INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 16, Issue: 6, Month: December, Year: 2021
Article Number: 4251, https://doi.org/10.15837/ijccc.2021.6.4251

CCC Publications
AGORA UNIVERSITY PRESS

C2 Advanced Multi-domain Environment and Live Observation Technologies

F.J. Perez, A. García, V. Garrido, M. Esteve, M. Zambrano

Francisco J. Pérez, Alberto García, Víctor J. Garrido, Manuel Esteve
Departamento de comunicaciones
Universitat Politècnica de València, España
*Corresponding author: frapecar@upvnet.upv.es

Marcelo Zambrano
1. Universidad Técnica del Norte, Ibarra, Ecuador
2. Instituto Superior Tecnológico Rumiñahui, Ecuador
omzambrano@utn.edu.ec

Abstract

Nowadays, the free movement of people and goods within the European Union is one of the topical issues. Each member state and border practitioner exploits its own set of assets in their goal of border surveillance and control. States have invested significantly in these assets and infrastructures necessary to manage and control the transit in the border areas. As new capabilities and assets become available and as current Command and Control (C2) systems become older, border control practitioners are faced with the increasing challenge of how to integrate new assets, command and control all of them in a coordinated and coherent way without having to invest in a completely new C2 systems built from the ground up. Therefore, and bearing in mind that the systems already developed up to date are very old and are not framed in a global standard data model, it has been identified, on one side the need to define a platform that allows to interact with multiple UxVs (land, sea and air), and on the other, unify all data models so that it can globalize and generate a much more concise analysis of what happens in places of conflict.

Keywords: UxVs, Border Control, Distributed Infrastructure, Situational Awareness.

1 Introduction

The free movement of persons and goods within European Union (EU) has led to the gradual abolition of border controls between member states of the Schengen agreement [14]. This had led to new challenges concerning the security of European external borders. Recent events (such as the correlated waves of illegal immigration and refugee crisis or the attacks in some countries) have contributed to the increasing pressure and stress put on the external borders of the EU. According to European Border and Coast Guard Agency [15], around 700 million people cross Europe's external borders every year. This creates a need to detect illegal activity without causing delays or obstacles for people. During 2018, various national authorities working in collaboration apprehended over 15 tonnes of cocaine and 46 tonnes of cannabis in the Atlantic and Mediterranean regions according to data extracted by Maritime Analysis and Operations Centre.[16]

Each member state and border practitioner exploit its own set of assets in their goal of border surveillance and control. States have invested significantly in these assets and infrastructures necessary to manage and control the transit in border areas. As new capabilities and assets become available (e.g. unmanned systems which are frequently based on stove piped control segments) and, as current Command and Control (C2) systems (some of which are proprietary, use closed interfaces and are monolithic) become older, border control practitioners are faced with the increasing challenge of how to integrate new assets and command and control all of them (old and new) in a coordinated and coherent way without having to invest in building it from the ground up. In order to make this task easier, a distributed platform has been designed and developed in the frame of CAMELOT project [17] founded by the EU. CAMELOT is based on the work from the Unmanned Aerial Systems Control Segment (UCS) [2] and the EUCISE 2020 [20].

The composition of this paper follows: Section 2 is architecture in that you can find subsections concerning the services, data model, adapters and CAMELOT command and control system, Section 3 is our expert results and the scenario analysis, Section 4 is our conclusions, future lines of research and discusses some issues of this work, and the final section is our references.

1.1 Objectives

Traditionally, UxVs manufacturers (OEM) have focused their designs on the un-manned platforms rather than on their Ground Control Stations (GCS) and this fact has led to the existence of a multitude of proprietary protocols in their communications and consequently, to a lack of interoperability among UxVs from different vendors.

The purpose of this paper and consequently of the CAMELOT project is to design an architecture that will allow the interoperability between the GCS of third party UxVs as well as with other manned assets, sensors or legacy command and control (C2) systems integrated in a common platform, as well as the development and implementation of a platform capable of monitoring and controlling multiple Unmanned Vehicles (UxVs) in order to organize and manage any mission with high degree of situational awareness, like for example rescue, search or supervision operations, among others [12].

One of the most relevant aspects is the capacity to manage different UxVs (ground, sea or air), since in this way global missions can be established and each stage of the mission can be managed in a different way. It is possible to reconfigure the mission to address aspects that arise during the process.

The architecture of the CAMELOT platform is based on a distributed system that will offer 1) scalability, 2) availability and 3) security capabilities. Therefore, CAMELOT is undoubtedly a unique platform that will allow to tackle the missions that were not possible or were very complex to carry out until now.

1.2 Related works

It is important to emphasise that CAMELOT is a totally innovative and pioneering project. No similar work exists, at least in practice and within the European Community. As we have been able to analyse multi-domain management is not a simple or trivial task, therefore, to date, management and operation tasks were being planned in a unitary way for a single area of application such as, for example, the management of multiple UAVs [11] or UGVs [1], only one type of domain, but so far there is no platform capable of managing different domains. Thus, the CAMELOT platform has been developed because of the advanced needs extracted by the end users of the project.

Table 1: Comparative CAMELOT framework VS Others

	CAMELOT framework	Others frameworks
Multi-domain	Yes	No
Customization	Yes	No
Data model	UCS (but ready to follow any standard)	Not specific
Adapters accepted	All	Only one

1.3 Motivation

Traditionally, UxVs were procured with their own stovepipe control segment (Ground Control Station or GCS). These GCS were, and often still are, typically closed systems utilizing proprietary interfaces. The proliferation of stovepipe control systems on a single C2 system raises issues concerning interoperability, training, capital investment, adaptation work, footprint and logistic support among others. Therefore, for end-users it becomes essential to develop the ability of commanding and controlling multiple UxVs as well as other sensors and delivering complex services (such as automatic asset tasking, mission planning and re-planning or 3D representation of threats to name a few) while using the same systems and environments (to rationalize costs and improve efficiency). This enables future cost effective capability upgrades and reuse of selected C2 components which can be sourced from multiple vendors.

Although some efforts and work have already been done in the command and control systems field (in particular control segments for UxVs in the defense domain), a lot still needs to be done in order to achieve a single widely supported standard. Furthermore, it makes little sense for end-users and practitioners to develop and maintain their own large software repositories associated with the C2 capabilities (it would be easier to maintain a list of available services with associated metadata). Industry also has incentives to adopt such standards: namely that services or modules developed based on an industry widely accepted model would be easier to integrate with a standard compliant system. Specialist providers (in particular SMEs) can supply niche components without the need to invest in gaining expertise in the complete range of system interactions. This model has proven to work before, for example the adoption of NATO standard STANAG 4586 (UAS interoperability) by component suppliers of vehicles and sensors [9].

2 Architecture

2.1 Overview

The CAMELOT platform architecture is based on a distributed system that will offer scalability, availability and security capabilities, as required according to the end-users.

The core of the system will be designed in a flexible and modular framework so that implementation of new functionalities is easy and cost-effective. CAMELOT architecture can be found in Figure 1.

The main modules developed along the CAMELOT framework are discussed as follows:

- Automatic asset tasking and control (AATC).
  The platform applies state-of-the-art solutions currently used in some security systems such as video surveillance, to provide tasks in the context of border surveillance as it uses different methods to control different assets connected to the platform. This module has the ability to support a variety of sensors (from various manufacturers) and focuses on a microservices based paradigm.

  Most unmanned vehicles accept the assignment of tasks either automatically (through mission programming) or through modifications by operators during execution. This is true for UAVs, UGVs, USVs and also UUVs (although in the latter the capability may not be in real time). Most vehicles now have an autonomous decision making capability, in addition to the programming of route points and the modification of the mission.

  The project involves the implementation of an algorithm that allows cooperative behavior among different vehicles optimizing their capabilities in terms of detection/location performance and

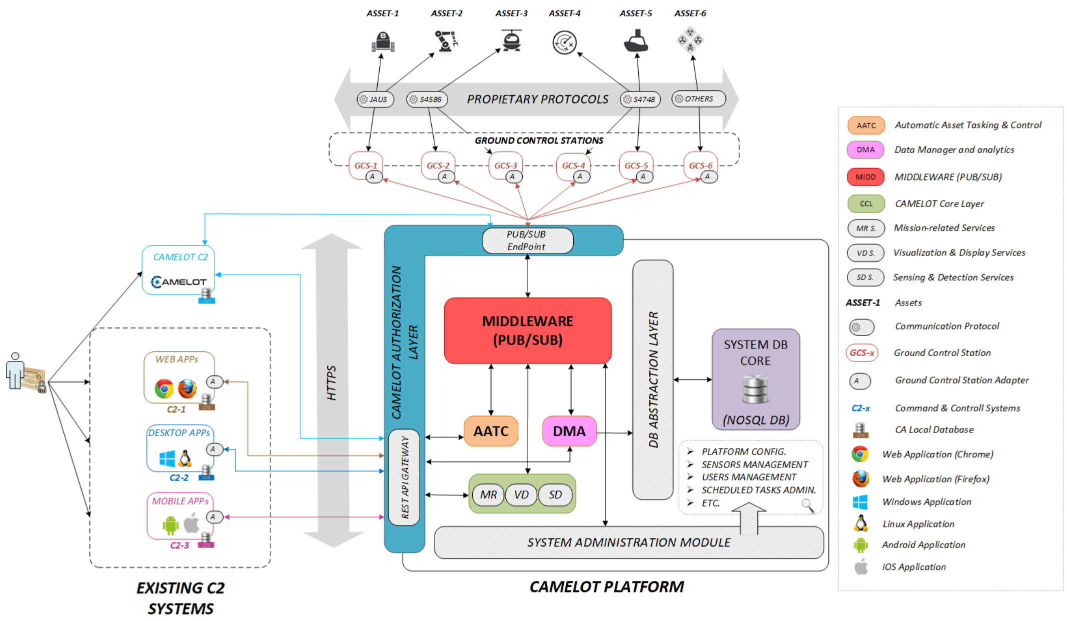


Figure 1: CAMELOT proposal Architecture

area coverage as well as allowing signaling to allow platforms to confirm detections by other sensors and platforms in order to reduce false positives and human intervention.

Today, when an operator of a control unit carries out the programming of any sensor, it is done fully manually or at least partially manually. At the very least, the operator is required to have the following set of skills:

- Knowledge of the sensors needed to carry out the mission.
- Knowledge of programming all the sensors involved in the mission.

These activities greatly increase the workload of the operator without an effective gain. In addition, manual programming makes it more difficult to distinguish between bad configurations and false negatives.

Therefore, thanks to CAMELOT, the management of the sensors will be reduced basing on their availability and operational capacity, enabling automatic management and facilitating decision making.

- Data manager and analytics (DMA).
  DMA brings to the platform a set of advanced analysis and data management functions to perform information operations. This Component is responsible for retrieving information from the CAMELOT middleware, processing it and producing alerts and warnings based on the end-user operational needs. DMA components consists of several services and modules.

  The Early Warning Engine (EWE) facilitates the business logic of DMA, thereby producing meaningful information in form of alerts. Besides the core functionality of the DMA, a set of additional services is enforced in order to ease the operations of the EWE. The overall architecture of the DMA is illustrated in Figure 2.

  The Early Warning Engine (EWE) is responsible for processing the incoming data and producing meaningful information like alerts and warnings. The implemented mechanism consists of several modules that undertake messaging operations and extract valuable situational insights. Specifically, the EWE consists of the following modules:

  - Rule engine
  - Alert generator
  - Data serializer

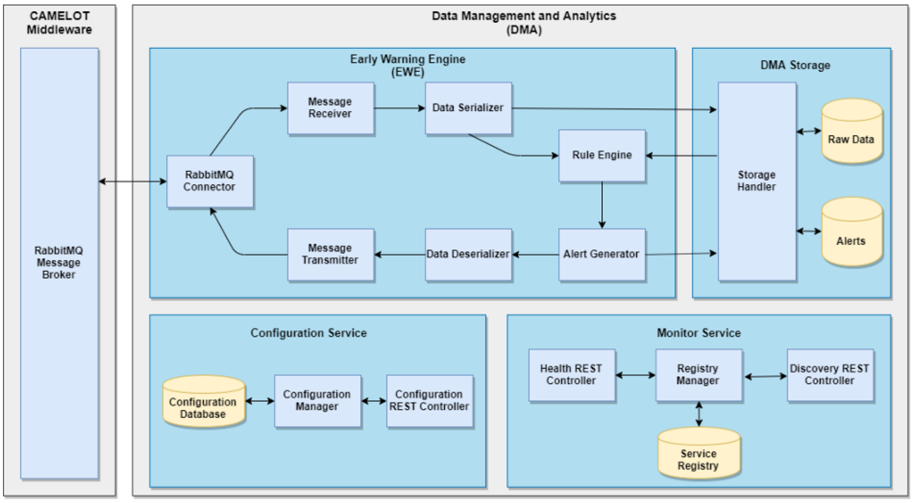


Figure 2: DMA architecture

- Data deserializer
- Message received
- Message transmitter
- RabbitMQ connector

The Rule Engine performs cross checks between data and rules in order to detect possible threats or abnormal acts. Thereafter, the Alert Generator consumes the intelligence produced by the Rule Engine and formulates the appropriate alerts. The processed data originates from a) the CAMELOT middleware (real-time data) and b) the DMA storage (historical data).

The rest of the EWE modules are facilitating the messaging operations for communicating with the CAMELOT middleware. The translation of the data between the format it is stored and the format it is transmitted is facilitated by the Data Serializer and the Data Deserializer. The Message Receiver and the Message Transmitter are interconnected with the integrated RabbitMQ Connector [18] for handling the flow of data between the DMA's EWE and CAMELOT's middleware. The DMA Storage maintains all the information that is handled by the EWE. Its database consists of the following types of data:

- Raw Data: received from the CAMELOT middleware.
- Alerts: produced by EWE's Alert Generator.

The Storage Handler module manages the CRUD operations upon the databases. The Rule Engine of the EWE retrieves the historical data of the DMA Storage and combines it with real-time data for validating the enforced rules. If an anomaly is detected, then the Alert Generator produces a relevant alert that is transmitted to the middleware but is also stored in the DMA Storage for future use by the Rule Engine. The Monitor Service offers service status monitoring and service discovery functionalities. It is responsible of tracking the status of the DMA services and informing them about the details of each service. This is made possible by maintaining a registry that contains information about the DMA services. The service registry contains the following information:

- Service identifier
- Service name
- Service provider
- Register timestamp
- Last reported status

- – Last status timestamp

  A DMA service shall communicate with the Monitor Service through a) the Discovery REST Controller and b) the Health REST Controller. The first is responsible for service discovery operations and the latter for service status monitoring operations. For both types of operations, a separate REST interface is provided for establishing a communication between the Monitor Service and the rest services of the DMA.

  The Configuration Service contains parameters related to the DMA configuration. It functions as a registry for connection parameters, rule definitions, database configurations, and other runtime options. The EWE, the DMA Storage and the Monitor Service retrieve their respective configurations from the Configuration Service when the DMA initiates. This approach offers a centralized parameterization registry, thus providing a practical method for configuring the DMA. The Configuration Service consists of the following modules:

  - Configuration Database: a non-relational (NoSQL) database that contains the configuration parameters.
  - Configuration Manager: manages CRUD operations upon the Configuration Database.
  - Configuration REST Controller: provides the requesting services with the necessary configuration parameters.

  The following configuration parameters will be available in the Configuration Service:

  - CAMELOT Middleware connection parameters.
  - VPN connection parameters.
  - RabbitMQ connection parameters.
  - RabbitMQ Connector message subscriptions.
  - EWE rule definitions.
  - EWE active alerts.
  - DMA Storage connection parameters.
  - Monitor Service connection parameters.
  - Configuration Service connection parameters.

- CAMELOT core layer.
  It contains all internal functionalities required by the platform to perform CAMELOT main capabilities. This module includes Mission-Related Services (MRS), Visualization and Display Services (VDS) and Sensing and Detection Services (SDS).

- Adapters.
  These ensure communication with assets that make use of different protocols and avoid ad-hoc implementation of the CAMELOT data model. Whenever a new asset from a different provider is added to the system, an adapter from the CAMELOT data model to the corresponding protocol will be implemented. Further information can be seen in following sections.

- Middleware.
  It allows the interaction and the information exchange among the different submodules and services deployed in CAMELOT platform using Publish-Subscribe paradigm. The mechanism helps the CAMELOT' module providers to define several Open Systems Interconnection (OSI) layers without having to manually program each one. This will allow faster implementation of the CAMELOT guaranteeing the scalability, modularity and distributed characteristics that the consortium desires. RabbitMQ was the selected choice for the architecture middleware and JavaScript Object Notation (JSON) as the payload for the messages. See Figure 3.

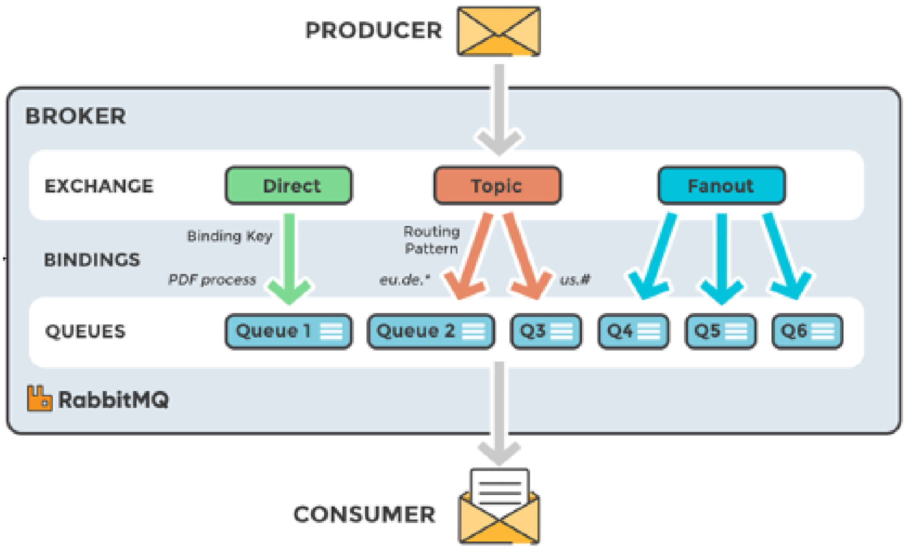


Figure 3: Middleware concept schema

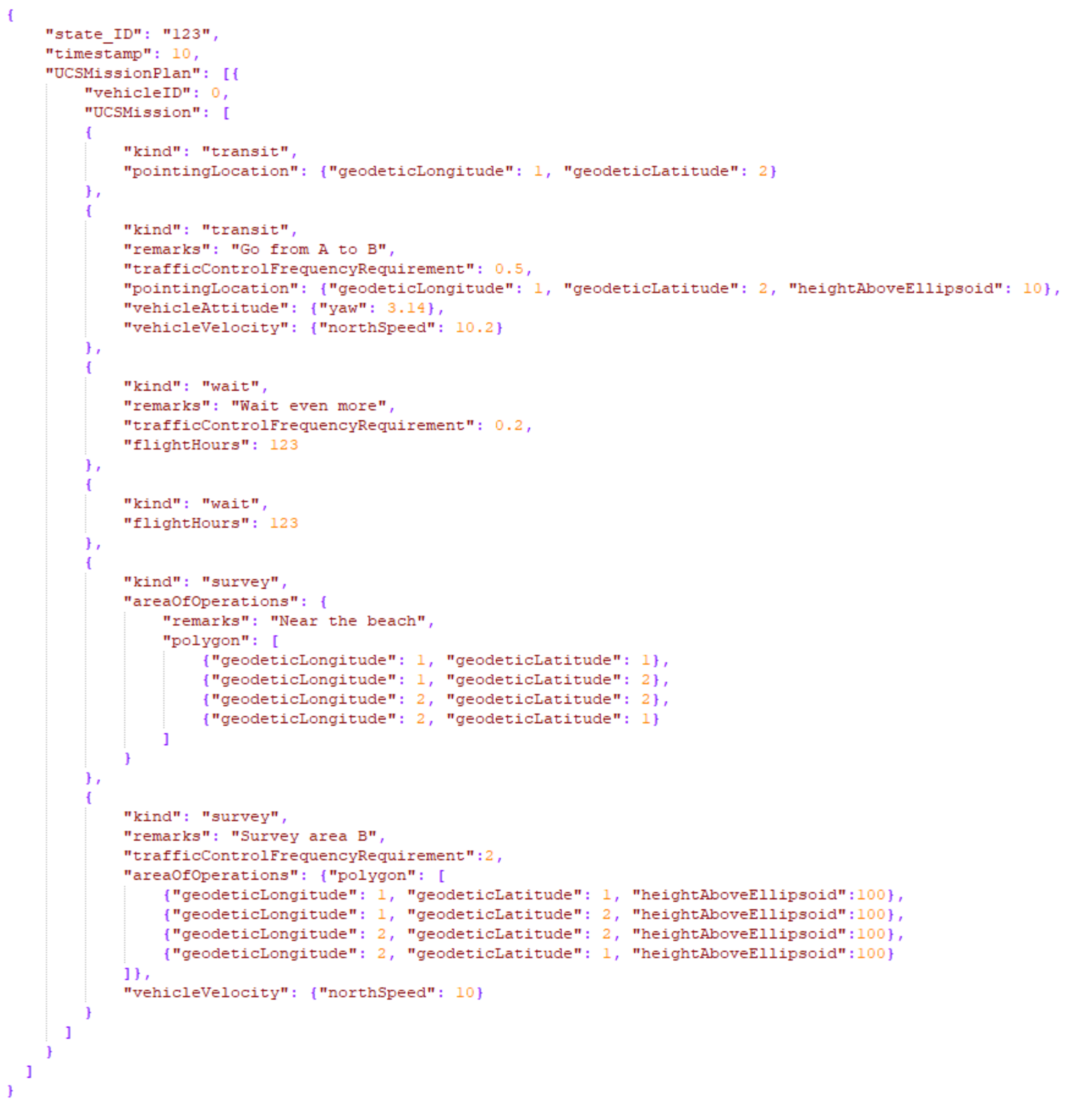
- CAMELOT authorization layer.
  The platform will expose different interoperability services accessible through a Virtual Private Network (VPN) that will allow the different C2 systems and GCSs to interact with the platform feeding and retrieving information or even consuming the available services.

- Rest API gateway.
  It allows to receive requests from both public internet and internal services and forward them to the best suited microservice instance. API Gateway come with a specific and helpful tool such as an authorization module. It provides a solid layer of defense against user authentication or data validation.

- C2 system.
  It allows end users to view and receive information from the different services available on the platform through a Pub/Sub End Point or Rest API Gateway. Within the platform there will be the possibility of interacting with the existing C2 that will have access to the services they are able to manage. On the other hand, CAMELOT's own C2 has been specifically developed so that it is compatible with all the services available on the platform.

2.2 Data model

CAMELOT needs to discover, control, and receive data from a wide variety of UxVs and their sensors. As such, the data model must provide a common description for entities within the system, their relationships, and the messages between them. The platform must also be interoperable with other systems via the Common Information Sharing Environment (CISE) [19] interface. The data model is a specification of the platform that cannot be completely closed at the current stage. The majority of the interfaces have been defined but the need to add specific fields may appear throughout the implementation of the system. Additionally, in order to comply with the interoperability requirements, an abstraction level between the data model platform specific protocols (JAUS, JANUS, STANAG 4586, ROS, MOSS, etc.) are being developed. The messages may be implemented through JSON in order to unify the same structure for all of them.



```
{
    "state_ID": "123",
    "timestamp": 10,
    "UCSMissionPlan": [{
        "vehicleID": 0,
        "UCSMission": [
        {
            "kind": "transit",
            "pointingLocation": {"geodeticLongitude": 1, "geodeticLatitude": 2}
        },
        {
            "kind": "transit",
            "remarks": "Go from A to B",
            "trafficControlFrequencyRequirement": 0.5,
            "pointingLocation": {"geodeticLongitude": 1, "geodeticLatitude": 2, "heightAboveEllipsoid": 10},
            "vehicleAttitude": {"yaw": 3.14},
            "vehicleVelocity": {"northSpeed": 10.2}
        },
        {
            "kind": "wait",
            "remarks": "Wait even more",
            "trafficControlFrequencyRequirement": 0.2,
            "flightHours": 123
        },
        {
            "kind": "wait",
            "flightHours": 123
        },
        {
            "kind": "survey",
            "areaOfOperations": {
                "remarks": "Near the beach",
                "polygon": [
                    {"geodeticLongitude": 1, "geodeticLatitude": 1},
                    {"geodeticLongitude": 1, "geodeticLatitude": 2},
                    {"geodeticLongitude": 2, "geodeticLatitude": 2},
                    {"geodeticLongitude": 2, "geodeticLatitude": 1}
                ]
            }
        },
        {
            "kind": "survey",
            "remarks": "Survey area B",
            "trafficControlFrequencyRequirement":2,
            "areaOfOperations": {"polygon": [
                {"geodeticLongitude": 1, "geodeticLatitude": 1, "heightAboveEllipsoid":100},
                {"geodeticLongitude": 1, "geodeticLatitude": 2, "heightAboveEllipsoid":100},
                {"geodeticLongitude": 2, "geodeticLatitude": 2, "heightAboveEllipsoid":100},
                {"geodeticLongitude": 2, "geodeticLatitude": 1, "heightAboveEllipsoid":100}
            ]},
            "vehicleVelocity": {"northSpeed": 10}
        }
        ]
    }
  ]
}
```

Figure 4: Mission plan message

UCS Architectural Model was developed by the USA Department of Defense [21] (acquired a posteriori by the Society of Automotive Engineers), using UML diagrams [7], with SoaML [22] and custom tailored extensions. These extensions characterize the functionality, features, and constraints that might be implemented in an unmanned control station. The authoritative UCS Architectural Model package comprises four packages: the Service Contract and Non-Functional Property Model, the System Use Case Model, the Domain Participant Model and the Information Model (Data Model). The UCS services are described by the service packages within the Domain Participant Model. The Information Model (Data Model) is composed by two major data models, the conceptual and logical data models. These models are described below:

- Conceptual data model (CDM)
  This model comprises the abstract data that is required to build, operate manage and maintain the UCS, being structured into a common package and an air package.

- Logical data model (LDM)
  It contains refinements and projections of the elements from the CDM. Each Conceptual Data Model Element (CDE) may be refined by multiple realizations at the LDM level, each adding a different selection of logical representation details. The details added in the LDM include units, reference frames, value kinds, constraints, and measurement precision. In addition, conversion relationships between logical elements may be specified. It also provides the projection of the CDM onto concrete message types.

2.3 Adapters

The proposed architecture is an extension of existing candidate architecture and data model from NATO NIAG SG.202 Conceptual Data Model for Multi-Domain Unmanned Platform Control System Services [8]. It Iincludes a specific adapter layer for each GCS and C2 that allows interoperability of systems designed to work with external protocols. This layer converts bi-directionally between the external protocol of the other system and the LDM of an instantiation of the MDCS architecture and will be able to make requests and receive data through a pub/sub endpoint or the rest API gateway.

Centralized C2 nodes ashore and afloat, as well as individual cross-domain (air/sea/ground) unmanned systems, must be able to exchange information seamlessly for networked operations that support distributed control and flexible hierarchies conforming to evolving tactical scenarios. The overwhelming number of unmanned air systems relative to sea or ground vehicles supported early establishment of the NATO Standard Agreement (STANAG) 4586 messaging protocols as a candidate for networking unmanned systems [6].

Fortunately, a limited number of these standards and protocols have been identified, so an implementation should be able to develop this limited set of protocol adapters and attach them to the required communication device(s) used by the specific instance of the platform. The details of manipulating the communication device and the implementation of the platform specific protocol (e.g. 4586, JAUS or 4748) are the responsibility of the protocol adapter.

Each protocol adapter must be configured and must enable discovery and initialization of the protocol specific devices and must convey this information to instances of the CAMELOT services so that they can be properly configured to use the protocol specific ground control stations. The adapter generated for each ground control station will provide communication with the platform via pub/sub as shown in Figure 5. On the other hand, the adapter is in charge of transforming the data to fit each proprietary protocol (STANAG 4586, JAUS and STANAG 4748) and exchange data with the Ground Control Station.

Below sections briefly present each standard, which are the most widely adopted and the most interesting for the scope of the CAMELOT project.

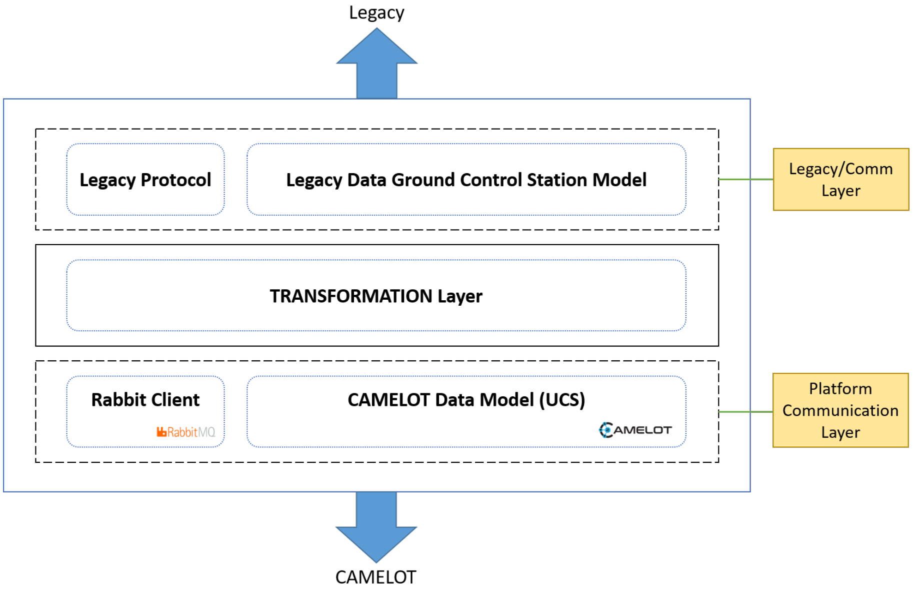


Figure 5: CAMELOT adapter schema

- Joint architecture for unmanned systems (JAUS).
  JAUS establishes a common set of message formats and data protocol for UGVs made by various manufacturers. It enables communication with a control of unmanned systems across the entire unmanned system domain. It incorporates a component-based, messaging-passing architecture that promote the stability of capabilities by projecting anticipated requirements as well as those

currently needed. JAUS is built based on four principles: vehicle platform independence, mission isolation, computer hardware independence and operator use independence [3].

- STANAG 4586.
  STANAG 4586 is an unmanned aerial equivalent of JAUS. It was developed to provide a level of unmanned aerial vehicle (UAV) interoperability across different entities to allow to quickly assign tasks to available assets, mutually control these vehicles and their payloads and rapidly disseminate tactical information to the collective force as required [3].

- STANAG 4748 (JANUS).
  JANUS is the protocol used by UAVs for initial contact establishment between UAV and operator. It is an open-source robust signaling method for underwater communications [4]. It has been designed to minimize the changes required to bring existing UW communications equipment into compliance, leveraging the inherit flexibility of modern digital communications systems and existing acoustic frequencies and bandwidths [10].

2.4 CAMELOT command and control system

This section describes the process of Human Machine Interface (HMI) design and the graphical/visual organization of the HMI as well as the structuring of the main functions. Regarding the HMI structure, it becomes clear that the HMI needs some main navigation, where the user can choose between the different CAMELOT tools. Furthermore, a subnavigation is needed, where the user can choose among the actual functions of a selected tool. This basic idea is reflected in the following general navigation schema for the HMI [23]. Figure 6 shows a partitioning of the HMI into four main parts:

- Main navigation area:
  For this a tab-based navigation was chosen.

- Submenu area:
  Represented by a list of selectable tool functions.

- Component area:
  This area is assigned to the tool functions selected by the user - it can be further partitioned: an optional function - specific navigation area on the top and a main area, for the actual user interaction, below.

- Notifications box:
  Specific notifications can be shown to the user here.

Other general functions are located at the top of the HMI:

- Login/logout:
  Allows the access of a user.

- Settings:
  Allows a logged user to specify preferences or change settings.

- Notifications:
  Allows the notifying of certain system events to logged users.

As far as the functionalities of the HMI are concerned, three clearly differentiated parts can be extracted:

- Overview:
  Provides a general understanding of the territory to be explored. Serve to visualize active or inactive alerts and assets, explore areas, plan missions, get ideas to generate plans, generate irruption alerts, manage your assets and visualize video streams. Overall, it is the first tool used in order to do any possible task.

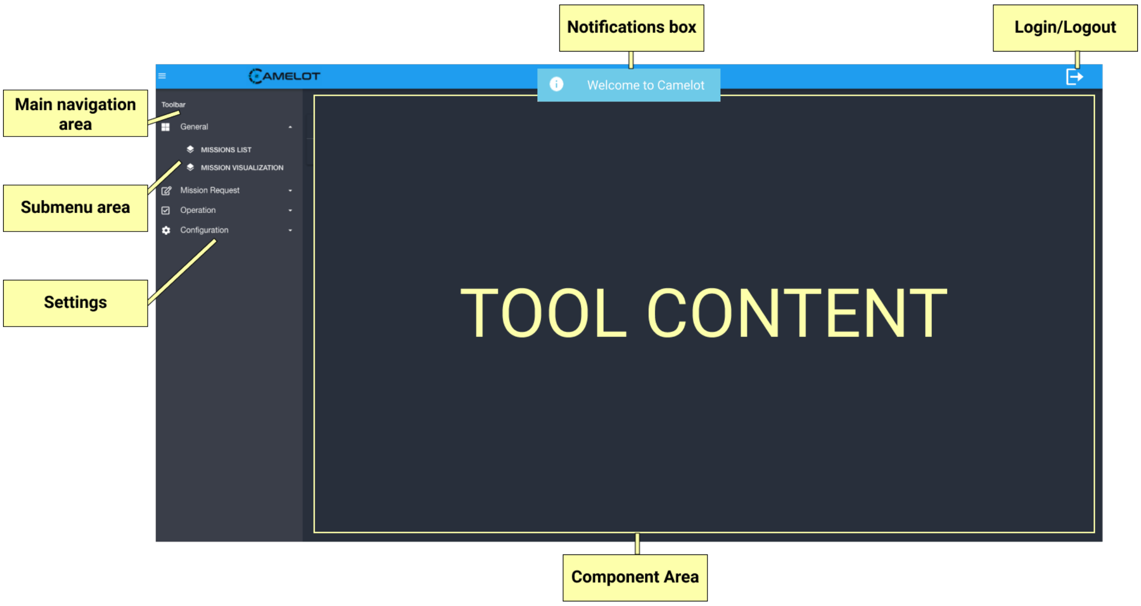


Figure 6: HMI component distribution

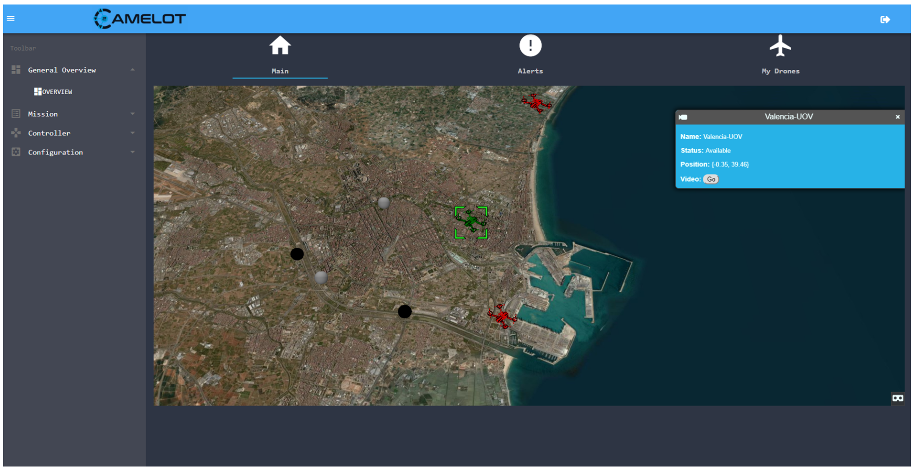
- Mission plan:
  Makes it possible to manage everything related to the creation of new mission plans and the search for a specific mission. This is why the application has two possible options, in the search section will be established to analyze, so that any mission present within that region will be shown to the end user through a list. A new mission can be generated from one shown in the list or it can simply be used to obtain new ideas on how it could be generated. On the other hand, it provides a powerful tool for the generation of new mission plans, for it must include metadata references and the route or area you want to cover. It will be possible to identify areas where the passage is prohibited or if you simply want to bypass that area. Finally, once it is created, it will be included in the database.

- Energy management:
  Mitigates the problem of energy overload in relation to radio transmissions. In order to solve the problem, it is proposed to reduce the number of transmission points, which leads to a significant reduction in battery consumption, although on the other hand the UxV will lose control of it and increase uncertainty. It is for this reason that an a priori estimate should be made on how to proceed in the number of points to avoid. Therefore, in order to solve the logic of the estimation of the position, the following equations are proposed:

$$\begin{gathered}
V[k] = (wdm[k] + wim[k])r/2, \\
W[k] = (wdm[k] - wim[k])r/b, \\
X_{est}[k+1] = X_{est}[k] + V[k] * T * cos(T * W[k] + \Theta_{est}[k]), \\
Y_{est}[k+1] = Y_{est}[k] + V[k] * T * sin(T * W[k] + \Theta_{est}[k]), \\
\Theta_{est}[k+1] = \Theta_{est}[k] + T * W[k],
\end{gathered}$$

  In summary, the main idea of the energy saving module is to minimize the data transmission between the base station and the asset. For this purpose it is intended to send only one information package from the base in given time periods. Thus, with the previously mentioned equations it can be estimated where the asset should be situated.

  Continuous position corrections will be applied as packages arrive.



Name: Valencia-UOV
Status: Available
Position: {-0.35, 39.46}
Video: Go

Figure 7: Web HMI platform's components

3 Scenario and results

In order to validate the scenario, the use case proposed is represented in the following timeline (Figure 8), taking into consideration all the limitations and regulations present in the project.

In this way it is possible to analyze and extract the issues that can be found and establish possible solutions. A UAV is flying over a specific area and suddenly detects a motions event and notifies the CAMELOT Automated Surveillance System/Platform and Sensors. Due to the pattern detection the drone can verify if the object is a person, a group of persons, an inflatable boat or others. Also running motion detection and sensors at the same time can verify if an illegal crossing of the front boundary of the border zone occured. On the other hand, the motion detection can identify an object left on the river bank and detect if an additional illegal crossing of a border occurs.

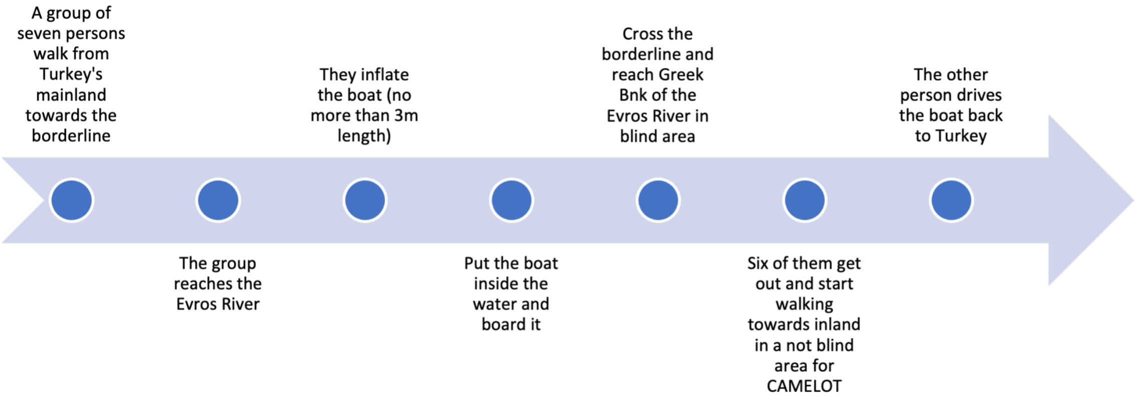

A group of seven persons walk from Turkey's mainland towards the borderline
The group reaches the Evros River
They inflate the boat (no more than 3m length)
Put the boat inside the water and board it
Cross the borderline and reach Greek Bnk of the Evros River in blind area
Six of them get out and start walking towards inland in a not blind area for CAMELOT
The other person drives the boat back to Turkey

Figure 8: Use case timeline

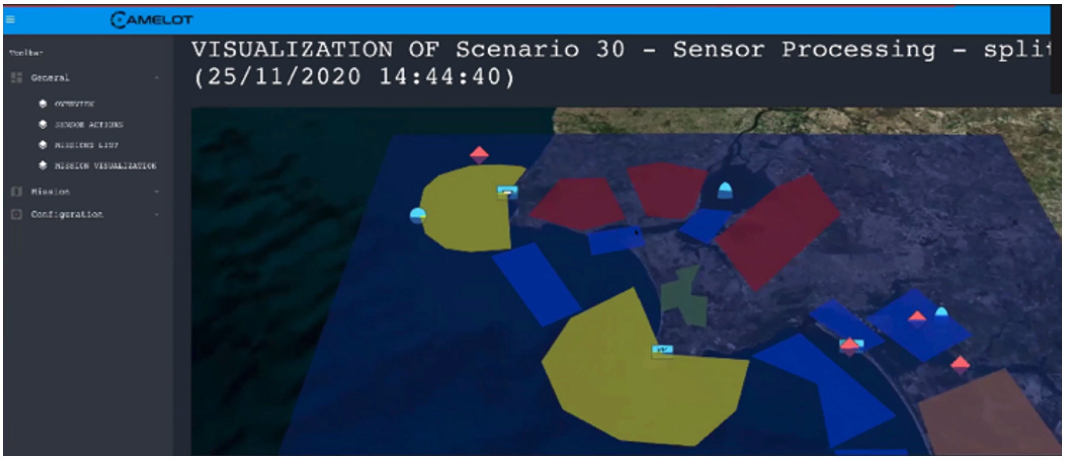
Finally, upon the detection of an event, the system should be able to send a real-time alert to the monitoring station including the following information:

- Location event
- Time of event
- Type of event
- Initialize mission planning operations
- Initialize any other UxV engagement
- Initialize manage mission data

- Information distribution to unit level/rescue team

In this way, the identification of the problem can be achieved extremely quickly in order to manage and plan the actions to be executed, intervene, and finally establish a pattern of lessons learned. This is a very important step because if the incident was not recorded before it will be necessary to establish rules of action. On the other hand, if the incident was already recorded, it will be important to reinforce what was learned and check that the pattern to follow is suitable.

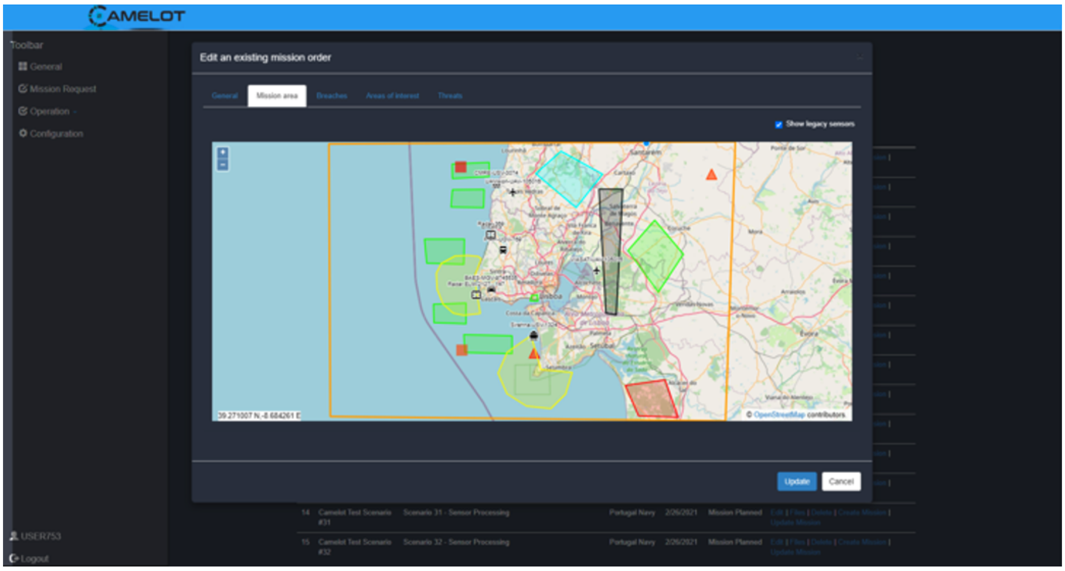
The main expectation to be covered is a complete management and control capability of multiple manned and unmanned devices on the same platform, despite several obstacles such as the adaptation of each device to the project environment or the interoperability between different platforms as shown in Figure 9. For validation, the aforementioned use case is proposed, including an extra mission to detect drug smuggling.



VISUALIZATION OF Scenario 30 - Sensor Processing - split
(25/11/2020 14:44:40)

Figure 9: General context of the mission

The platform is not only prepared to operate with multiple devices, but also makes it possible to manage several missions at the same time. The added value it brings to the territorial defense domain therefore greatly enhances its situational awareness as indicated in Figure 10.



Edit an existing mission order

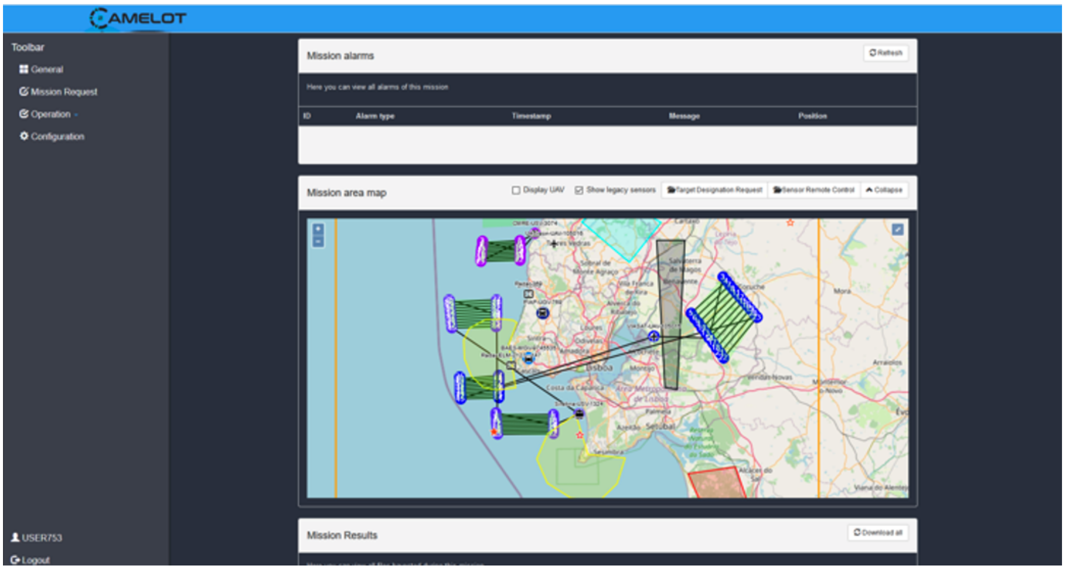
☑ Show legacy sensors

39.271007 N, -8.684261 E

```
14  Camelot Test Scenario  Scenario 31 - Sensor Processing  Portugal Navy  2/26/2021  Mission Planned  Edit | Files | Delete | Create Mission |
    #31                                                                                              Update Mission
15  Camelot Test Scenario  Scenario 32 - Sensor Processing  Portugal Navy  2/26/2021  Mission Planned  Edit | Files | Delete | Create Mission |
    #32                                                                                              Update Mission
```

Figure 10: Mission planner

To implement the envisaged solution, full interoperability of the different components is expected. The platform is able to autonomously manage [5] which of the available devices are best suited to cover each patrol area as shown in Figure 11. Therefore, once the mission is launched, the areas to be covered by each device will be self-assigned [13], with the flexibility to re-plan the mission at any time if an extra event should occur.

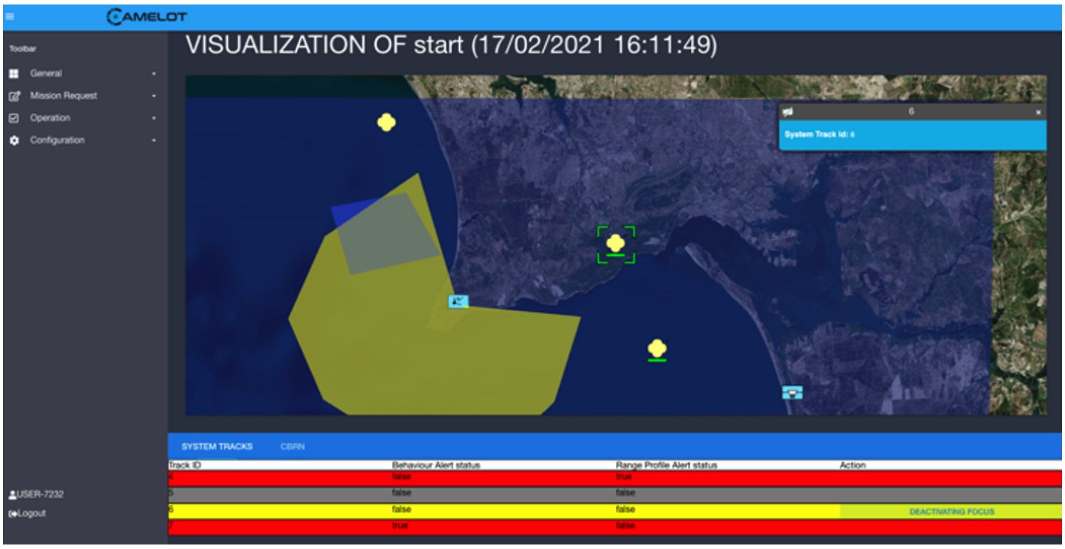


Here you can view all alarms of this mission

ID	Alarm type	Timestamp	Message	Position

Figure 11: Device optimizer for the mission

All of this must take into account national and EU legal frameworks, the time constraints of each mission (duration of device autonomy), human resource constraints (lack of personnel, training, equipment support) and, finally, the reduction of risk for human personnel, as this way they are not confronted with risk in the foreground. Thus, through the platform and the missions generated in CAMELOT and thanks to the service for detecting suspicious elements and elements that should not fall within a specified range, it is possible to set up alerts that will be analysed and evaluated instantly as indicated in Figure 12. To do this, the platform has a management table of possible alerts in which it can establish monitoring patterns and alert that new threats have occurred, having at all times global awareness of what is happening in our control framework.



VISUALIZATION OF start (17/02/2021 16:11:49)

Track ID	Behaviour Alert status	Range Profile Alert status	Action
4	false	true	
5	false	false	
6	false	false	DEACTIVATING FOCUS
7	true	false	

Figure 12: Alert manager

Finally, several demonstrations have been developed with end-users who have found that the platform has a rich variety of tools that not only improve situational awareness of missions, but also make it possible to carry out all missions at the same time, with a successful outcome.

4 Conclusions

This paper presents a summary of the proposed architecture, the different adapters used to manage the communication between the Ground Control Station and the Middleware, the data model used and the development of a specific C2 system to control the assets. The prototype will be validated through a real scenario with a set of modules deployed in order to control the flow of events that happen in a specific location. The selection of the deployed components will be made taking into account the improvement of suspicious event detection tasks. In addition, due to the importance of energy saving in missions, the platform has been built with a module that plans and controls the frequency of data transmission between the radio and the asset finding a balance between uncertainty and control. In the future work will focus on the use of the platform in various use cases and scenarios in order to verify and validate its use in any field, and on the development and integration of various components needed to perform additional tasks.

Being able to manage multiple UxVs globally means greater response to any event. Operators will not only be able to acquire greater situational awareness, they will also be able to achieve more advanced optimisation of resources. This is why the study of use cases related to advanced management with a wide variety of use cases is so important.

There are multiple study limitations, as to realise a prototype capable of performing all the above-mentioned tasks requires conceptual and laboratory tests that do not come close to reality with all its obstacles. The clearest example is the energy management module. It is not feasible to test with real UxVs without having obtained excellent results in the laboratory (which do not guarantee success due to the issues that a real sample may encounter). Replicating real scenarios requires a great deal of resources and management, making research and testing even more complicated.

It is important to highlight the capacity to solve multiple problems at the same time. After exhaustive analysis, it was concluded that this is a unique platform, as previously it had not been considered, due to the difficulty of having multiple suppliers, to have a single platform that could control different UxVs at the same time. The resolution capacity increases notably and the response time decreases, so it is concluded that the implications generated by this project are extremely beneficial for the correct analysis and management of any mission.

4.1 Future research

The main focus of future research is to improve the data model and provide a global interoperability architecture for command and control systems allowing the interactions with unmanned devices. However, the possibility of having a system that can guarantee rescues and make the CAMELOT platform a global portal that allows not only asset management but also gives the possibility to operators to perform multiple tasks should be analysed as aspects to be taken into account.

CAMELOT was conceived as a platform capable of interoperating and presenting immediate responses to any simple or complex problem that might occur. However, and thanks to the analysis and tests carried out, it was possible to extract the need to extend the platform's capabilities to improve data processing and anticipatory management through machine learning and Artificial Intelligence algorithms. Nowadays, it is essential to be able to foresee any event and execute a specific action to solve the problem. For this reason, various behavioural patterns have been determined for the autonomous and anticipated identification of when a problem may arise and will be used to train a dataset of casuistry that will be used to improve the algorithm. Thanks to this, operators will be able to an-ticipate and guarantee correct operation without having to carry out exhaustive monitoring. This will increase the effectiveness rate and problem-solving capacity.

Finally, it would also be interesting to improve the video transmission algorithms in order to be able to process and extract any data that may be vital in an operation. Thanks to the way the project's development architecture has been designed, it is possible to quickly include any service that could be considered as a requirement, that is why CAMELOT is prepared for any future line of development.

Since it is a working prototype, it is possible to extract and analyse some advantages and disadvantages that may arise. Thus, we could say that the project will have a changeable life cycle due to its innovative nature.

Funding

This work has been performed under the H2020 740736 project CAMELOT, which has received funding from the European Union's Horizon 2020 Program. This paper reflects only the authors' view, and the European Commission is not liable to any use that may be made of the information contained therein.

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

References

[1] Anisi, D.; Ogren, P.; Hu, X.; Lindskog, T. (2008). Cooperative surveillance missions with multiple unmanned ground vehicles (UGVs), *2008 47th IEEE Conference on Decision and Control.* 2444–2449, 2006.

[2] Batavia, P.H; Ernst, R.; Fisherkeller,K.; Gregory, D.; Hoffman, R.D. (2011). The Uas Control Segment Architecture, *SPIE Defense, Security, and Sensing*

[3] Blais, C.L. (2016). Unmanned systems interoperability standards, *The NPS Institutional Archive*

[4] Heidemann, J.; Wei Ye; Wills, J.; Syed, A.; Yuan Li. (2006). Research challenges and applications for underwater sensor networking, *IEEE Wireless Communications and Networking Conference, 2006. WCNC 2006.* 1, 228–235, 2006.

[5] Hong, Y.; Jung, S.; Kim, S.; Cha, J. (2021). Autonomous Mission of Multi-UAV for Optimal Area Coverage, *Sensors.* 21(7):2482, 2021.

[6] Incze, Michael L.; Sideleau, Scott R.; Gagner, Chris; Pippin, Charles A. (2015). Communication and collaboration of heterogeneous unmanned systems using the joint architecture for Unmanned Systems (JAUS) standards, *OCEANS 2015 - Genova.* 1–6, 2015.

[7] Liu, J.; Zhou, J; Liu, C.; Yang, Z.; Wang, Z.; Zhang, Y. (2014). Modeling and Analyzing Flight Control Software of Unmanned Aerial Vehicle Using UML and B Method, *Journal of Software.* 9(4)

[8] Pastore, T.; Galdorisi, G.; Jones, A. (2017). Command and Control (C2) to enable multi-domain teaming of unmanned vehicles (UxVs), *Oceans 2017 - Anchorage.* 1–7, 2017

[9] Pradhan, M.; Tiderko, A.; Ota, D. (2017). Approach towards achieving an interoperable C4ISR infrastructure, *2017 International Conference on Military Technologies (ICMT).* 375–382, 2017

[10] Potter, J.; Alves, J.; Green, D.; Zappa, G.; Nissen, I.; McCoy, K. (2014). The JANUS underwater communications standard, *2014 Underwater Communications and Networking (UComms).* 1–4, 2014

[11] Tan, Y.; Zhao, R; Zhu, X.; Zhang, B. (2007). The Design and Implementation of Autonomous Mission Manager for Small UAVs, *2007 IEEE International Conference on Control and Automation.* 177–181, 2007

[12] Vargas-Ramírez, N.; Paneque-Gálvez, J. (2019). The Global Emergence of Community Drones (2012–2017), *Drones.* 3,76. 2019

[13] Weldon, W.T.; Hupy, J. (2020). Investigating Methods for Integrating Unmanned Aerial Systems in Search and Rescue Operations, *2007 IEEE International Conference on Control and Automation*. 4(3):38, 2020

[14] [Online]. Available: https://ec.europa.eu/home-affairs/what-we-do/policies/borders-and-visas/schengen_en/, Accesed on 17 May 2021.

[15] [Online]. Available: https://europa.eu/european-union/about-eu/agencies/frontex_en/, Accesed on 17 May 2021.

[16] [Online]. Available: https://cdt.europa.eu/pt/node/2498/, Accesed on 17 May 2021.

[17] [Online]. Available: https://www.camelot-project.eu/, Accesed on 17 May 2021.

[18] [Online]. Available: https://www.rabbitmq.com/, Accesed on 17 May 2021.

[19] [Online]. Available: https://www.efca.europa.eu/en/content/common-information-sharing-environment-cise/, Accesed on 17 May 2021.

[20] [Online]. Available: http://www.eucise2020.eu/, Accesed on 17 May 2021.

[21] [Online]. Available: https://saemobilus.sae.org/content/AS6518/, Accesed on 17 May 2021.

[22] [Online]. Available: https://www.omg.org/spec/SoaML/1.0.1/PDF, Accesed on 17 May 2021.

[23] [Online]. Available: https://www.defensedaily.com/wp-content/uploads/post_attachment/206477.pdf, Accesed on 17 May 2021.

Cite this paper as:

Pérez P.J.; García A.; Garrido V.J.; Esteve M; Zambrano Marcelo. (2021). C2 Advanced Multi-domain Environment and Live Observation Technologies, *International Journal of Computers Communications & Control*, 16(6), 4251, 2021.
https://doi.org/10.15837/ijccc.2021.6.4251