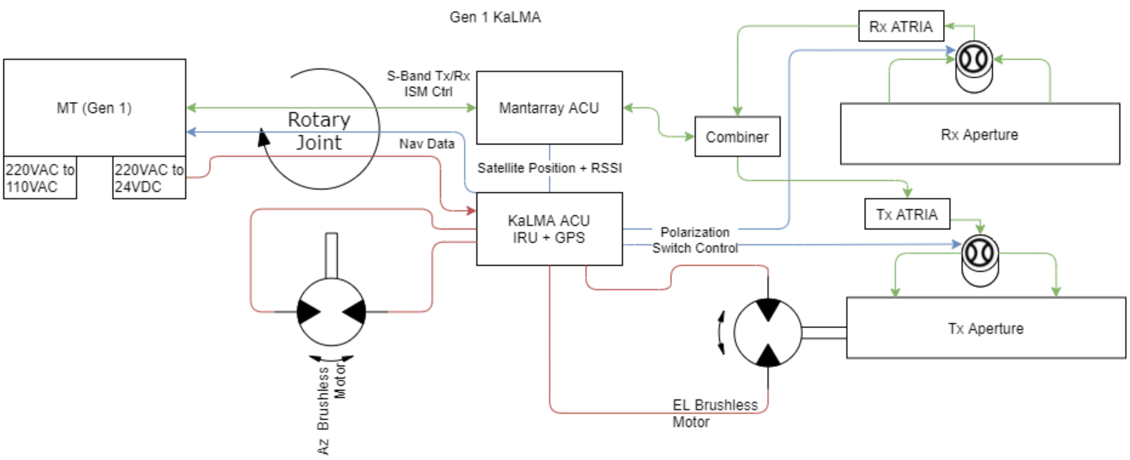

Figure 22 - Architecture diagram of the KaLMA antenna

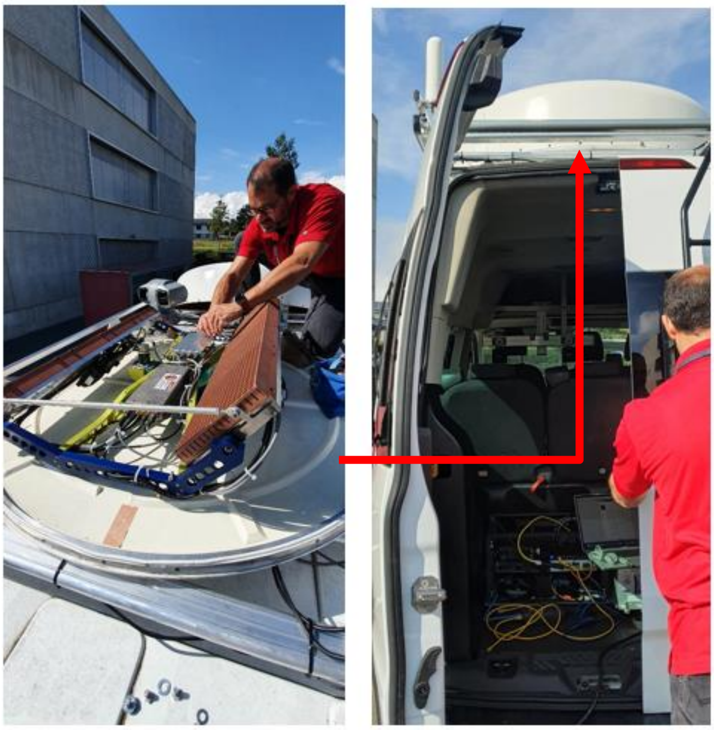

Figure 23 - KaLMA antenna mounted on the roof of the Viasat Land Vehicle

5.1.6.2.4 A maritime use case

In this section we describe a maritime use case in which the Trellisware MANET is utilized for the radio communication between the field units and a satellite link as backhaul. The satellite link is established by a Cobham Sailor satellite antenna mounted on a ship that will act as the communication hub for the other field unites (vessels, UAVs and helicopters). The Cobham Sailor satellite antenna, shown in Figure 24 operates in Ku-Band and Ka-Band (Rx: 10.70 to 12.75 GHz, Tx: 13.75 to 14.50 GHz) and has overall dimensions: Height: 150 cm and Diameter: 130 cm, which makes it ideal for deployment on a larger ship.

Figure 24 - Cobham sailor satellite antenna

As shown in Figure 25 the ship acts as the communication hub and all the field units are equipped with Trellisware radio terminals. These terminals have a single hop range of 30km with the ability to relay signals from other terminals for up to 8 hops, thus providing a large covering range. This means that if a field unit equipped with a Trellisware terminal is not in range to directly communicate with the communication hub, it can use another field unit in range as a relay to communicate with the communication hub.

The communication hub receives the data streams from the field units and uses the satellite link to establish a bidirectional connection with the C2 center via the Viasat satellite network.

The complexity of the installation of the Cobham Sailor satellite antenna and the requirements for the vessel that it infers, make this solution very difficult to demonstrate on the field. In order for the demonstration to be possible, several modifications need to be made:

- Update the design of the vessel with static and dynamic structural verification
- Cabling of the area where the antenna will be installed providing:
    - Power
    - Ethernet
    - Compass and Gyroscope data
    - RF coaxial cabling
- Testing the vessel equipment for non-interference with actual systems.

A demonstration of a maritime use case using a satellite link as backhaul may be set in place using existing satcom-ready vessels even if the installed equipment will not be as proficient in terms of data throughput as the one proposed by Viasat.

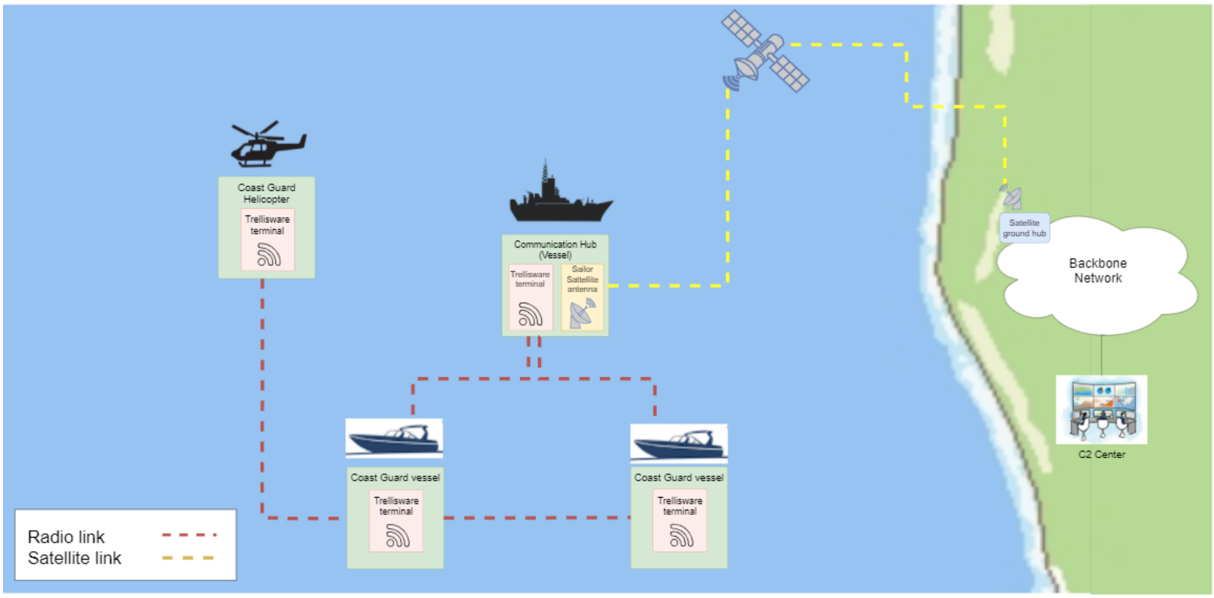

Figure 25: Network architecture schema for the maritime use case

5.1.7 Voice and Video communication

The goal of Viasat in terms of video and voice communications is to provide connectivity between the on-field operators and the corresponding C2 center. As described in the previous section, the communication hub provides connectivity to the on-field operators by using LTE and satellite links as backhaul. The used backhaul link is completely transparent to the voice and video applications used by the on-field operators and the corresponding C2 center.
In order for voice and video streams to be exchanged between the on-field operators and the C2 center, the C2 center has to provide a public IP address which is reachable from the communication hub in order for the on-field operators to be able to initiate the connections to the C2 center. This means that IP cameras mounted on the field will have to push the real-time video stream to the C2 center's public IP address. In order to demonstrate an implementation of this proposal, the following section presents several real-time video tests performed by Viasat.

5.1.7.1 Real-time video tests

The goals of this test are to evaluate the capability of the communication network to transport real time HD video streams and observe how the video continuity, quality and latency is affected by the fail-over between the backhaul links. The video source is the 'AXIS Q6215-LE' IP camera on-board the communication hub (Viasat Land Vehicle).

The tests defined in this category are separated into 2 test scenarios. The first test scenario consists of tests where the real time video stream from the Axis camera is played by accessing the web interface of the camera from an EC2 instance. The second tests scenario consists of tests where the real time video stream is played using a video player (in this case - VLC) hosted in an AWS EC2 instance using RTSP pull. The detailed setup of the testing scenarios is illustrated in Figure 26.

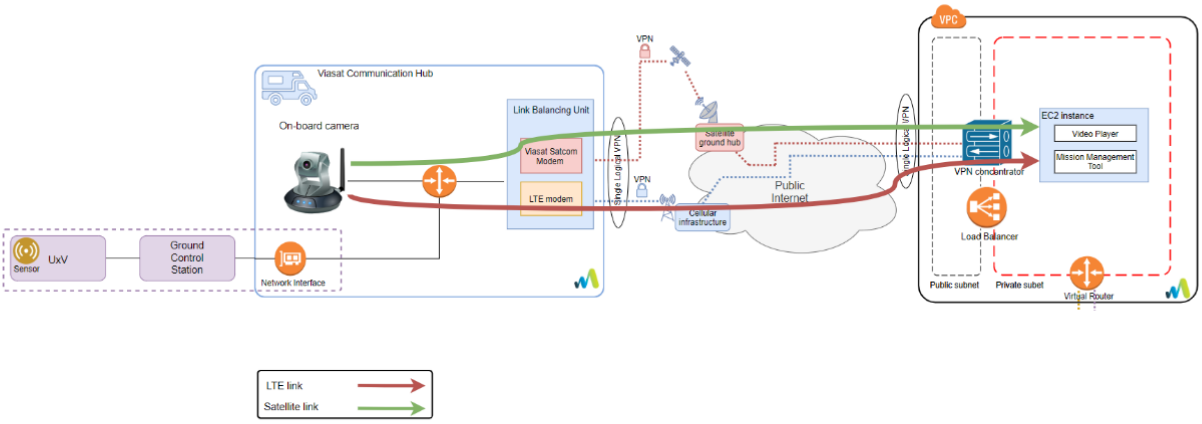

Figure 26 - Diagram for the RT video tests

HTTP tests

- **TST_COMS_003_1**: Real-time HD video stream with H.264 compression using HTTP with LTE as backhaul.
- **TST_COMS_003_2**: Real-time HD video stream with H.264 compression using HTTP with satellite as backhaul.

VLC tests

- **TST_COMS_003_3**: Real-time HD video stream with H.264 compression using RTSP and VLC with LTE as backhaul.
- **TST_COMS_003_4**: Real-time HD video stream with H.264 compression using RTSP and VLC with satellite as backhaul.

Test case ID	
TST_COMS_003_1	
Test description	Real-time HD video stream with H.264 compression using HTTP with LTE as backhaul.
Test Scenario	
This test is performed to evaluate the ability of the communication network to transport an H.264 real time video stream with a resolution of HD using LTE as backhaul. The main goals of the test are to observe the quality of the video (whether there are any disruptions to the video) and the latency. In order to do this, we play the video from the Axis camera using HTTP by accessing the camera web interface by its IP address: 192.168.10.11 from an EC2 instance and observe the difference in the timestamps between the shown video in the EC2 instance and the video shown on a laptop connected physically in the same subnet as the Axis camera.	
Test Results	

Video quality	Video is played without disruption
Latency	0.5-1s

Test case ID	
TST_COMS_003_2	
Test description	Real-time HD video stream with H.264 compression using HTTP with LTE as backhaul.
Test Scenario	
This test is performed to evaluate the ability of the communication network to transport an H.264 real time video stream with a resolution of HD using LTE as backhaul. The main goals of the test are to observe the quality of the video (whether there are any disruptions to the video) and the latency. In order to do this, we play the video from the Axis camera using HTTP by accessing the camera web interface by its IP address: 192.168.10.11 from an EC2 instance and observe the difference in the timestamps between the shown video in the EC2 instance and the video shown on a laptop connected physically in the same subnet as the Axis camera.	
Test Results	
Video quality	Video is played without disruption
Latency	2-3s

Test case ID	
TST_COMS_003_3	
Test description	Real-time HD video stream with H.264 compression using RTSP and VLC with LTE as backhaul.
Test Scenario	
This test is performed to evaluate the ability of the communication network to transport an H.264 real time video stream with a resolution of HD using RTSP and LTE as backhaul. The main goals of the test are to observe the quality of the video (whether there are any disruptions to the video) and the latency. In order to do this, we play the video from the Axis camera using an RTSP pull with VLC from an EC2 instance. In addition, we observe the difference in the timestamps between the shown video in the EC2 instance and the video shown on a laptop connected physically in the same subnet as the Axis camera.	
Test Results	

Video quality	Video is played without disruption
Latency	0.3-0.5s

Test case ID	
TST_COMS_003_4	
Test description	Real-time HD video stream with H.264 compression using RTSP and VLC with satellite as backhaul.
Test Scenario	
This test is performed to evaluate the ability of the communication network to transport an H.264 real time video stream with a resolution of HD using RTSP and satellite as backhaul. The main goals of the test are to observe the quality of the video (whether there are any disruptions to the video) and the latency. In order to do this, we play the video from the Axis camera using an RTSP pull with VLC from an EC2 instance. In addition, we observe the difference in the timestamps between the shown video in the EC2 instance and the video shown on a laptop connected physically in the same subnet as the Axis camera.	
Test Results	
Video quality	Video is played without disruption
Latency	1-1.5s

5.1.7.2 Voice Communications

In order to evaluate the performance of the proposed network solution in terms of voice communications, we have identified a VoIP implementation that closely simulates the applications that will be used by the partners in the ARESIBO project. The proposed VoIP application is a free and open-source software based on the Tox protocol which is a peer-to-peer instant-messaging and video-calling protocol that offers end-to-end encryption.

5.1.7.2.1 Tox protocol

Users are assigned a public and private key, and they connect to each other directly in a fully distributed, peer-to-peer network. Users have the ability to exchange messages, join chat rooms, voice/video chat, and send each other files. All traffic over Tox is end-to-end encrypted using the NaCl library, which provides authenticated encryption and perfect forward secrecy. Tox uses the cryptographic primitives present in the NaCl crypto library, via libsodium. Specifically, Tox employs Curve25519 for its key exchanges, xsalsa20 for symmetric encryption, and Poly1305 for MACs. Due to the fact that the tox protocol can be used by many different applications and the tox network broadcasts the used client, it is also possible for clients to use additional encryption when sending to clients which support the same features.

5.1.7.2.2 Test setup

The test set up for the Tox protocol will consist of running a client (“qTox” client for Windows) on a laptop on-board the communication hub and another client in an instance in the AWS VPC. These clients will use the Tox core library to communicate with each other using the Tox protocol. The diagram of the testing setup is illustrated in Figure 27.

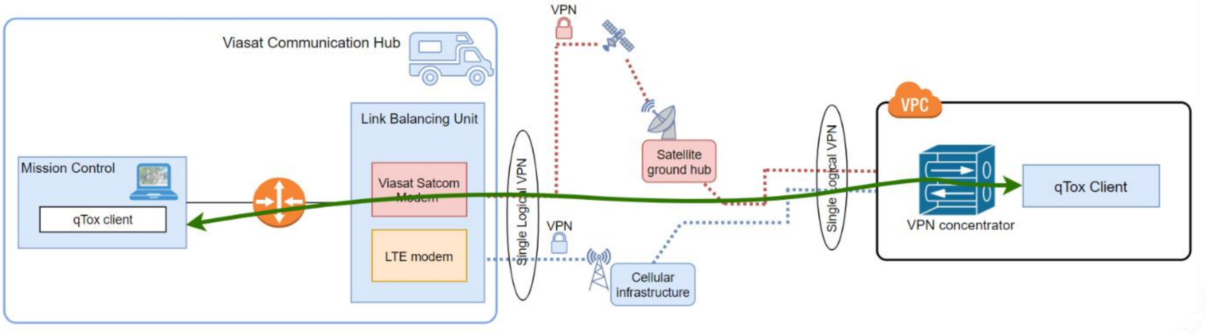

Figure 27 - Voice communications test setup diagram

5.1.8 Collective intelligence for UxV swarming and optimised human-robot collaboration module

Component name	**Resource Controller**
Responsible partner	CEntre for Research & Technology Hellas – CERTH
Provided functionalities	Resource Controller is the module responsible for translating high-level objectives and commands defined by the end-users and the C2 operators for a mission, to low-level commands and actions for the involved assets. One of the major tasks of Resource Controller is to calculate paths for a team of UxVs, in order to fulfil in the best way possible an operator defined high-level objective e.g. completely cover a region of interest.
Technical Description	Resource Controller is a back-end module. This means that the operator of the ARESIBO platform does not have any

	straight interaction with it. The interaction of the Resource Controller module with the operators, takes place through the front-end modules of the platform. This module incorporates a set of state-of-the-art multi-robot path planning algorithms, optimized for the scenarios that will be faced in the project. These algorithms allow the operator to define high-level objectives for a mission, without caring about technical details to carry them out.
Expected input	Operator defined objectives and involved assets for a mission. For example, this information could include: • a polygon Region of Interest in WGS84 coordinates • the objective that it should be completely covered • the number and name of UxVs that will be utilized for this mission • the detection tasks that should take place • the sensors that will be used for them.
Expected output	• A set of guidelines and instructions for the UxVs and possibly other modules of the ARESIBO platform involved in a mission. • A message describing the mission commands and actions, as calculated from the Resource Controller, formatted in a way appropriate for visualization to the operator.
When it is used	The Resource Controller module will be used every time that a mission needs to take place through the ARESIBO platform, as after the definition of it, this module is responsible to process it and provide specific commands and actions for the involved assets.
Relation to other	Consumes information from the MDL Editor and the UxVs' GCSs. Produces output for the UxVs GCSs, the ARESIBO

components	Dashboard and other modules involved in a mission that need to take actions.
Candidate technologies to be used	Java, Python, Apache Kafka, Docker
Relevant Tasks	Task 3.3 Collective intelligence for swarming robots and optimised human-robot collaboration.
Comments	The modules that will consume output from the Resource Controller, apart from the Dashboard and the GCSs, will be specifically defined during the integration stage, as it is likely that new demands will arise at this stage (e.g. activation of some modules).

In the following diagram is shown the communications that are expected to take place during the execution of Resource Controller. The inputs expected are the mission parameters as defined in the Mission editor and the telemetry of the vehicles from their GCSs. The produced outputs should be consumed by the GCSs (commands/waypoints), the UIs of the ARESIBO platform, e.g. Dashboard, (visualization messages etc.) and any other module that will probably need trigger or activation.

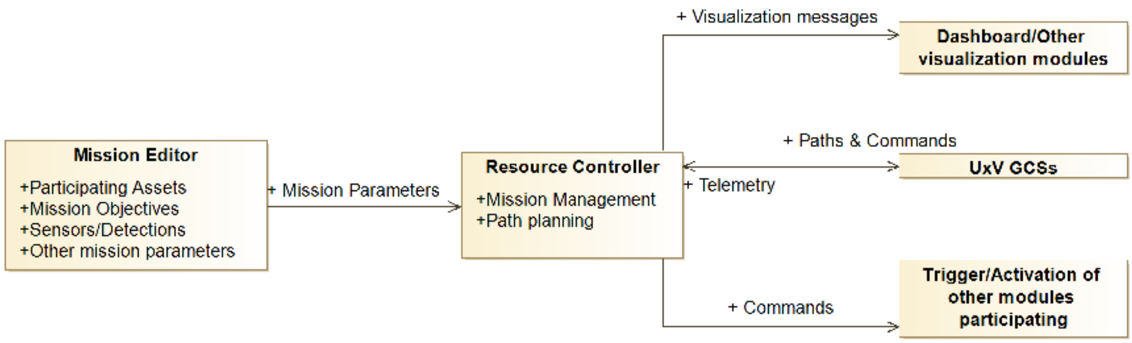

Figure 28: Diagram for ARESIBO Resource Controller

5.1.9 Visual recognition module

Visual recognition (VisRec) module is responsible to process visual data acquired from various cameras such as RGB and thermal deployed on the ARESIBO assets and the pilot sites. The outcomes of the analysis process are intended to contribute to the augmentation of sensing

and contextual understanding, through the identification of specific objects of interest. General objects will be possible to be identified and tracked such as people, cars, boats etc., but further object classes such as speedboats, cargos are relevant according to the Users' needs (D2.1). To use these classes, the VisRec module will be retrained on sufficiently large data sets.
The VisRec module will receive as input data video streams acquired from different types of optic sensors (cameras with different representation capabilities) and will extract details for the identified objects. A high-level representation of the module is shown in the UML diagram below. The module consists of the following sub-components:

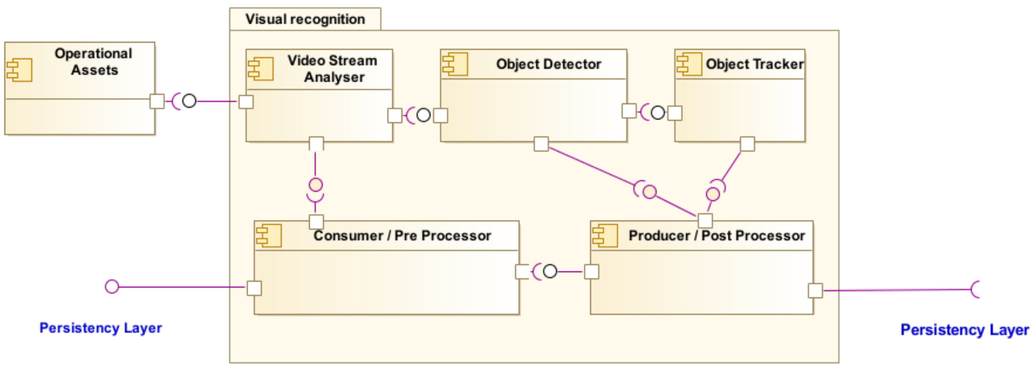

Figure 29: High level representation of the VisRec module

Video stream analyser: A thread per video stream will be used to read each video stream, break it into frames and regulate the frames per second ratio. Frames are put into a queue to be processed by a pool of workers and sent to Object Detector and Tracker.

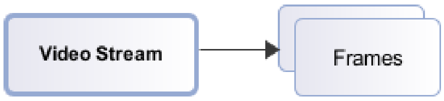

Figure 30: Video Stream Analyser task

Object Detector: The module is responsible to identify multiple objects of interest based on optical data and to provide a set of bounding boxes containing the detected objects and the confidence score per object. The confidence scores reflect how accurate the model is to identify the detected objects.

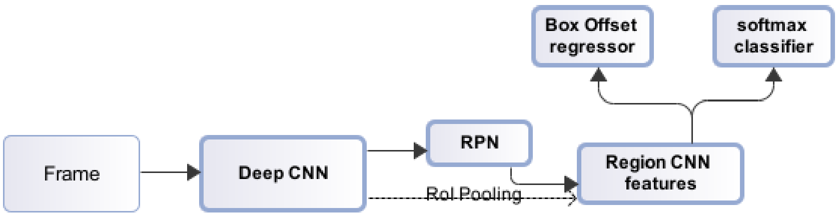

Figure 31: Object Detector will be based on SoTA deep learning architectures such as the Faster R-CNN network.

Object Tracker: The service is responsible to monitor objects' spatial and temporal changes during a video sequence, including its presence, location within the projection plane etc. This is done by solving the temporal correspondence problem which refers in matching the target region in successive frames within an image sequence taken at closely-spaced time intervals. Object detection and object tracking processes operate complementarily since tracking usually is initiated by a detected object of interest, while the detection is repeatedly necessary in subsequent image sequence to improve the tracking process.

As it is included in the following section, communication bus is the main integration tool between different and diverse system components. A typical flow is for a component holding a new piece of information that needs to be processed by another component to publish a message on the corresponding message bus topic, providing in a link to the current location of the information to be processed. The VisRec Consumer is the receiving component that parses the message and triggers the Video Stream analyser to retrieve the visual information via the supplied link for further process. In turn the VisRec Producer collects the new piece of information generated by the Detector and the Tracker and publish the message that need to be passed to another system component.

Component name	**Visual recognition module**
Responsible partner	CERTH
Provided functionalities	Visual recognition module will be responsible to identify specific objects of interests with a resulted frame rate greater or equal to ~> 5fps depending on the configuration of the image acquisition system - detection of objects - tracking of objects
Technical Description	Deep learning architectures will be utilized in order to recognize specific objects from optical data. General object classes will be used such as people, cars, boats etc., but further object classes such as speedboats, dangerous cargoes are relevant.

Expected input	- Video streams. - Single Image files (e.g. jpg, bmp); - Information about resources such as coordinates, altitude gimbal angle of the attached cameras, timestamps
Expected output	- .json with the prediction scores for each object that exists inside the image frame and its bounding box (i.e. where this object is located inside the image) - Overlaid video frames that display the detection outcomes
When it is used	The Visual Object Identification module will be constantly online to process data from visual inputs.
Relation to other components	Consumes information from UxVs and visual sensors. Produces information for the highest integration level, e.g. the Knowledge Base reasoning service.
Candidate technologies to be used	- Machine learning libraries; - Tensorflow; - Keras or Torch; - OpenCV.
Relevant Tasks	Task 3.6: Sensing optimisation: Visual object recognition, Dynamic Data Driven Assimilation (DDDAs) and active sensing.
Comments	- Video streams should be live streamed via RTSP or RTP (depending on the available equipment). - Bad lightning conditions will possibly affect the accuracy of the detection when visual data are used

	- Low detection capabilities (execution time) due to insufficient resources could be observed.

5.1.10 Active sensing component

Component name	**Increased Swarm Situational Awareness**
Responsible partner	CEntre for Research & Technology Hellas – CERTH
Provided functionalities	There are certain scenarios where continuously monitoring a region of interest in order to track the evolution of an event is necessary. In such cases, it is easy to understand that off-line path planning methods are not sufficient, as the events taking place will probably be dynamic. The UxV(s) involved in a mission should adjust their position in order to provide information relevant to the event that is monitored. This module continuously calculates and provides to a *swarm of UxVs* the best monitoring positions inside a region of interest, in order to maximize the count and accuracy index of an object detection software, while simultaneously penalizing the overlapping detections (detection of the same object with more than one UxV). The objective of this module is to provide a powerful tool for cooperative monitoring of a region of interest, with increased quantity and quality in detecting objects of interest.
Technical Description	Utilizing an innovative adaptive optimization algorithm (ConvCAO toolset), this module is able to calculate the best monitoring positions for a team of UxVs, in a completely unknown region of interest, in order to maximize the quantity and quality of detected objects. Thanks to its adaptive nature, the algorithm is capable of handling unknown and

	dynamically changing operational areas and tracking dynamic events.
Expected input	• A polygon Region of Interest in WGS84 coordinates • Number and name of UxVs that will be utilized for this mission • The class of objects that should be handled (e.g. cars, people etc.) • The position of UxV participating (continuously) • Videostream / Depth information / Output of detection modules
Expected output	• Best monitoring position for each UxV
When it is used	The Increased Swarm Situational Awareness component will be used in scenarios where detecting a set of objects inside a region of interest, with increased accuracy and decreased possibility on not “losing” some of the objects of interest, inside the region, is critical (e.g. detection of humans in land border trespassing).
Relation to other components	This module consumes: • Information regarding the mission from the Resource Controller (Task 3.3) • Sensor readings • Output of detection modules and produces: • the best next monitoring position for each UxV involved

Candidate technologies to be used	Python, MS AirSim, Yolo V3, Faster R-CNN, Tensorflow, PyTorch, Apache Kafka, Docker
Relevant Tasks	Task 3.6 Sensing optimisation: Visual object recognition, Dynamic Data Driven Assimilation (DDDAs) and active sensing.
Comments	For this module to operate properly, minimum delays between any transfer of information (telemetry, image feed, commands, feedbacks) is critical.

This module receives the mission parameters from the Resource Controller and telemetry / sensor readings from the GCSs and produces the GCSs with the best monitoring positions for each vehicle.

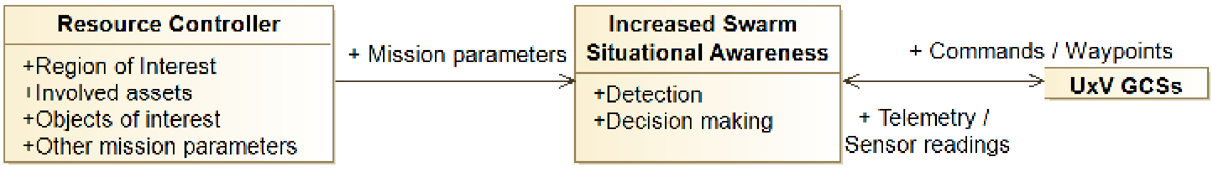

Figure 32: Diagram for ARESIBO Increased Swarm Situational Awareness

5.1.11 Data communication and Cyber-security component

5.1.11.1 Data communication protocols

5.1.11.1.1 Data protocols interoperability

Data communication protocols are of most importance in this project since they are necessary to interconnect all the systems and provide interoperability between all assets. ARESIBO system relies in different types of equipment, from different manufacturers. Each of those system has different communication architectures and use different protocols. This raises big challenges to the ARESIBO system mainly in the interoperability between all assets.

To achieve this interoperability, ARESIBO will define an interoperability layer, which consist of a series of protocol translators.

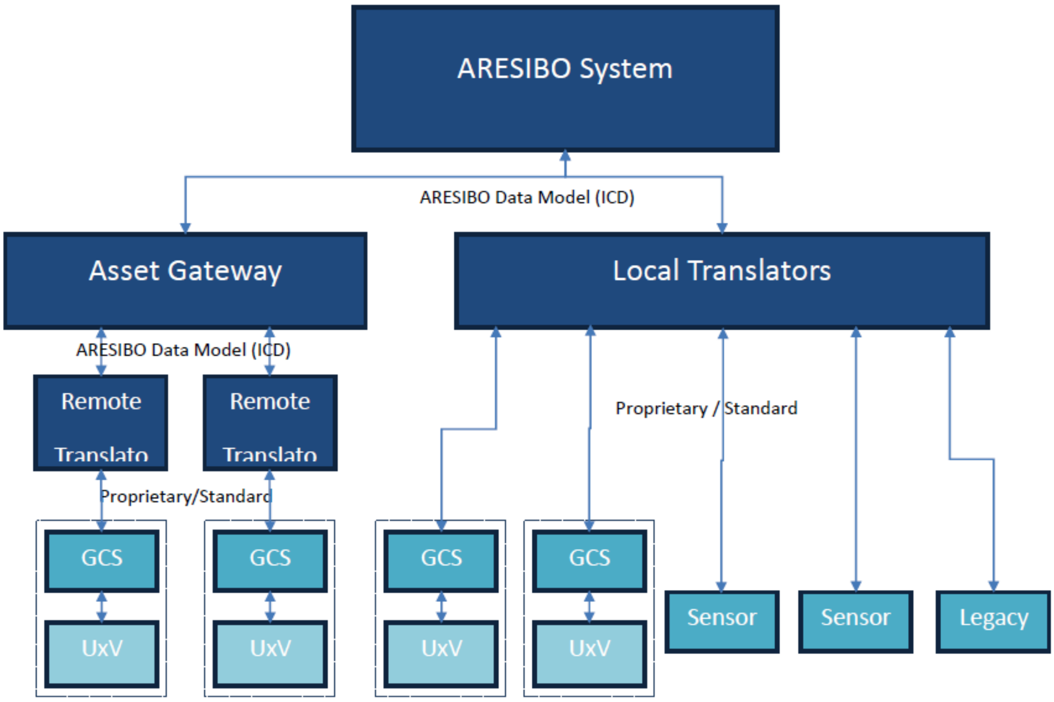

Figure 33: ARESIBO's Interoperability Layer

Each protocol translator will guarantee a proper translation between ARESIBO data model and platform specific protocol.

5.1.11.1.2 Revision of existing standards and protocols adaptors

STANAG 4586 took the first step towards interoperable control segments, documenting a standard for unmanned aircraft/GCS system level command and control interoperability. This is the first step towards enabling GCS to control and monitor multiple types of unmanned aircraft, improving overall cost by reusing GCS, and enabling competition at the system level for complete GCS solutions. https://community.ucsarchitecture.info/dodarchive/mission-and-objectives.html

The aim of STANAG 4609 is to promote interoperability of motion imagery systems in a NATO Combined/Joint Service Environment. This STANAG was developed to enhance the warfighting capability of the forces and increase flexibility and efficiency to meet mission objectives through sharing of assets and common utilization of information generated from motion imagery systems. This STANAG focus mainly on the transmission of imagery.

JAUS stands for Joint Architecture for Unmanned Systems and is a international standard that sets a common set of messages formats and communication protocols for supporting interoperability between unmanned vehicles and control stations. Several JAUS Standard protocols are defined, for different kind of services like core, mobility, environment, manipulator, HMI, mission spooling, ground services and more.

The **Unmanned Aircraft System (UAS) Control Segment (UCS) Architecture** is a software interface, data-model, and business system architecture, defining the rules and conventions for developing interoperable software components for UAS Ground Control Stations (GCS).

The operational objectives include support for both UA platform and sensor C2, sensor product availability, and UA status (https://community.ucsarchitecture.info/dodarchive/mission-and-objectives.html).

5.1.11.2 Cyber-security

To guarantee the cyber-security of the data being exchanged by the several actors of the ARESIBO operational landscape, an additional layer must be built and integrated, that can guarantee:

- That the data exchange between any 2 actors be kept private - although the data itself may be made available to other actors afterwards;
- That the integrity of the data exchanged between any 2 actors be guaranteed by design - so that no other actor can tamper with, insert or remove any part of that data without such action being noticed;
- That the authenticity of the data exchanged between any 2 actors be guaranteed - any part of the data communicated by an actor can only have been sent by that actor;
- That the data exchanged between 2 actors can only be read by any actor down stream if it is properly authorized to do so;
- And finally, every actor is hardened in its exposed interfaces, to reduce the probability of a data breach through the use of an exploit or vulnerability.

5.1.11.2.1 Privacy, Integrity and Authenticity

The first 3 requirements, privacy, integrity and authenticity, can be established at the network level, in which for every level of interface links an appropriate technology must be chosen.

The interfaces link levels are defined to be:

Interface	Level	Data Exchanged
Sensor to local data hub	L1	Control, sensor data
AR equipment to data hub	L2	Control, sensor data, voice, video
Terminal do data hub	L3	Control, messaging, sensor data, voice, video
UxV to Ground Station (short range)	L4	Control, sensor data, video
Data Hub to Island Gateway	L5	Control, messaging, sensor data, voice, video
Ground Station to Island Gateway	L6	Control, messaging, sensor data, voice, video
Island Gateway to Cloud C2	L7	Control, messaging, sensor data, voice, video
Ground Station to Cloud C2	L8	Control, messaging, sensor data, voice, video
UxV to Cloud C2 (long range)	L9	Control, sensor data, voice, video

5.1.11.2.2 High bandwidth and high-performing connections

For levels L5 through L8, and in same cases L3, the communication is fully based on IP protocols, with few constraints on data bandwidth and computing power available for the extra cyber-security overhead, the mechanisms used should be based on COTS security layers, with proven track records and a huge user community to search and resolve vulnerabilities that may be found.

The mechanism put forth for these types of connection is the use of **Transport Layer Security (TLS), combined with mutual or both-way authentication with a PKI certificate**.

This simplifies the management of interconnections between these actors, as no prior knowledge is needed between them except for the respective endpoints, and an agreed upon Public Key Infrastructure.

For connections occurring at L1 through L4 and L9, additional constraints exist, both at bandwidth or connection reliability and at computing power availability.
Therefore, the mechanism to be used will depend on the constraint group.

5.1.11.2.3 Bandwidth constrained OR unreliable connections

In this constraint group, typically L4 and L9 connections, including radio or satellite links, every byte sent over the connection counts, so network overhead must be kept at a minimum, and there must bear a higher tolerance for packet loss. Additionally, since it is typical to use more lightweight transport protocols (layer 4 and higher), such as UDP (example RTP for video), in which, it must be possible to encapsulate data securely at layer 3.
Therefore, the mechanism put forth for these types of connections is the use of **IPsec with Tunnel Mode**. In this mode, the authentication and encryption definitions are defined at tunnel setup, and all subsequent communication bear little data overhead or communication round-trips.
The configuration of these IPsec tunnels requires a point-to-point negotiation and key exchange, but since the components involved are already tightly coupled, this should not be a problem.

5.1.11.2.4 Power constrained component connections

For this constraint group, the major constraint is related to the computing power required for the cyber-security layer. Additionally, some of the connections actually forgo the use of IP, relying on direct raw data links or simplified layer 3 network layers.
As such, the mechanism put forth for these types of connections is the use of a **Pre-shared key symmetric encryption of the data layer only**.
This implies that all these connections need to be set up ahead of time, which should not be a problem as this typically applies to components in relatively close proximity, typically L1 to L3 connections.
The actual encryption algorithms to be used in each data exchange mode will be further studied, including post-quantum (also known as quantum-resistant) cryptography, and evaluated for availability, usability and applicability.

5.2 Augmented Intelligence Components

5.2.1 Reasoning services

The reasoning services module will be implemented as the Knowledge Base service (KBS). The KBS is responsible for ontology population, semantic enrichment of sensor-based data as well as reasoning and inferencing at a higher level in real time. The KBS will consume data from streams, map, translate and integrate them into the relevant semantic vocabularies and populate the semantic models through a persistency layer.

The ARESIBO ontology as well as the populated instances will be stored in an RDF triplestore. GraphDB will be utilized for storing and querying purposes as an efficient and robust graph database that will perform semantic inferencing and real-time reasoning while offering geospatial features. The KBS will achieve that by applying SPARQL-based rules and running a set of queries against the graphs. The set of queries derive from the User's requirements and focus on discovering new facts and generating new relations among the existing information.

The KBS module will receive as input data from the visual recognition module and sensors and will integrate them in order to identify points of interest.

A high-level representation of the module is shown in Figure 34. The module consists of the following sub-components:

- The ARESIBO **ontology** that will semantically represent all notions of the projects and will be used to enrich the data with contextual information.
- The **graph database**, *GraphDB 9.1 FREE*, that will be hosting the ARESIBO ontology and act as the Knowledge Base of the system.
- A **populating tool**, written in *Java* and utilizing the *Jena* and *rdf4j* frameworks, that will integrate information from multiple heterogeneous sources and semantically represent them in the Knowledge base
- The **reasoning service** that will be discovering events and facts of interest by running SPARQL queries on the graph database.

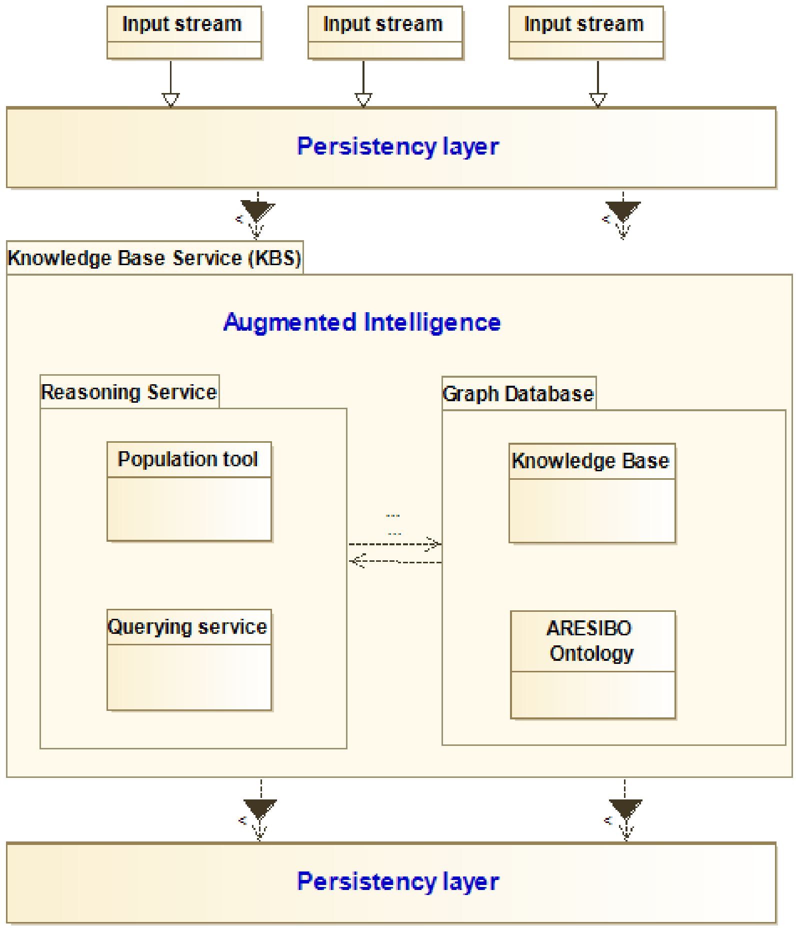

Figure 34: High-level representation of the KBS

Component name	**Knowledge Base Service (KBS)**
Responsible partner	Centre for Research & Technology Hellas (CERTH)
Provided functionalities	The KBS will be a back-end service responsible for inferring logical consequences and deriving knowledge from existing facts and data. The main objective is to semantically integrate information from various sources and generate notifications according to the derived events of interest.
Technical Description	The KBS component uses the ARESIBO ontology to semantically represent all of the project-related concepts and a tool written in Java to populate the knowledge base with

	the ontology instances and their relations. A graph database will host the ontology and the knowledge base, and will be the core of the reasoning functionality by answering SPARQL-based queries inferring new knowledge and reporting events of interest.
Expected input	Information from visual recognition module and details for detections, Cyber-security component and possibly numerical data from sensors and operational assets' metadata.
Expected output	Responses from Select/Update queries on the Knowledge Base in the form of alerts that will include details about the inferred knowledge, detections and incidents.
When it is used	The Knowledge Base service is continuously running and processing information when they are available from the relevant modules.
Relation to other components	Consumes information from recognition modules, Cyber-security module and potentially other sensors. Produces information and data for the highest integration level.
Candidate technologies to be used	*TopBraid Composer*: Development environment for authoring the ontology *Eclipse IDE*: Development environment for implementing the knowledge base population and querying *GraphDB 9.1 FREE*: RDF triplestore that will host the KB *RDF4J/Jena*: Frameworks for RDF data manipulation, inferencing and querying

	SPARQL 1.1: Query language for RDF *SHACL*: Language for RDF graph validation
Relevant Tasks	Semantic Representation and Reasoning (T4.1)
Comments	N/A

5.2.2 DSL-based Mission editor

Component name	**DSL-based Mission editor**
Responsible partner	National and Kapodistrian University of Athens (NKUA)
Provided functionalities	The MDL Editor will be a set of web tools for the definition and generation of robotic missions in ARESIBO. The main objective is to provide means to the end-users and especially C2 operators to easily specify robotic missions based on UxVs and relevant assets (e.g., sensors).
Technical Description	It is a web tool. The tool consists of a graphical and a textual editor both synchronized to be mutually consistent i.e., the user may use both of them according to her needs. The editors come along with a set of supporting editing tools that will facilitate the editing phase of a mission such as the validation of editing actions (e.g., selection of appropriate resources to be part of the mission).
Expected input	Information about resources (e.g., unmanned systems, available sensors, …) that will be part of a mission, coordinates to be followed by each resource in certain

	timesteps of the missions, actions to be taken (e.g., get measurements, activate/de-activate network interface).
Expected output	- A set of guidelines and instructions that will be deployed to unmanned systems as well as to other components of the ARESIBO architecture. - Data Model overlay object; - Data Model tasking object.
When it is used	The Mission editor will be used each time a new mission is needed by the end-users. In static environments where the missions are pre-defined (e.g., border surveillance of predefined areas) it is expected to be used only a few times. However, in dynamic environments the tool may be used to adapt existing missions according to new parameters or even generate completely new missions.
Relation to other components	Consumes information from UxVs and sensors. Produces output for the Resource Manager and the ARESIBO Dashboard.
Candidate technologies to be used	Java, Javascript/HTML, Eclipse Modeling Framework, X-TEXT
Relevant Tasks	Task 4.2 DSL-based specification of autonomous robotic missions
Comments	The status of each resource should be known in real-time when the mission is edited (e.g., AVAILABLE / ALLOCATED / NOT_AVAILABLE).

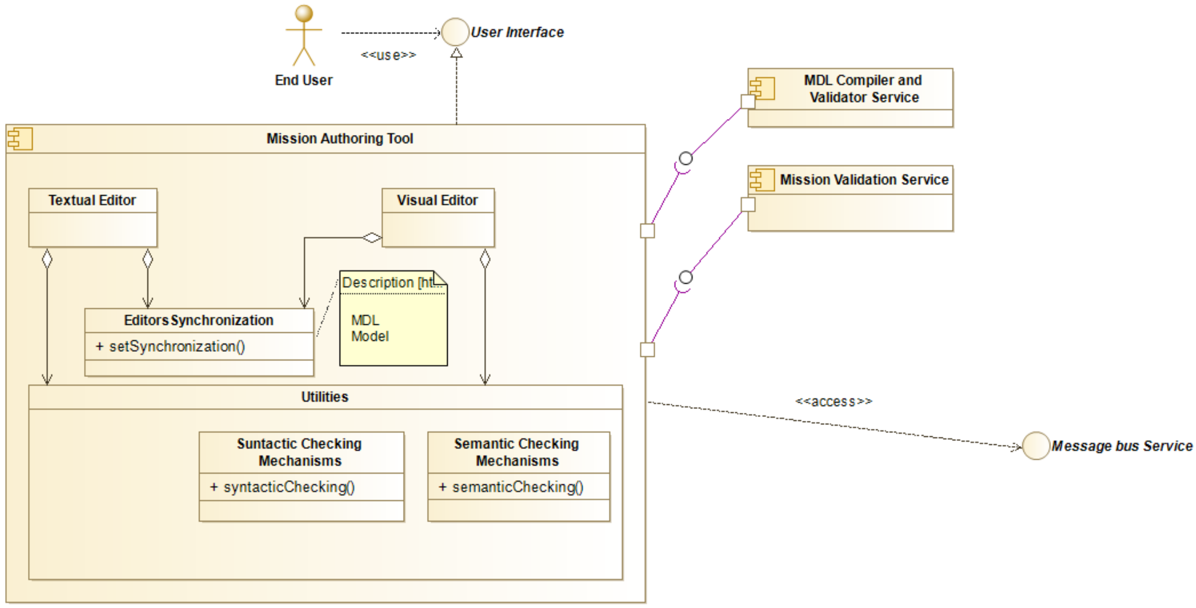

Figure 35: Package diagram for ARESIBO DSL-based Mission editor

The user interface of the ARESIBO Mission Authoring tool is separated in two areas, the Visual Editor and the Textual Editor. The user can interact with both of them to create missions. The Textual Editor is area where the user can interact with the ARESIBO MDL and the Visual Editor is the area where the user can create mission graphically. The user can synchronize the information between the editors.

The editor offers a set of utilities such as Syntactical Checking and Grammatical Checking of the entered information.

When the user enters information for the creation of a mission, this info is Validated, Compiled and as long the info / data is valid it gets dispatched to the ARESIBO Message Bus.

5.2.3 Decision support system

Component name	**Decision Support System**
Responsible partner	IES Solutions
Provided functionalities	The DSS will provide, only at Control Room level and not on the field, a set of functions to support decision making. Functions foreseen are based on calculations of current

	conditions and will output simple messages to be displayed on ARESIBO AR Tablets. It will not be a full-fledged DSS (i.e. it will not provide suggestions for all the resources managed in a control room) but a DSS targeted at suggesting which ARESIBO provided asset/resource may be useful to be deployed in a particular scenario. It should take into account all the info coming from the various ARESIBO assets (i.e. vehicles, drones with the capabilities of producing certain type of info that can be relevant for the specific event happening) and suggest the deployment of a specific ARESIBO asset for the ongoing mission. DSS will need information from sensors (i.e. Sensor Fusion) provided by ARESIBO, from Drones metadata and background information from C2
Technical Description	It is a set of web services to perform dynamic calculation for evaluation of the situation. It may create messages that will be displayed mainly on ARESIBO Glasses or AR Tablets meant to be used in Control Rooms. It may have a visual component for direct interaction but it's main expected use is to be used as a service.
Expected input	Information about ARESIBO-provided resources (e.g., unmanned systems, available sensors, …) that will be part of a mission, including the status (metadata) of that particular resource (i.e. available or potential remaining flight time for a drone). Information about all sensors, not only in terms of acquired data but also in terms of sensor metadata and capabilities. Most input should arrive from Risk analysis, Sensor Fusion and C2 system.

Expected output	- Clear messages for glasses of commanders or AR tablet in control room containing tips on new sensor to mobilize according to the current ongoing mission - Messages will be implemented by commanders or discarded by them. An example of a suggestion may be: "Send THAT SPECIFIC ARESIBO teams equipped with AR Glasses with a Rubber Hull inflatable boat (RHIB) to a location (expressed as text codes, such as Google Open Location Code or What 3 words codes) because this will give you additional information on the motorboat that is approaching.
When it is used	The DST will be used each time a new valuable asset is available and could be mobilized within a mission by the end-users. In static environments where the missions are pre-defined (e.g., border surveillance of predefined areas) it is expected to be used constantly.
Relation to other components	Consumes information from UxVs, sensors fusion and risk analysis. Produces output for C2 and the ARESIBO Dashboard.
Candidate technologies to be used	Java, Javascript/HTML, Eclipse Modeling Framework, PHP, MySQL, MVC Frameworks such as CakePHP.
Relevant Tasks	Task 4.4 DSL-based specification of autonomous robotic missions
Comments	The status of each resource and all available metadata should be known in real-time.

5.2.4 Simulation environment

Component name	**Simulation Engine**
Responsible partner	NATO STO Center of Maritime Research and Experimentation
Provided functionalities	The Simulation Engine is in charge of providing simulated behaviours for the assets, sensors and threats that are part of the ARESIBO ecosystem. Moreover, the simulator will provide the most relevant environmental information needed to make the simulation as realistic as required. The simulated data produced by the simulator will be used by the serious game engine during the training session.
Technical Description	The Simulation Engine is a set of distributed interoperable simulators based on the High-Level Architecture standard (STANAG 4603). The Simulation Engine comes with very limited graphical interface since it can be considered as the back end of the serious game engine. Each simulator needs to be configured through a set of JSON configuration files that can be generated manually or using the graphical tool provided by the Serious Game Engine.
Expected input	• Geographical location and extension of the simulated area • Weather and environmental conditions • List of vehicles and their sensors • Vehicle and sensor performance • List of threats and their behaviors • Mission to be followed
Expected output	• Vehicle telemetry: position (latitude, longitude, altitude), attitude (roll, pitch and yaw angles), speeds. The

	telemetry can be "real" or "estimated", i.e., affected by the sensors' navigation error. • Sensor data: status of the sensor (on/off/working/not working), target in sight, detections; • Environmental data: weather conditions (sun, rain, fog, cloud, humidity, air temperature, water temperature, water salinity, waves, water turbidity, etc.
When it is used	The Simulation Engine will be used during the training sessions of ARESIBO operators as a generator of simulated input and feedback.
Relation to other components	From a logical and a conceptual point of view the data produced by the simulation engine will be consumed by all the ARESIBO modules that, in the real ARESIBO deployment, will consume data coming from vehicles and sensors. However, from a technical point of view the simulation engine will share information only with the serious game engine that will act like a software bridge between the actual ARESIBO system and the Simulation Engine.
Candidate technologies to be used	The technologies used are C++, Java, C#.
Relevant Tasks	Task 4.3 Simulation Environment for border security operations.
Comments	N/A

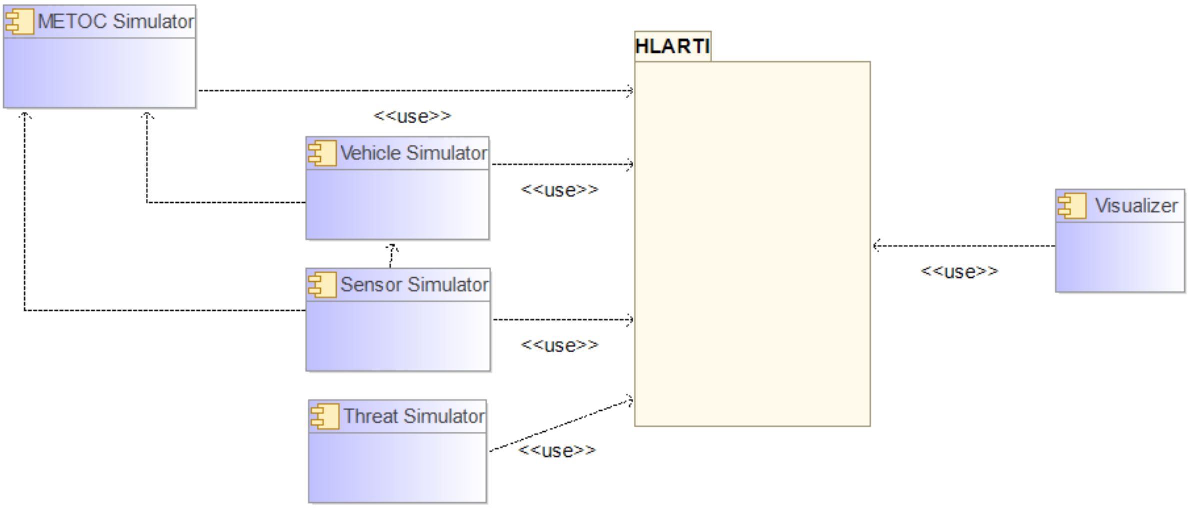

Figure 36: Package Diagram for ARESIBO Simulation Engine

Package diagram for the Simulation Engine showing the main components (simulators) connected via the HLA RTI (High Level Architecture Run-Time Infrastructure). Note that the Sensor Simulator is dependent on the Vehicle Simulator and both are dependent on the Environmental (METOC) Simulator.

5.2.5 Sensor Fusion engine

Component name	**Sensor Fusion engine**
Responsible partner	National and Kapodistrian University of Athens (NKUA)
Provided functionalities	The Fusion Box is a versatile middleware platform, that allows the treatment of incoming data flows, coming from various sensors, through complex, yet fully controllable data processing workflows.
Technical Description	It is a web-based tool. It allows the domain (and not the IT) expert to easily specify the needed processing, through a Domain Specific Language and automatically transform such specification to executable workflows.

Expected input	Different types of data sources, coming from all the available sensors, and a data processing scenario in a Domain Specific Language.
Expected output	- The detection of pre-defined events. - The decision or inference regarding the characteristics of an observed entity - An interpretation of the observed entity in the context of its surroundings and relationships to other entities
When it is used	The Fusion Box will be used each time the end user needs to deploy a mechanism in order to detect an event and/or process data from heterogenous sources.
Relation to other components	Consumes information from sensors. Produces output for the ARESIBO Dashboard.
Candidate technologies to be used	Java, Javascript/HTML, PostgreSQL
Relevant Tasks	Task 4.5 Real-time sensor fusion and interoperable provision of sensor data
Comments	N/A

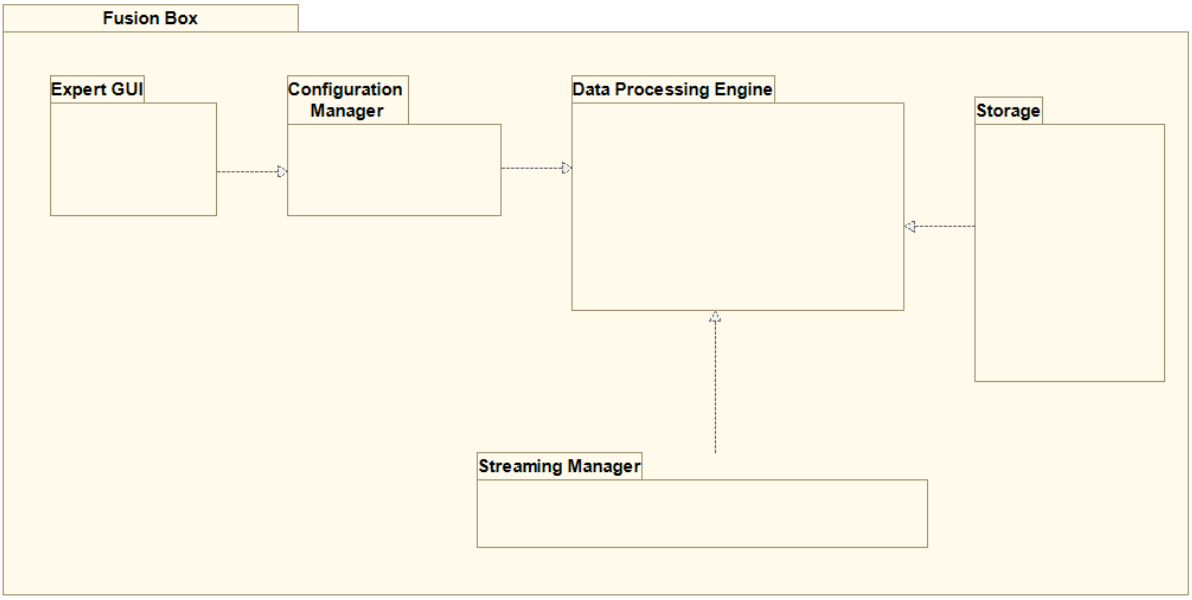

Figure 37: Package diagram for ARESIBO Sensor Fusion Engine

The Fusion Box Expert GUI is an easy to handle management tool targeting the fusion-box end user. It is web based and provides functionalities for monitoring of the input data sources, deploying processing scenarios to the Data Processing Engine, through Configuration Manager and monitoring deployed applications.

The Data Processing Engine compromises the application layer of the Fusion Box architecture, which provides the appropriate runtime environment for all processing scenarios. It receives input from the Streaming Manager, which allows the selection of specific sources based on specific characteristic of the source and uses persistent storage to store intermediate results and configuration settings.

5.2.6 Risk analysis module

Component name	**Risk Analysis Module**
Responsible partner	National and Kapodistrian University of Athens (NKUA)
Provided functionalities	The Risk Analysis Module engine will be responsible for analysing the current situational picture by predicting possible future states, such as the coordinates of a moving hostile entity. The predicted future state will be correlated

	with CIRAM's threat function and provide alerts for possible threats as they develop over time.
Technical Description	The Risk Analysis Module consists of a processing engine enhanced with a set of predictive algorithms responsible for predicting future states of a hostile entity. It is coupled with a graphic user interface that supports directed graphs of processors capable of ingesting data from external systems, compute and export data to other systems.
Expected input	The module is expected to receive constant and as frequent as possible (<=1 sec) telemetry data collected from a tracked enemy entity such as current position (GPS), speed and bearing in order to successfully predict future positions. Minor configuration and tuning might be need from the operator to successfully track the entity and define the threat scenario.
Expected output	A threat alert issued with a future timestamp that is expected to come true sent out from the Risk Analysis Module to the interested recipients. CIRAM terminology is used to label tactical and operational risks.
When it is used	The Risk Analysis Module will be used whenever possible to determine the future state of hostile entities tracked in order to intercept or define a mitigation plan.
Relation to other components	Consumes information from radars and sensors. Produces output for the ARESIBO Dashboard.
Candidate technologies to be used	Java, Python, Apache NiFi, Django, PostgreSQL, GEOS Api.

Relevant Tasks	Task 4.6 Risk analysis for integrated situation awareness
Comments	The quality of the predicted outcomes depends on the frequency and telemetry data collected of the tracked entity.

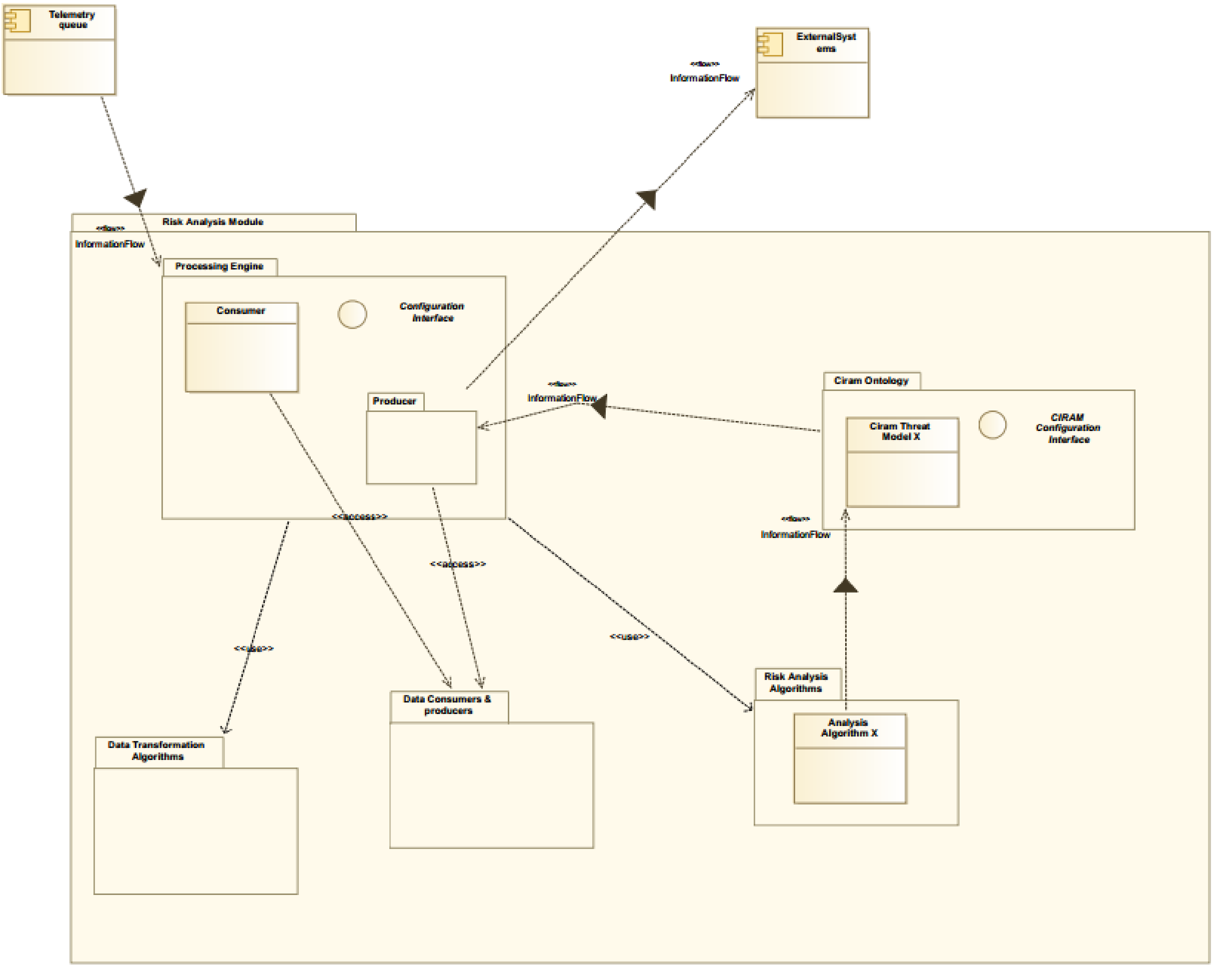

Figure 38: Package diagram for ARESIBO Risk Analysis Module

The Risk Analysis Module consumes data from incoming queues, using telemetry data from the observed hostile entity. The operator has a variety of consumers to choose from in order to ingest the data to the risk analysis module (i.e. mqtt, kafka consumers etc). After the data can be ingested to the module, the operator can select and interconnect appropriate data transformation algorithms that are able to transform the data so that they are compatible with the selected risk analysis algorithm. The risk analysis algorithm will calculate a future state of the observed hostile entity and forward the output to the CIRAM Ontology. The operator can select points of interest in a map where crossing them or even getting close to them would constitute a high threat situation. Based on the proximity of the hostile object on a future time as predicted an alert is issued with a threat factor that is being sent to the message bus/queue and the interested components can consume for visualization or in order to notify officers.

5.3 Augmented Reality Components

5.3.1 Data communication gateway for AR functionality

Component name	**Data communication gateway for AR functionality**
Responsible partner	UBIMAX, VTT, ADS
Provided functionalities	Encoding and decoding of data to be communicated by the sub clients and ARESIBO components
Technical Description	An interface that converts Kafka to MQTT and vice versa based on the application requirement.
Expected input	Relevant information from all the end users or ARESIBO system (C2 and Tactical Commander components or data from Sensor Fusion/DCS)
Expected output	Relevant information from all the end users or ARESIBO system (eg. url for the video/voice stream, location of the Field Officer)
When it is used	It is used when a message is requested or send from the application using KAFKA protocol or application using MQTT protocol.
Relation to other components	Consumes the input from MQTT application and converts to Kafka format and vice versa.

Candidate technologies to be used	AVRO, KAFKFA, MQTT, JSON
Relevant Tasks	T5.1 Interoperable AR Data Interface Framework T5.2 AR Functionalities, Interfaces and Tools on Field Operations T5.3 AR Functionalities, Interfaces and Tools on the C2
Comments	N/A

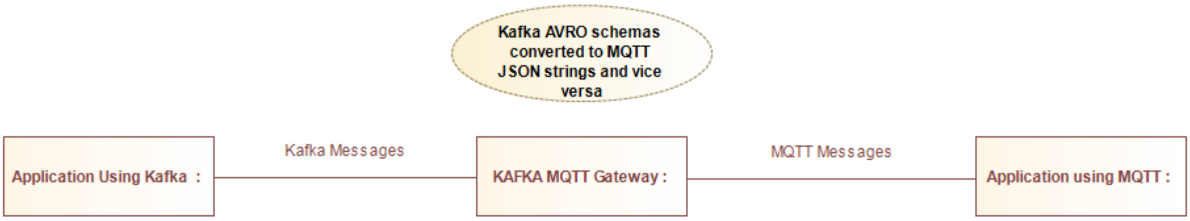

Figure 39: Diagram for ARESIBO Data communication gateway

5.3.2 Field Officer AR component

Component name	**Field Officer AR component**
Responsible partner	UBIMAX
Provided functionalities	The AR Component will include the transformation, visualization and user interaction of relevant data for Field Officer that is available in the ARESIBO system. The main objective is to develop a platform that displays the data on monocular devices for the Field Officers.
Technical Description	A software that can be used on the monocular devices to interact with the data available on the ARESIBO system. The

	component primarily includes the software and a java interface that interacts with other end user components (C2, Tactical Commander) through MQTT messages.
Expected input	Relevant information from other end users or ARESIBO system (C2 and Tactical Commander components or data from Sensor Fusion/DCS)
Expected output	Relevant information from Field Officers that is necessary to other end users or ARESIBO system (e.g., URL for the video/voice stream, location of the Field Officer)
When it is used	The component will be activated when a Field Officer has an active mission. They will be equipped with monocular devices which will serve as the medium of interaction during the mission.
Relation to other components	Consumes information from other ARESIBO components through the Kafka Message Bus. Produces output for the Kafka Message Bus.
Candidate technologies to be used	Java, XML, JSON, AVRO
Relevant Tasks	T5.2 AR Functionalities, Interfaces and Tools on Field Operations
Comments	N/A

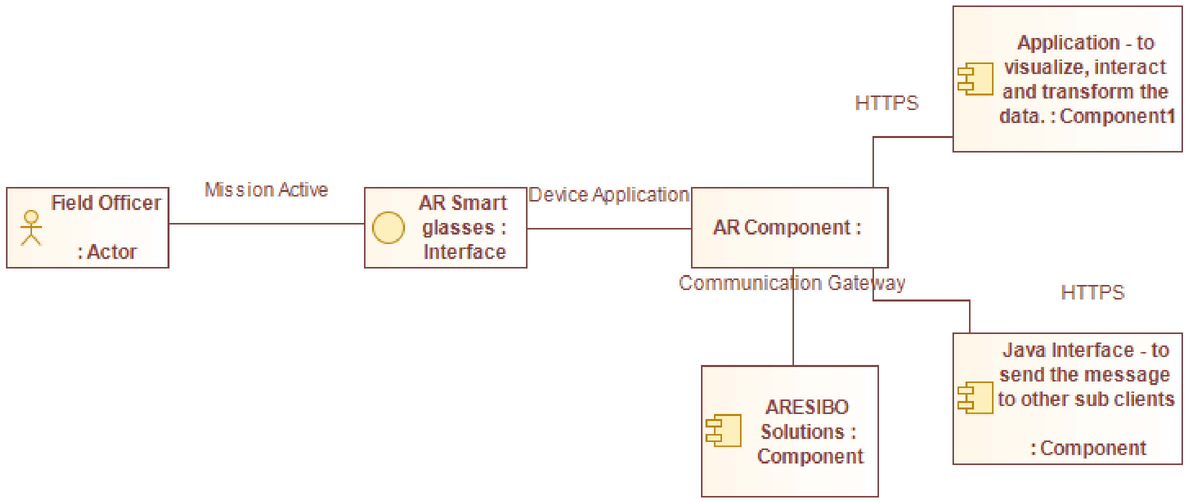

Figure 40: Package Diagram for ARESIBO Field Officer AR component

5.3.3 Tactical Commander AR components

Component name	Tactical Commander AR components
Responsible partner	VTT Technical Research Centre of Finland Ltd
Provided functionalities	Component will allow user (Tactical commander) to receive alerts, access camera streams and communicate with the Field Officer and C2
Technical Description	A Unity application, which uses Kafka message bus. User operates map in AR mode, reacting to the alerts and monitoring incident investigation progress by giving commands and accessing camera video streams
Expected input	Information about the incident (coordinate and camera stream), UxV sensors and telemetry data
Expected output	Command (destination coordinate, text message) and information about the incident (text message and camera stream)

When it is used	When Tactical Commander is required to assist in the incident investigation
Relation to other components	Consumes information from: - ARESIBO server - Risk module
Candidate technologies to be used	Unity3D, Hololens 2, ARCore, ARKit, Vuforia
Relevant Tasks	Task 5.3 AR functionalities, interfaces and tools
Comments	-N/A

5.3.3.1 Tactical Commander AR component on tablet/smartphone

Tablet or smartphone version of Tactical Commander AR component user interface consists of an AR UI object and additional 2D UI elements (as shown in Figure 41). User is able to see alerts and sensor data visualization, including UxV objects, their paths, and waypoints as pins on the map.

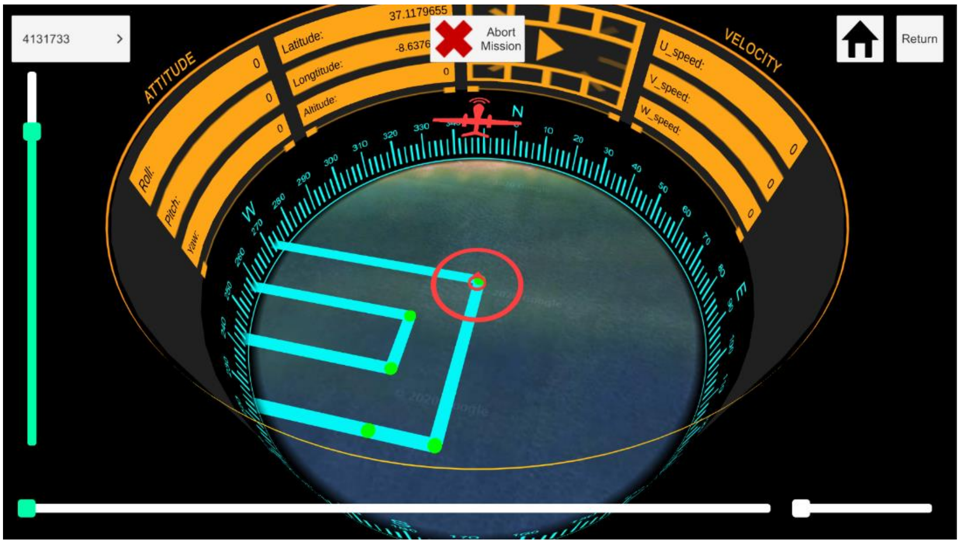

Figure 41: The tablet version of Tactical Commander AR component

User can select a UxV object on the map to read the telemetry data. Telemetry display includes video stream from UxV camera.
Additional sliders in tablet/smartphone version are used for controlling the map (zoom, scale, and rotation). Map also can be manipulated directly by using touch screen.

5.3.3.2 Tactical Commander AR component on binocular devices

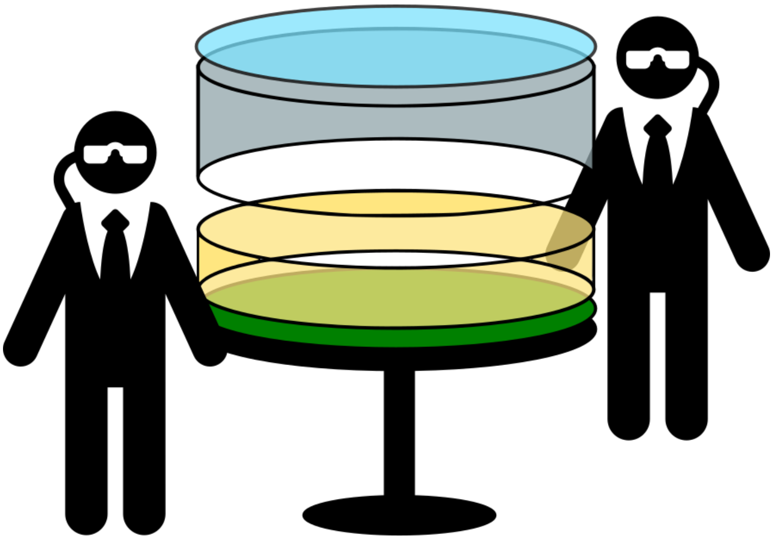

Figure 42: Tactical Commander AR component in multi-user tabletop mode

Binocular devices (like Microsoft HoloLens 2) are using the same version of AR object as the tablet variant of the system. Due to the spatial nature of the AR technology, design of the AR object is fully utilized on the binocular devices. The map and information display are the same as in the tablet version, but the additional 2D controls (zoom, scale, rotate) are replaced by HoloLens-specific ones.

5.3.4 C2 Operator AR component

5.3.5 Time-based visualisation component

Component name	**Time-based visualisation**
Responsible partner	VTT Technical Research Centre of Finland Ltd
Provided functionalities	Component will allow user (C2 personnel, Tactical commander and Field officer) see situation in different timeline. To past based on collected information and to future based on simulation.
Technical Description	It is Unity application, which request information via Kafka message bus. User could use time bar to go backwards and/or forward on situation. Situation on the AR tool will be updated for selected time step.
Expected input	Information of selected resources (e.g., unmanned systems, available sensors, …) with time stamp.
Expected output	Time-based situation on AR application
When it is used	Time-based visualisation will be used to get better situation awareness. User could check what has been happened and what will be estimation of future situation
Relation to other components	Consumes information from: - ARESIBO server - Risk module - Serious Game

Candidate technologies to be used	Unity3D
Relevant Tasks	Task 5.4 Time-based visualisation
Comments	N/A

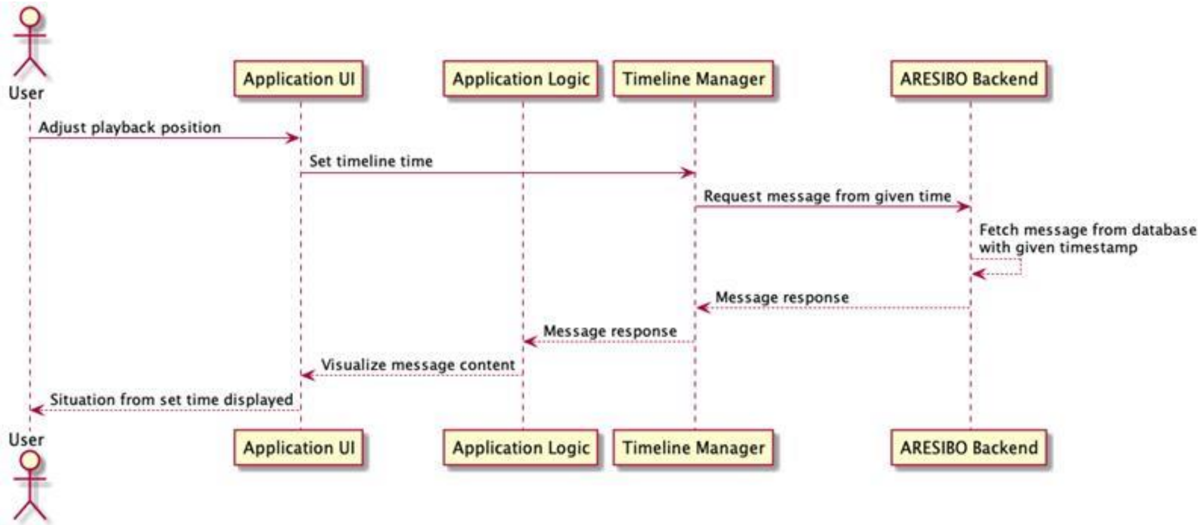

Figure 43: Diagram for ARESIBO Time-based visualisation

5.3.6 ARESIBO Serious game

Component name	**Serious Game Engine**
Responsible partner	NATO STO Center of Maritime Research and Experimentation
Provided functionalities	The Serious Game Engine is a software tool that will provide a user-friendly and GUI based interface to setup, run, supervise and analyse the results of a serious game-based training session on the ARESIBO system.

Technical Description	The Serious Game Engine will allow the technical interoperability between the Simulation Engine and the ARESIBO system. It will consist of three main modules: • A map based Graphical User interface used by the trainers to setup and supervise the training session • A High-Level Architecture STANG 4603 compliant module to connect the trainers GUI to the Simulation Engine • A Bridge Module to allow interoperability between the Simulation engine and the ARESIBO system
Expected input	The inputs expected before the training starts are: the training configuration (type of trainees, number of trainees etc..) and the training scenarios (land border scenario, blue border scenario etc.) The inputs during the training session are: mission, alerts and commands
Expected output	Telemetry, sensor data and video stream and data to compute training KPI still to be defined.
When it is used	The Serious Game Engine will be used during the training session of the ARESIBO system.
Relation to other components	The Serious Game Engine will be the bridge between the Simulation Engine and the ARESIBO system.
Candidate technologies to be used	The candidate technologies are: C++, Qt Library, Java
Relevant Tasks	Task 5.5 Serious gaming for users training

Comments	N/A

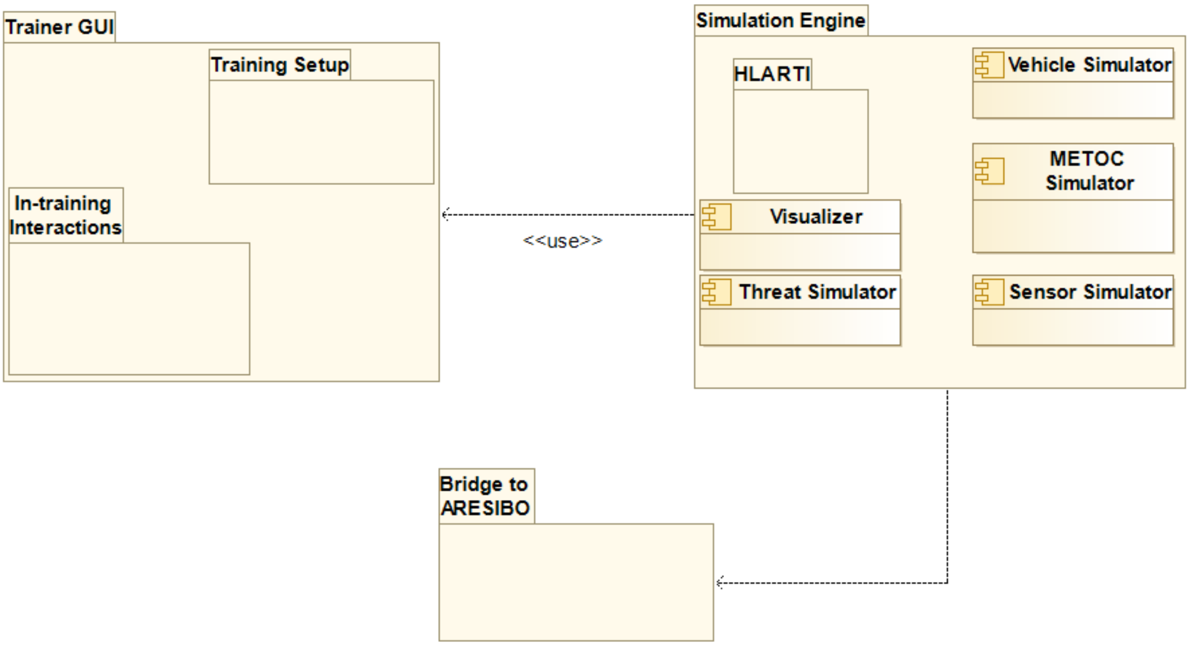

Figure 44: Package diagram for ARESIBO Serious Game Engine

The package diagram for the Serious Game, which contains the Simulation Engine (described in 4.2.4) as well as the GUI for the trainer and a bridge to the ARESIBO system.

5.4 Integration Components

5.4.1 Communication message bus

Component name	**Message Bus**
Responsible partner	National and Kapodistrian University of Athens (NKUA)
Provided functionalities	The Message Bus will be responsible for quick and secure message delivery between all the underlying components. Message format will be structured to avoid conflicts and incompatibility between different components and services. The message hub will have high availability and distributed architecture to prevent data loss.

Technical Description	The Message Bus will be consisting of several brokers located in different servers accompanied by schema servers that will work as a reference for the data format. To ensure synchronization and leader election between the servers a cluster coordinator will be used. The input and output of the messages will be performed by clients following the pub/sub paradigm.
Expected input	Structured messages containing information and data spanning for different uses from services functionality, data analysis, and data transformation to simple platform messages such as logging information, health status messages, and administrative commands.
Expected output	(same as input) Structured messages containing information and data spanning for different uses from services functionality, data analysis, and data transformation to simple platform messages such as logging information, health status messages, and administrative commands.
When it is used	The Message Bus will be used every time another component needs to send or receive information acting as input or output and will provide a fast and secure data delivery.
Relation to other components	The Message Bus component may interact with a multitude of different components, which are involved in communication actions across distinct application tiers. Message bus provides a loosely coupling mechanism for interaction and therefore the notion of a provided or requested interface does not strictly apply.

Candidate technologies to be used	Apache Kafka
Relevant Tasks	Task 6.2 – Software and hardware integration at component-level Task 6.3 – Full-system integration, customisation and operational testing
Comments	N/A

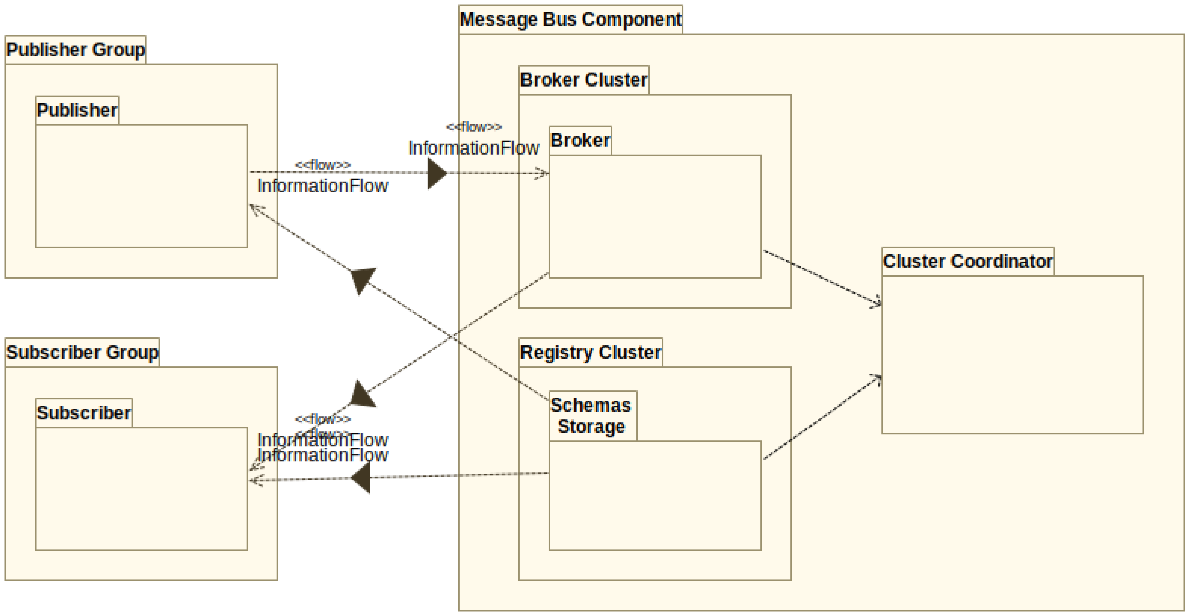

Figure 45: Package diagram for ARESIBO Message Bus

Message Bus will interact with the other component via clients following the Publish/Subscribe protocol. Publisher groups will consist of several publishers depending on the number of instances of a specific component and will deliver the messages to the Message Bus. The same will happen with the Subscriber Groups but instead of delivering the messages they will receive them. The Message Bus will consist of a message broker cluster which it's the main function will be the delivery of messages. The coordination and leader election will be performed from the cluster coordinator. Finally, the format of the messages will be handled from the Registry Cluster.

5.4.2 Data storage

Component name	**Data storage**
Responsible partner	National and Kapodistrian University of Athens (NKUA)
Provided functionalities	Data Storage will enable secure data preservation from the message bus. This will let the user browse available data sources for subject to analytical treatment as well as previous analysis tasks outcomes. Through the tool, data learning tasks can then be initiated and carried out for as long as needed depending on the nature of the analysis to be performed or the type of the data being analyzed.
Technical Description	The Data Storage will allow the user to access data sets stored in the different data repositories. By having access to the available data, the user can browse the available data and select the tables/data structures on which he wants his subroutines to be executed. Through it the user not only gains access to the data coming from the data source/sources, but also enables access and selection of results from previously executed jobs, which can afterwards be involved in other data analysis jobs.
Expected input	Structured messages containing information and data spanning for different uses from services functionality, data analysis, and data transformation to simple platform messages such as logging information, health status messages, and administrative commands.
Expected output	Structured messages containing information and data spanning for different uses from services functionality, data analysis, and data transformation to simple platform

	messages such as logging information, health status messages, and administrative commands.
When it is used	Data Storage will be used every time another component need to send or receive information acting as input or output through the Message Bus.
Relation to other components	Data Storage will consume all the messages from the Message Bus.
Candidate technologies to be used	Apache Hadoop, PostgreSQL
Relevant Tasks	Task 6.2 – Software and hardware integration at component-level Task 6.3 – Full-system integration, customisation and operational testing
Comments	N/A

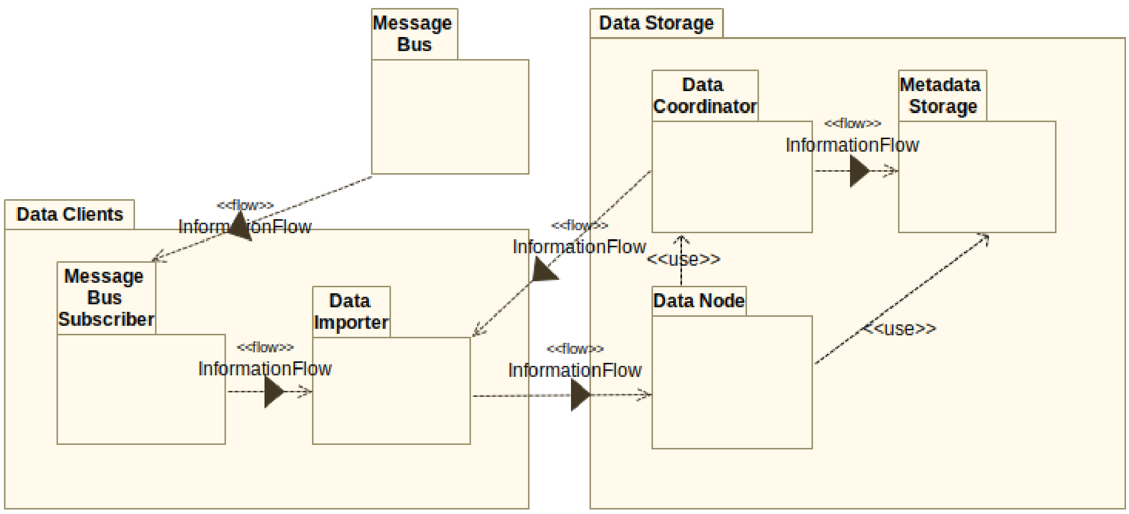

Figure 46: Package diagram for ARESIBO Data Storage

Data Storage will receive the messages from the Message Bus component through a middle layer of clients that will use the Message Bus API of subscribers to receive the messages and then will forward via message transformation to the Data Storage through dedicated data importers. Data Coordinator will be used to provide info and store metadata regarding the data consistency among the several data nodes of the Data Storage cluster.

6 Preliminary data integration

In this section we provide preliminary results from the data integration approach that was followed by the project in the first period. Based on (i) the outcomes of D4.1 and, in particular, the identified data types, and (ii) the architecture of the system as it is reflected in the previous sections, a set of basic data interfaces has been specified to support the required data flows in the system. The specified messages are following a JSON serialisation format. The messages are exchanged through the Kafka middleware following the ARESIBO system architecture as it was described in the previous sections of this deliverable.

Table 1 summarises the basic data interfaces that have been identified in the first period of the project. As a summary, a set of 33 required messages has been identified. A connection between the datatypes and the messages is described along with certain implementation parameters that should be considered (time intervals for data availability, components that produce/consume messages within the system, relevant tasks, partner responsibilities for the provision of data, etc.). In addition, the last three columns of the table intend to summarise the feedback collected from the end-users with respect to the visualisation layers i.e., what types of data should be visualised in each one of the three visualisation layers of the project; AR C2 application, AR Tactical commander application, AR Field Officer application.

This table constitutes the first step towards the integration of all ARESIBO modules into the 1st integrated prototype of the platform. It is expected that this table will be further refined during the development and integration phases in the second year of the project. The refined versions of the table will be provided in the integration reports that will follow in years 2 and 3 of the project (D6.2, D6.3).

Table 1. Preliminary integration approach – Connection between ARESIBO data types and Kafka messages

Data integration										Visualisation Layer		
#	TopicName	Need to Consume (optional) - This optional field refers to the type of input data required from the producer in order to produce messages of this topic name	Producer	Consumer(s)	Description	Message Format (Connection to the ARESIBO Data Model and D4.1)	Time interval	Data types that will be made available through this topic	Partner	AR C2 operator (ADS)	AR Tactical Commander (VTT)	AR Field Officer (UBI)
1	***FusionAlert***	Sensor data, Telemetry Data	Sensor Fusion (T4.5)	Reasoning Service (T4.1), DSS (T4.4), C2 Dashboard (T6.3),	This topic is used for sharing alert data produced by the sensor fusion module.	ThreatAlert	asynchronous (event-based)	Alert based on sensor fusion	NKUA	YES	YES	YES
2	***RiskAlert***	Sensor data, Telemetry Data	Risk Analysis (T4.6)	Reasoning Service (T4.1), DSS (T4.4), C2 Dashboard (T6.3)	This topic is used for sharing alert data produced by the risk analysis module.	ThreatAlert	asynchronous (event-based)	Alert based on risk analysis	NKUA	YES	YES	YES
3	***MissionDescription***	N/A	Mission Editor (T4.2)	Resource Controller (T3.3), DSS (T4.4)	The mission that will be created by the operator through the Mission Editor should be made available to ResourceController. This is different to the MissionPlan that the UxVs will consume (which is the optimised output of the resource control module)	Mission	asynchronous (event-based)	UxV missions	NKUA	NO*	NO	NO

4	***MissionPlan***	MissionDescription (T3.3 and T3.6), in addition for T3.6 VisualDetection and TelemetryData	Resource Controller (T3.3), RT-SAM (T3.6)	CS_Robotnik (6.2), CS_Tekever (6.2), CS_MST (6.2), DSS (T4.4)	Mission Plan to be sent to the UxVs	Plan, Waypoint, additions/changes will probably be needed	asynchronous (event-based)	Set of waypoints per vehicle for a mission, sensors activated and selected detections	CERTH-ConvCAO	NO	NO	NO
5	***MissionVisualization***	MissionDescription	Resource Controller (T3.3), RT-SAM (T3.6)	C2 Dashboard (T6.3)	Mission to be visualised	Plan, Waypoint, additions/changes will probably be needed	asynchronous (event-based)	Paths for all vehicles, ROIs, activated sensors, selected detections etc	CERTH-ConvCAO	YES	YES	YES
6	***MissionPlanWarning***	MissionDescription	Resource Controller (T3.3)	C2 Dashboard (T6.3)	Warning message to be visualized in a pop-up window to inform the operator for some kind of problem during the path calculation	It should be added in the next version of the data model	asynchronous (event-based)	Warning message	CERTH-ConvCAO	YES*	YES*	YES*
7	***VisualDetections***	VideoStream, MissionStatus, TriggerDetection, TelemetryData	Visual Recognition (T3.6)	Swarm Optimiser T3.6, Reasoning Service T4.1	This topics is used to propagate the results of object detection	VideoDetection	asynchronous (event-based)	Bounding boxes, Labels, Confidence Scores	CERTH-MKLab	NO	NO	NO
8	***ReasoningAlerts***	VisualRecognition, FusionAlert, RiskAlert	Data representation (T4.1)	DSS (T4.4), C2 Dashboard (T6.3)	Responses from Select/Update queries on the Knowledge Base in the form of alerts that will include details about the inferred knowledge, detections and incidents.	AlertType	asynchronous (event-based)	Alert based on semantic fusion	CERTH-MKLab	YES	YES	NO
9	***MissionStatus***	N/A	CS_Robotnik (6.2), CS_Tekever (6.2), CS_MST (6.2)	DSS (T4.4)	Update on the status of the mission being performed by UxV	MissionStatus	synchronous (time interval configurable)	Notification message		YES	YES	NO

10	***WaterVehicles***	N/A	CS_MST	DSS (T4.4)	Estimated state and position of the vehicle. In the case of UUV this report can be configured between 60sec and 300sec. This report is made via an acoustic modem and has interference in the data collected by the side-scan. So it can't be very frequent	UnderwaterVehicleType	1min to 5min	Estimated vehicle status and position	MST	YES	YES	NO
11	***UGVStatus***	N/A	CS_Robotnik (6.2)	DSS (T4.4), C2 Dashboard (T6.3)	Current status of the vehicle (such as operational status, set of sensors, availability, etc) to be consumed by MissionEditor, Dashboard and other monitoring and commanding tools	GroundVehicleType	synchronous (time interval configurable)	UGV status	ROB	YES	YES	NO
12	***TelemetryData (KinematicPlatform)***	N/A	CS_Robotnik (6.2)	DSS (T4.4), C2 Dashboard (T6.3)	Current position and attitude, and linear and angular velocities of the UGV in a world reference frames	TelemetryData	synchronous (time interval configurable)	UGV velocity	ROB	YES	YES	NO
13	***MissionAbort***	N/A	C2 Dashboard (T6.3)	CS_Robotnik (6.2)	Abort current mission being executed by UGV	N/A	asynchronous (event-based)	Notification message	ROB	NO	NO	NO
14	***MissionAcceptance***	N/A	CS_Robotnik (6.2)	C2 Dashboard (T6.3)	Tells the ARESIBO system that the mission delivered has been accepted by the UGV pilot	N/A	asynchronous (event-based)	Notification message	ROB	YES	YES	NO
15	***SensorPayload***	N/A	CS_Robotnik (6.2)	C2 Dashboard (T6.3)	Current payload of sensors equipped in the UxV (videostream, audiostream, etc)	Sensors	synchronous (time interval configurable)	Sensor payload	ROB	NO	NO	NO

16	***TacticalCommander Order***	TelemetryData, MissionVisualization, ReasoningAlerts, UGVStatus, WaterVehicles	TacticalCommander (5.3)	C2 (5.3) and Field officer (5.2)	Order from TacticalCommander to the FieldOfficer, including point of interest (POI), one-line text command from the predefined list (e.g. investigate/abort/stream_on), and a text commentary	N/A	asynchronous (event-based)	Coordinate of POI, text command (e.g. investigate, abort, stream_on/off), text commentary	VTT	YES	NO	YES
17	***TacticalCommander Report***	MissionStatus, UGVStatus	TacticalCommander (5.3)	C2 (5.3) and Field officer (5.2)	Report from TacticalCommander to the C2, including current GPS position of the TacticalCommander, one-line text report from the predefined list (e.g. mission_confirmed), and a text commentary	N/A	asynchronous (event-based)	Coordinate of TacticalCommander, text report (e.g. ready, busy, away), text commentary	VTT	YES	NO	YES
18	***TacticalCommander Stream***	SensorPayload	TacticalCommander (5.3)	Field officer (5.2)	Link to the selected video stream from the UxV or stationary camera to the FieldOfficer and a text commentary	N/A	asynchronous (event-based)	Link to the stream from UXV/camera to the FieldOfficer, text commentary, camera position and rotation if available	VTT	NO	NO	YES
19	***Suggestions***	N/A	DSS (T4.4)	C2 Dashboard (T6.3), XR visualisation (T5.1-4)	Distributes suggestions about which resource to deploy on the base of alerts and current resources status.	Suggestions (Not available in the first version of D4.1, please refer to D4.4)	asynchronous (event-based)	Suggestions produced by DSS	IES	YES	YES	NO
20	***C2SuggestionsFeedbacks***	N/A	C2 Dashboard (T6.3)	DSS (T4.4)	Allows to accept or discarge a suggestion previously received by C2.	Suggestions (Not available in the first version of D4.1, please refer to D4.4)	asynchronous (event-based)	Feedbacks provided by C2	IES	NO	NO	NO

21	***XRSuggestionsFeedbacks***	N/A	XR visualisation (T5.1-4)	DSS (T4.4)	Allows to accept or discarge a suggestion previously received by XR.	Suggestions (Not available in the first version of D4.1, please refer to D4.4)	asynchronous (event-based)	Feedbacks provided by XR	IES	NO	NO	NO
22	***UAV Platform Status***	Telemetry data	CS_Tekever	DSS	Current status of the vehicle (ID, position, attitude, velocity, wind speed, endurance, mission ID, mode, current wp, available sensors)	Telemetry Type	every second	Vehicle status, position and environemental variables	TEK	YES	YES	YES
23	***UAV EO Payload Status***	Telemetry data	CS_Tekever	DSS	Status of the EO Payload (selected camera, camera angles, zoom status, FOV, target position)	Telemetry Type	Every second	Payload status	TEK	YES	NO	NO
24	***UAV POI***	Mission Data	CS_Tekever	DSS	Point of interest being investigated	N/A	asynchronous (event-based)	location coordinates	TEK	YES	YES	NO
25	***UAV AOI***	Mission Data	CS_Tekever	DSS	Area of interest being investigated	N/A	asynchronous (event-based)	Set of location coordinates	TEK	YES	YES	NO
26	***UAV AIS***	Sensor Data	CS_Tekever	DSS	AIS signal received	N/A	asynchronous (event-based)	AIS UAV system has received (coordinates, MMSI, date, direction, vessel status)	TEK	YES	YES	NO
27	***UAV Detection***	Sensor Data	CS_Tekever	DSS	Automatic detection	N/A	asynchronous (event-based)	GPS coordinates and classification	TEK	YES	YES	YES
28	***UAV Sensor Payload***	Sensor Data	CS_Tekever	DSS	Video streams	N/A	asynchronous (event-based)	Video streams	TEK	YES	YES	YES
29	***Field Officer Mission Acceptance***	Mission Data, Mission Description	Field officer (5.2)		Accepting mission	Field Officer Data	asynchronous (event-based)		UBI	YES	YES	NO

30	***Field Officer Mission Data***	Misson Data	Field officer (5.2)	TacticalCommander (5.3), C2 (5.3)	Location to be investigated; Mission Objectives	Field Officer Data	Every 30 S	Field Officer Position	UBI	YES	YES	YES
31	***Field Officer Feedback***	N/A	Field officer (5.2)	TacticalCommander (5.3), C2 (5.3)	Feedback Collected during mission (Speech to text, Media, Location)	Field Officer Data	asynchronous (event-based)	Image, Position, Text	UBI	YES	YES	NO
32	***Field Officer Stream Link***	Stream Link Request	Field officer (5.2)	TacticalCommander (5.3), C2 (5.3)	Enabling stream link	Field Officer Data	asynchronous (event-based)	URL, Position	UBI	YES	YES	NO
33	***C2Order***	TelemetryData, MissionVisualization, ReasoningAlerts, UGVStatus, WaterVehicles	C2(5.3)	TacticalCommander (5.3)	Order from C2 toTacticalCommander, including point of interest (POI), one-line text command from the predefined list (e.g. investigate/abort), and a text commentary	N/A	asynchronous (event-based)	Coordinate of POI, text command (e.g. investigate, abort), text commentary	ADS	YES	YES	NO

7 Conclusions

This document describes the design of the ARESIBO system and intends to pave the way of the upcoming development and integration steps that will follow. The main advancements achieved in this deliverable are the following:

- A quick review of the existing architectures is provided along with the potential starting points for the ARESIBO developments;
- The project is positioned within the existing border security area of EU-funded projects by identifying unique features and advancements made in the project;
- Different views of the ARESIBO system are specified focusing on logical architecture, data architecture and physical topology;
- The different modules of the ARESIBO system are identified and a preliminary design of them is given (functional features, interconnections, interfaces with other modules, technologies to be used for the implementation, etc.);
- The ARESIBO modules are categorized on the three technology pillars considered in the project; Augmented Reality, Augmented Intelligence, Augmented Communication and Sensing;
- Based on the specified architecture, a set of preliminary data interfaces is presented that can prove as the starting point for the development and integration process;

The validation of the architecture will be carried out through a series of integration tests, field trials and full system deployments in real-life operations within the second and third year of the project.

8 Acknowledgment

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833805.

Annex A: Technological base and background for ARESIBO developments

This annex briefly indicates the baseline for the development activities of the ARESIBO project. For the different technical component of the project the following two items are briefly explained: (a) the starting point for the ARESIBO development is specified and (ii), on top of that, the expected technological advancements the project will bring are given.

Network solutions for border security. Viasat as a satellite ISP has wide knowledge and experience in fix and mobile communication systems. At Viasat Switzerland we are exploring new business areas related to connectivity systems for ground transportation markets where we need to develop antennas and networking systems that can assure coverage "anywhere, anytime and while the vehicle is on the move". Between these business cases we can find ARESIBO system where we are engaged to develop a mobile communication hub as part of the global solution.

Unmanned platforms. Robotnik has several platforms capable of performing outdoor navigation. They are being tested in the ROBORDER project [1]. The expected technological advances are headed towards improving the autonomous outdoor navigation, where we plan to give increased autonomy, sensorization, robustness and safety, as well as improving its communication system.

Underwater sensors. For ARESIBO requirements, new general-purpose hydrophones integrated with data loggers will be used, in order to meet the communication protocols for Manta Gateway. The system will communicate with the KAFKA system and provide real-time data from monitoring surface and subsurface noise, more particularly noise generated from speed boats, motorboats, ships and underwater vehicles.

UxV swarming and resource control. An innovative, optimized for real-life use, multi-robot coverage path planning algorithm is already developed from our research group for other projects. This algorithm provides complete coverage inside complex-shaped operational areas, while guaranteeing energy and operational resources efficiency. For the needs of this project, this algorithm is extended, added with additional capabilities such as the support of obstacles and no-fly-zones inside a region of interest and the assignment of different proportions of the operational area to each UxV according to their operational capabilities.

Moreover, this algorithm is used as a base for the development of a persistent coverage framework, incorporating the same features.

Visual recognition tools. Object detection is a common problem in computer vision and refers on identifying a region in an image where the target object is localized. After 2010, traditional ML methods based on handcrafted feature extraction reached a plateau. In 2014, Regions with CNN features broken the deadlocks [2], [3] and the field started to evolve at an unprecedented speed. In deep learning era, object detection can be grouped into two genres: "two-stage detection" and "one-stage detection", where the former frames the detection as a "coarse-to-fine" process while the later frames it as to "complete in one step". ARESIBO VisRes component will rely initially on Faster R-CNN as a two-stages detector [4] with proper modifications for alignment with the ARESIBO objectives. Target detection objectives in surveillance applications usually include combination of visible image-based methods and thermal image-based methods. The selection for the thermal based methods provides a rich source of temperature information, less affected by changing illumination or background clutters but thermal cameras lack in resolution and visual-based information that in general affects the detection accuracy. Therefore, image enhancement will also be investigated. There are various approaches to deal with image enhancement from Lanczos resampling [5] to internal/external similarity [6], [7], learning based methods such as random forest [8]. Since SRCNN [9] successfully introduced as a deep learning technique for the image enhancement problem, deep learning-based approaches have been used with large improvements in performance. Research in ARESIBO framework will motivate us to investigate novel methods in object detection and thermal image enhancement using CNN with the concept of RGB guidance. Finally, a major objective will be the decreased processing times in order to provide Visual Recognition capabilities in real and/or near-real times

Active sensing. The extensive experience of our group (ConvCAO - CERTH) in adaptive control and cognitive adaptive optimization methods (CAO toolset), is used for the development of an innovative solution for dynamic events monitoring. An adaptive optimization procedure, developed specifically for this project, acts in corporation with state-of-the-art object detection deep neural networks, in order to provide the best monitoring positions inside a region of interest, for increased quality and quantity for the detections. This solution provides increased situational awareness for dynamic evolving phenomena, in unknown and dynamically changing environments.

Communication protocols and cyber-security. The communication protocols are built on the experience TEKEVER has in building UAV systems and associated Ground Stations, and the integration with C2 Data Centers of Mission Operators. Guaranteeing the data exchange between all these components, and balancing all its constraints - bandwidth, reliability - and requirements - real-time, secure - is a difficult challenge, for which a prior background is a must. TEKEVER also brings in the experience of building large scale data exchange systems with data access authorization restrictions enforced by cryptography. Through this project, a standardization of encryption mechanisms and integration with cryptographically-enforced authorization mechanisms is expected.

Decision support tools. The interesting novelty of DST in the frame of ARESIBO is its strict dependence on data coming from Sensors together with attached sensors-provided metadata. Since its main aim is at supporting decisions on using in a mission an ARESIBO-provided resource/asset it will fill the gap of lack of knowledge on a specific sensor capability in a Control Room providing clear messages with detailed suggestions that may be taken into account by a Control Room operator. Hence it should be able to formulate clear short instructions that should be consumed in a Glass or an AR tablet device. Differently from traditional DSS, the system is mainly meant to produce shot messages for AR and not to provide a full-fledged investigation tools with a rich, computer based, gis-based user interface.

Simulation environments. The CMRE has previously implemented a federation of simulators for a maritime environment. This federation will be modified for ARESIBO sea scenarios and will serve as a starting point also for land scenarios. At the end of the project, we expect two technological advancements with regards to the simulation engine: the development of a federation of simulators for land environments and the ability to project both land and sea environments into virtual and/or augmented reality technology.

AR technologies and platforms for field operations. Ubimax is global market leader for industrial AR technologies by creating end to end solutions incorporating the state-of-the-art technologies. Their product is compatible with a wide range of heterogeneous AR smart glasses and devices. The development for ARESIBO Project in accordance with the end user requirements will help improve the overall product and the interoperability with various customer/partner solutions will help in scaling our solution on a larger scale.

AR technologies and platforms for tactical commander. In military domain it has been demonstrated that AR can raise the situational awareness by combining different sources of information such as terrain information and landmarks, and AR also has the potential to increase the feeling of presence and can be a tool to enhance communication between stakeholders. One of the bottleneck of the AR system has been usability of AR devices [10], [11]. ARESIBO will use latest AR technology for tactical commander e.g. Microsoft Hololens 2 [12] and/or iPad Pro (Fourth-generation 11") [13].

References

[1] "ROBORDER," [Online]. Available: https://roborder.eu/. [Accessed 5 6 2020].

[2] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proceedings of the IEEE conference on computer vision and pattern recognition,* p. 580–587, 2014.

[3] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Region based convolutional networks for accurate object detection and segmentation," *IEEE transactions on pattern analysis and machine intelligence,* vol. 38, no. 1, p. 142–158, 2016.

[4] S. Ren, K. He, R. Girshick and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems,* pp. 91-99, 2015.

[5] C. E. Duchon, "Lanczos Filtering in One and Two Dimension," *Journal of Applied Meteorology,* 1979.

[6] G. Freedman and R. Fattal, "Image and Video Upscaling from Local Self Examples," *ACM Transactions on Graphics (TOG),* 2011.

[7] J. Yang, Z. Lin and S. Cohen, "Fast Image Super-resolution Based on Inplace Example Regression," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2013.

[8] S. Samuel, C. Leistnerand and H. Bischof, "Fast and Accurate Image Upscaling With Super-Resolution Forests," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2015.

[9] C. Dong, C. C. Loy, K. He and X. Tang, "Image Super-Resolution Using Deep Convolutional Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI),* no. 9470, p. 947004, May 2015.

[10] E. Gans, D. Roberts, M. Bennett, H. Towles, A. Menozzi, J. Cook and T. Sherrill, "Augmented reality technology for day/night situational awareness for the dismounted Soldier. In Display technologies and applications for defense, security, and avionics IX; and head-and helmet-mounted displays," *International Society for Optics and Photonics,* vol. 9470, p. 11, May 2015.

[11] S. Lukosch, M. Billinghurst, L. Alem and K. Kiyokawa, "Collaboration in augmented reality," *Computer Supported Cooperative Work (CSCW),* vol. 24, no. 6, pp. 515-525, 2015.

[12] [Online]. Available: https://en.wikipedia.org/wiki/HoloLens_2. [Accessed 7 5 2020].

[13] [Online]. Available: https://en.wikipedia.org/wiki/IPad_Pro. [Accessed 7 5 2020].