Interoperability Open Architecture of Unmanned Systems

Wilmuth Müller
Interoperability and Assistance Systems
Fraunhofer IOSB
Karlsruhe, Germany
wilmuth.mueller@iosb.fraunhofer.de

Florian Segor
Interoperability and Assistance Systems
Fraunhofer IOSB
Karlsruhe, Germany
florian.segor@iosb.fraunhofer.de

Dirk Mühlenberg
Interoperability and Assistance Systems
Fraunhofer IOSB
Karlsruhe, Germany
dirk.muehlenberg @iosb.fraunhofer.de

Bart Driessen
TNO
The Hague, The Netherlands
bart.driessen@tno.nl

Vincent de Geus
TNO
The Hague, The Netherlands
vincent.degeus@tno.nl

***Abstract*—Within the INTERACT Project, funded by the DG-DEFIS of the European Commission and managed by the European Defence Agency (EDA), interoperability concepts aiming at enhancing the capabilities of European armed forces to safely, effectively and flexibly operate unmanned and manned systems in joint or combined operations have been developed. The challenge lied in creating overarching interoperability concepts for defence systems in general and unmanned systems in particular. The different interoperability concepts have been integrated into an open interoperability architecture for unmanned systems. In this paper the developed open interoperability architecture is presented.**

Keywords— unmanned systems, interoperability, open architecture

I. Introduction

As UxSs increasingly become a key instrument in civilian and defense operations, the necessity for interoperability support is more pressing than ever before. Multiple aspects are to be considered, such as piloting, controlling multiple UxS by a single control station, exchange of data collected during a mission and data link interoperability.

NATO Standardization Agreements (STANAGs) are commonly used for existing systems, and are widely adopted in the defense industry. However, for the complete UxS area considering all the involved domains (ground, maritime, air) no STANAG has been promulgated (NATO STANAG 4817 - Multi-Domain Control Station is under development and not publicly available yet).

In response to these challenges, the INTERACT project provided a set of interoperability solutions and standardisation proposals, which will enable the coordinated use of multiple platforms by a single, standardised control station and the controlled hand-over of platforms to other control stations. This coordinated use of multiple platforms and the hand-over possibility together with an envisioned standard proposal for an interoperable data link including standardized security will enable the deployment of complex UxS platforms in a network-centric environment within future multi-national operations. In addition to this inter-system interoperability and in order to ease the upgrade and adoption of novel payloads and maintaining and upgrading equipment and components to the state-of-the-art, INTERACT also designed and proposes as standard a set of interoperable interfaces between the subsystems and payloads within an unmanned system (intra-system interoperability). These standard interfaces will lead to plug-and-play connection capabilities, making it thus easy to install new mission payloads or change them on demand. This will enable the adaptability of unmanned systems to specific mission requirements.

INTERACT defined concepts and proposed standards to be used for ensuring interoperability of payloads and carriers, defined the functionality needed for implementing an interoperable control station, capable of simultaneously controlling multiple heterogeneous UxS as well as responding flexibly to changes and requirements in relation to other control units, proposed the use of standards and defined requirements at functional level of future communication needs, including their adaptability to future environments and high-data rate exchanges. INTERACT also define the means to enable the use of heterogeneous unmanned systems in a variety of combinations, including their usage as an autonomously acting team. In addition, INTERACT developed an Interoperability Open Architecture of interoperable UxS and control stations.

This paper presents the developed Interoperability Open Architecture, specifically the viewpoints Concepts and Service Specification (see Chapter III Architecture Framework for an explanation of the viewpoints).

II. Related Work

The goal of interoperability of unmanned systems is handled in a series of scientific papers. However, mostly only one single domain (air, maritime, ground) is addressed, without providing a complete multi-domain architecture.

The development of interoperability and standards have been addressed in a series of projects that focus on the cooperation and interaction between several UxVs and the operator, to perform tasks associated to military missions like: search and rescue (SAR), reconnaissance, patrolling, payload transportation, among other tasks.

The project Deployable Search and Rescue Integrated Chain with Unmanned Systems (DARIUS) [1] focused on the deployment of UxVs, from multiple agencies, in various environments where SAR scenarios are being played, contributing to the development of interoperability between UxVs, and determining the requirements for future SAR UxVs. The development of DARIUS aims to benefit its users by enabling the control of multiple UxVs, sharing of UxVs (as well as collected information) across different users within the same mission, and the integration of UxVs in command and control and communication chains.

The project Integrated Components for Assisted Rescue and Unmanned Search Operations (ICARUS) [2] also developed a system of UxVs to be used in SAR Operations. It focused on developing unmanned SAR technologies for detecting, locating and rescuing human victims. With this objective the creation of a set of tools for unmanned SAR usable by the end-users has been associated. The ICARUS system establishes a network between the control stations and the UxVs so that vehicles can communicate with each other and with the control stations, in order to promote autonomous navigation, victim detection and identification, and information flows, which allows the control station to process the incoming data into a geographical information, and thus allowing the creation of a situational map that promotes situational awareness and decision-making. ICARUS uses UAVs, UGVs and USVs that integrate ROS and Joint Architecture for Unmanned Systems (JAUS) controllers, which connect to the command and control module through JAUS's bridges, which achieves interoperability and promotes the use of JAUS as a standard.

[3] analysed interoperability of unmanned systems in military maritime operations and developed a controller for unmanned aerial systems operating in maritime environments. Their work focused on the autonomous landing of an UAV on board of a maritime vessel. Their system consists of an unmanned aerial vehicle and a ground control station (GCS), with the addition of an onboard controller at the UAV. This controller establishes a communication link with the GCS so that the operator can control the UAV through the on-board controller. The controller than performs its commands by communicating with the auto-pilot module via the MAVLink protocol.

The Joint Architecture for Unmanned Systems (JAUS) [4] is an international standard that establishes a common set of message formats and communication protocols for supporting interoperability within and between unmanned vehicles and ground control stations. It was originally chartered by the United States Department of Defense (DoD) to provide an open architecture for the domain of Unmanned Ground Robots. JAUS was later converted to an international industry standard. The main goal of JAUS is to structure communication and interoperation of unmanned systems within a network. A JAUS system is made up of subsystems connected to a common data network. A Subsystem typically represents a physical entity in the system network, such as an unmanned vehicle or operator control unit. The JAUS network is further subdivided into hierarchical layers. Subsystems are divided into Nodes, which represent a physical computing end-point in the system. For example, a Node might be a computer or microcontroller within a Subsystem. Nodes can then host one or more Components, which are commonly applications or threads running on the Node. Finally, Components are made up of one or more Services. A Service simply provides some useful function for the system.

In [5] the interoperability topic is discussed blending a review of the technological growth from 2000 onwards with recent authors' in-field experience; the paper focuses on the aspect of interoperability among unmanned maritime vehicles (UMVs). The paper describes the experience from a sea trial exercise, where interoperability has been demonstrated by integrating heterogeneous autonomous UMVs into the NATO Centre for Maritime Research and Experimentation (CMRE) network, using different robotic middlewares and acoustic modem technologies to implement a multistatic active sonar system.

The authors of [6] focus in their study towards standards for public safety small unmanned aircraft systems (sUAS) pilot training requirements for disaster management.

In [7] an overview of existing unmanned systems interoperability standards is provided. The standards are listed and briefly described. Among others, JAUS, the US National Information Exchange Model, the Unmanned Aircraft System (UA) Control Segment (UCS) Architecture, the US Joint Common Unmanned System Architecture, and STANAG 4586 are presented.

Interoperability between unmanned ground systems and command and control systems is addressed in [8]. The paper describes the research and experiment efforts of the NATO STO group IST-149-RTG capability concept demonstrator for interoperability within unmanned ground systems and C2. The main purpose of the group was to investigate possible standards for controlling UGVs and tests them in a real-world scenario.

STANAG 4586 [9], [10] provides the definitions of the architecture and messages required for the interoperability of UAVs in complex NATO Combined/Joint Services Operational Environment. Essentially, it comprises the 'rules' that will allow an operator to have a defined level of control over any Unmanned Aerial Vehicle (UAV) that is standard compliant. At the same time, C4I systems will have access to UAV payload product.

The micro air vehicle link (MAVLink) [11] is an open source communication protocol, which contains a messages' library, focused for small UAVs. It was developed in 2009 by Lorenz Meier. MAVLink specifies a comprehensive set of messages exchanged between unmanned systems and ground stations. This protocol is used in major autopilot systems, mainly ArduPilot and PX4, and provides powerful features not only for monitoring and controlling unmanned systems missions but also for their integration into the Internet. However, the protocol is vulnerable to network attacks [12].

The Robot Operating System (ROS) [13] is a middleware which collects software frameworks for robot software development. It is not an operating system, it just provides computer cluster like hardware abstraction, device control, convey message between the process and packet management. It creates note which multiplex sensors, actuators, controls and other messages. ROS is under open source licensed [14]. ROS 2 is the successor of the original ROS with the aim to mitigate some of the most important drawbacks of the ROS approach. The Robot Operating System – M is based on ROS 2 and focuses on the application of ROS for military robots and robotic application [15].

III. Architecture Framework

The framework that is used to create the INTERACT Architecture is the NATO Architecture Framework v4 (NAFv4). This framework is described in detail in a publicly available deliverable from NATO [16]. the following sections a brief overview of the different viewpoints that are included in this framework, as well as the meta-model that is used to model these viewpoints are presented.

A. Viewpoints

Part of the NAFv4 framework describes a number of different viewpoints which can be modelled to address different concerns. These viewpoints are shown in a grid which can be seen in Figure 1. The rows of the grid represent the Subject of Concern, while the columns represent the Aspect of Concern. So, for example the Service Specifications row is all about Services and each column represents a certain way to look at (or organize) the Services. These Aspects of Concern are summarized in Table 1.

Table 1: NAFv4 Aspects of Concern

ASPECTS	DESCRIPTION
TAXONOMY	Specialization hierarchies of the main architecture elements of each row
STRUCTURE	Composition diagrams, which describe how elements are assembled
CONNECTIVITY	Interconnection diagrams, focusing on dependencies, interfaces and interactions
BEHAVIOUR	Diagrams focusing on how things work: • Processes: Process flows and decomposition • States: State diagrams • Sequences: Sequence diagrams
INFORMATION	Diagrams showing what information/data is used and how it is structured
CONSTRAINTS	Diagrams which model rules that govern the elements of each row
ROADMAP	Project timelines and milestones that affect elements in the architecture

The rows of the grid (Subject of Concern) can be seen as a level of abstraction which assists in the process of modelling complex systems by breaking the problem down in smaller areas of focus with traceability between them. The Subjects of Concern are summarized in Table 2.

Table 2: NAFv4 Subjects of Concern

SUBJECTS	DESCRIPTION
CONCEPTS	Focus on Capabilities (an ability to do something) in line with enterprise strategy
SERVICE SPECIFICATIONS	Focus on Services (a unit of work through which a provider provides a useful result to a consumer) which directly support the operational domain and therefore can be linked to Capabilities
LOGICAL SPECIFICATIONS	Focus on a solution-independent description which models how the system can accomplish missions
PHYSICAL RESOURCE SPECIFICATIONS	Focus on a description of Resources (e.g. people, organizations, software, hardware) and how they should be configured and connected to deliver Capabilities and Services
ARCHITECTURE FOUNDATION	Focus on the Architecture itself including the administrative aspects

As described in [16] the selection of viewpoints must be tailored to the specific architecture effort. The viewpoints that are developed for the Interoperability Open Architecture are highlighted in Figure 1.

In the next chapter (IV Views and Subviews), the modelled viewpoints are presented.

B. Meta-Model

The Meta-model describes the model behind the model, i.e. the element types and valid relations that can be present in each viewpoint. More details can be found in the modelling guide from NATO [17] or in the set of documents of the specific NAFv4 extension for Enterprise Architect that was used [18].

The following types of elements will be used throughout the architecture, grouped by the different rows of the NAFv4 grid:

Concepts

Capability	A high level specification of the enterprise's ability to do something
EnterpriseVision	The future state of the enterprise, without regard to how it is to be achieved
EnterpriseGoal	Goals that must be satisfied to ultimately reach the EnterpriseVision
EnterprisePhase	A current of future state of the enterprise
ActualEnterprise-Phase	The current state of the enterprise
WholeLifeEnterprise	A purposeful endeavor of any size involving people, organizations and supporting systems

Service Specifications

ServiceSpecification	The specification of a set of functionality provided by an element for the use of others
ServiceSpecificationRole	Usage of a ServiceSpecification in the context of another ServiceSpecification to create a whole-part relationship
ServiceFunction	An activity that describes the functionality associated with a ServiceSpecification, indepently of how it is implemented

Logical Specifications

OperationalPerformer	A logical entity that is capable to perform OperationalActivities
OperationalRole	Usage of an OperationalPerformer in the context of another OperationalPerformer or OperationalArchitecture to create a whole-part relationship
OperationalActivity	An activity that captures a logical process, indepently of how the process is carried out.
ActualEnvironment	A description of the circumstances of an environment
OperationalArchitecture	A type used to denote a model of the Architecture, described from the operational perspective

Infor-mationElement	An item of information that flows between OperationalPerformers and is produced and consumed by OperationalActivities that the OperationalPerfomers are capable to perform

Physical Resource Specifications

CapabilityConfiguration	A composite structure representing physical and human resources in an enterprise, assembled to meet a capability
ResourceRole	Usage of a CapabilityConfiguration or System in the context of another CapabilityConfiguration or System to create a whole-part relationship
DataElement	A formalized representation of data that is managed by or exchanged between resources
ResourceInterface	A declaration that specifies a contract between two or more CapabilityConfigurations or Systems which have an interaction
ResourcePort	An interaction point for a CapabilityConfiguration or System through which it can interact with the outside environment and which is defined by a ResourceInterface
System	An integrated set of elements, subsystems or assemblies that accomplish a defined objective. These elements include products (hardware, software, firmware), processes, people, information, techniques, facilities, services and other support elements

Architecture Foundation

Architectural-Description	A work product used to express the architecture of some System of Interest. It provides executive-level summary information about the architecture description to allow quick reference and comparison
ProtocolStack	A sub type of Protocol that contains ProtocolLayers, defining a complete stack
ProtocolLayer	Usage of a Protocol in the context of another Protocol to create a whole-part relationship
Protocol	A Standard for communication over a network. Protocols may be composite, represented as a ProtocolStack made up of ProtocolLayers
Standard	A ratified specification that is used to guide or constrain the architecture. A Standard may be applied to any element in the architecture

IV. Views and Subviews

A. Concepts

Within the Concepts view the Enterprise Vision, the Capability Taxonomy, and the Capability Dependencies have been developed.

The C2 Viewpoint (Figure 3) is concerned with scoping the architecture effort and providing the strategic context for the capabilities described in the architecture. The Enterprise Vision related to the scope of the INTERACT project is to deliver the effective means to ensure the interoperability and standardization of different unmanned systems, equipment, components, and procedures in order to maximize benefit and optimize adoption and integration of unmanned systems in the operation of EU armed forces. This is related to Enterprise Phase 1 – UxS Interoperability Research.

From this Enterprise Vision the capabilities needed to realize the vision have been derived and organized into a taxonomy. The C1 Viewpoint (Figure 2) shows the Capability Taxonomy.

Between these capabilities different dependencies exist. For performing Joint Intelligence, Surveillance, and Reconnaissance (JISR) using unmanned vehicles a capability for transporting data is needed, thus there exists a dependency between the “JISR using UxVs” capability and the “Data Transport” capability (Figure 4).

B. Service Specifications

For implementing the needed capabilities, a series of services have been specified, together with their mapping to the respective capabilities.

For example, Tasking and Re-tasking services are needed to implement the JISR Tasking capability, Data Collection Services and Payload Data Processing Services are needed to implement the JISR Collecting capability.

Figure 5 shows the upper-level services and their mapping to capabilities.

The service specifications and their taxonomy are also defined with a specific viewpoint, the S1 – Service Taxonomy view. In addition to the high-level service taxonomy, each service may be refined into a taxonomy. Figure 6 shows the taxonomy of the Payload data processing services.

The structure of services is described with the S2 viewpoint. The S2 presents

- How Service Specifications are structured (which services are part of higher level services)
- How Service Specifications depend on other Service Specifications
- Which Standards shall be used to implement the Service Specifications

Figure 7 shows the structure of the specified services, their dependencies, the roles of the different services and to what standards the respective service conforms.

The specified services are capable of performing a series of functions. The S4 viewpoint identifies the functions that are performed by the different services.

Figure 8 shows the functions performed by the Payload data processing services.

Figure 1: NAFv4 viewpoints

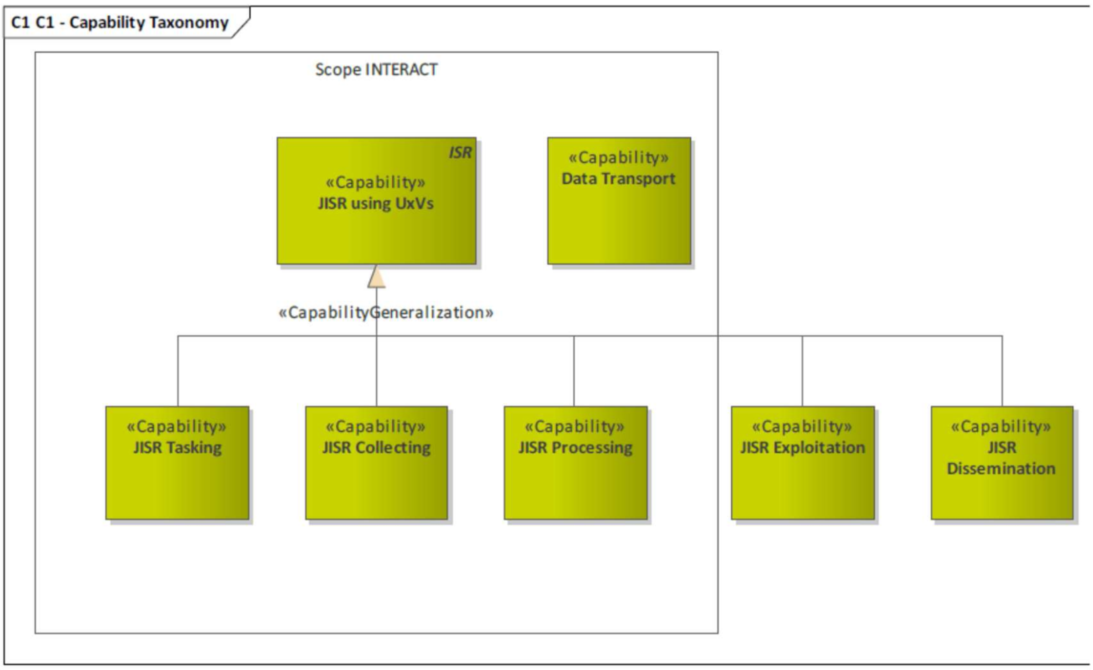

Figure 2: C1 – Capability Taxonomy

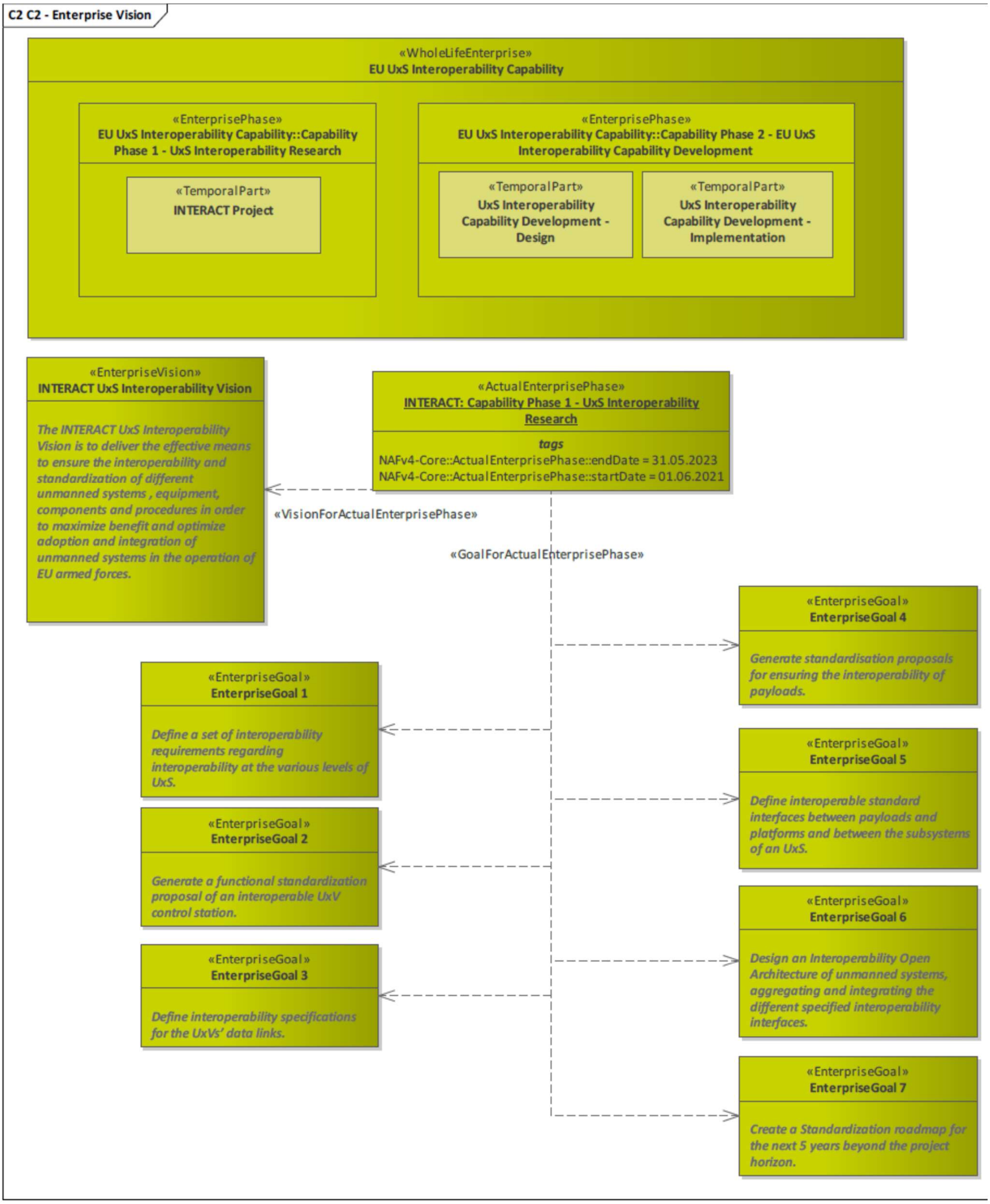

Figure 3: C2 – Enterprise Vision

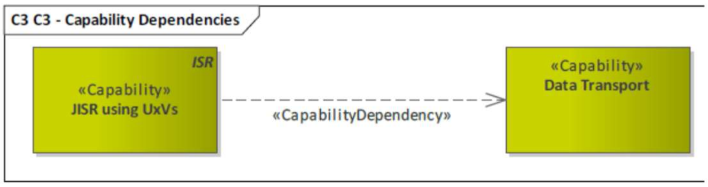

Figure 4: C3 – Capability Dependencies

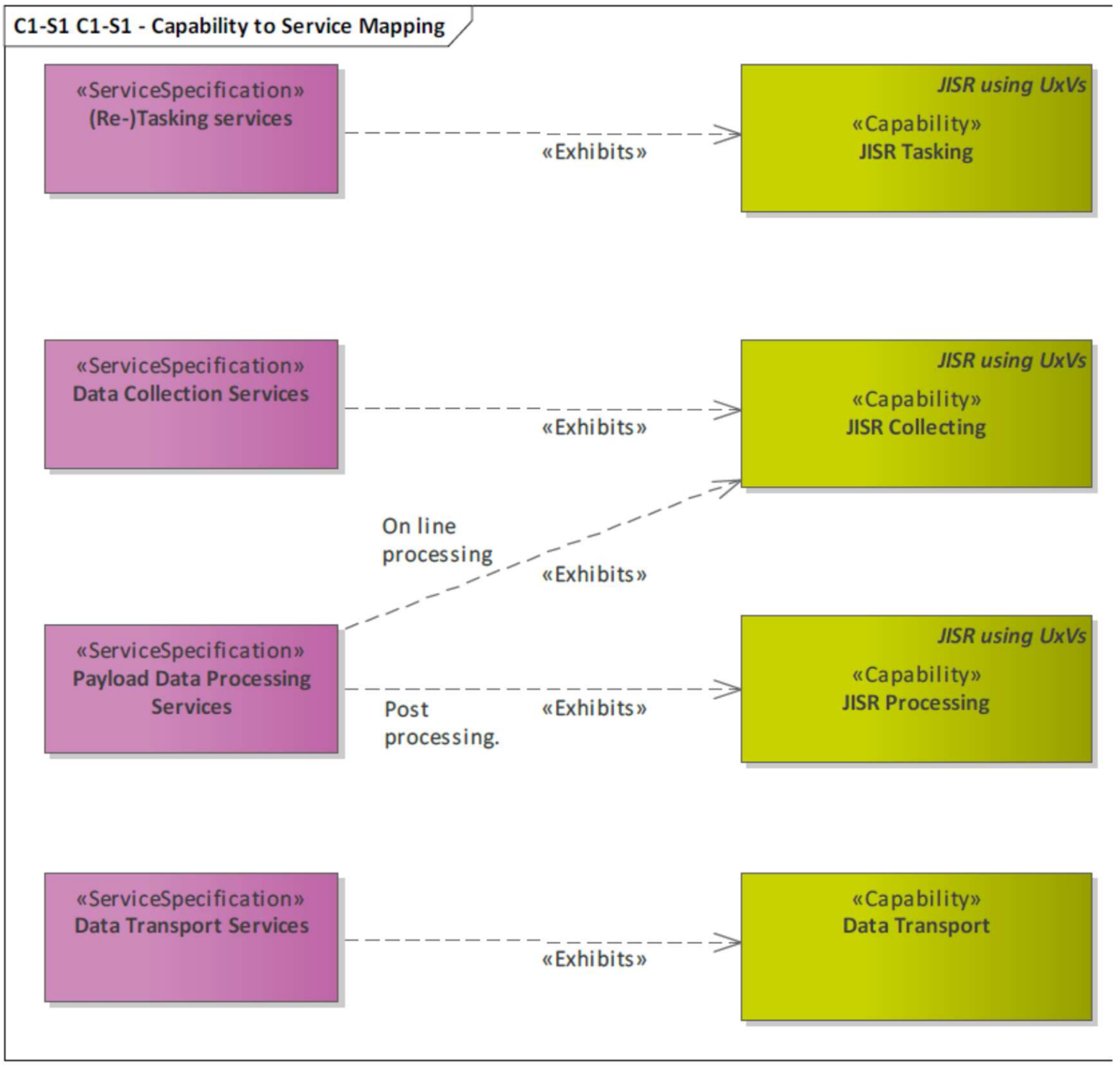

Figure 5: C1 – S1 – Capability to Service Mapping

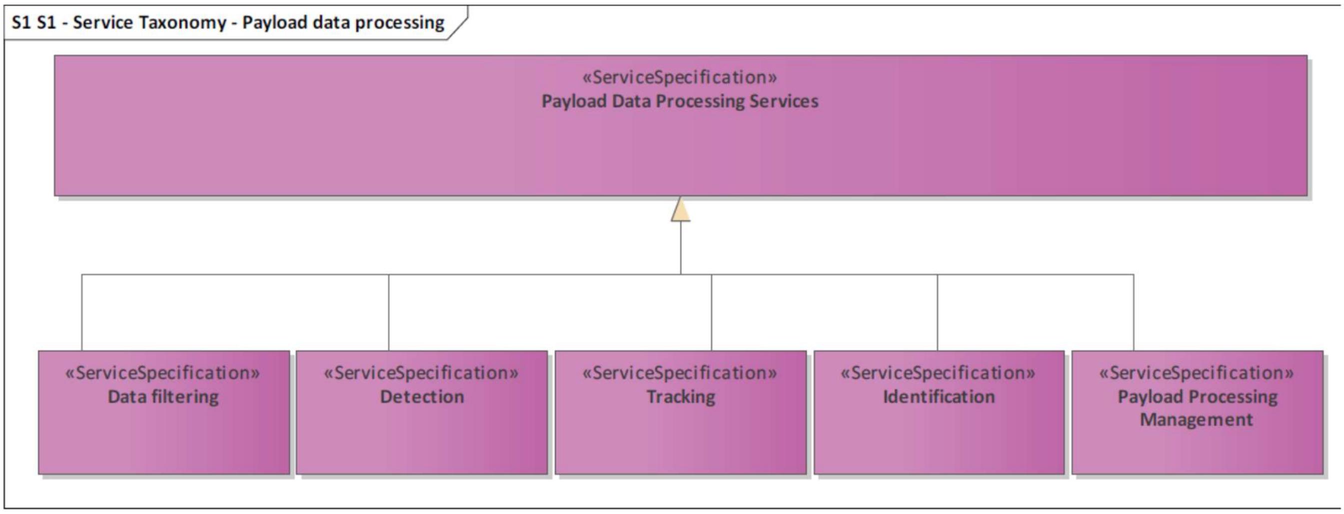

Figure 6: S1 – Service Taxonomy – Payload data processing

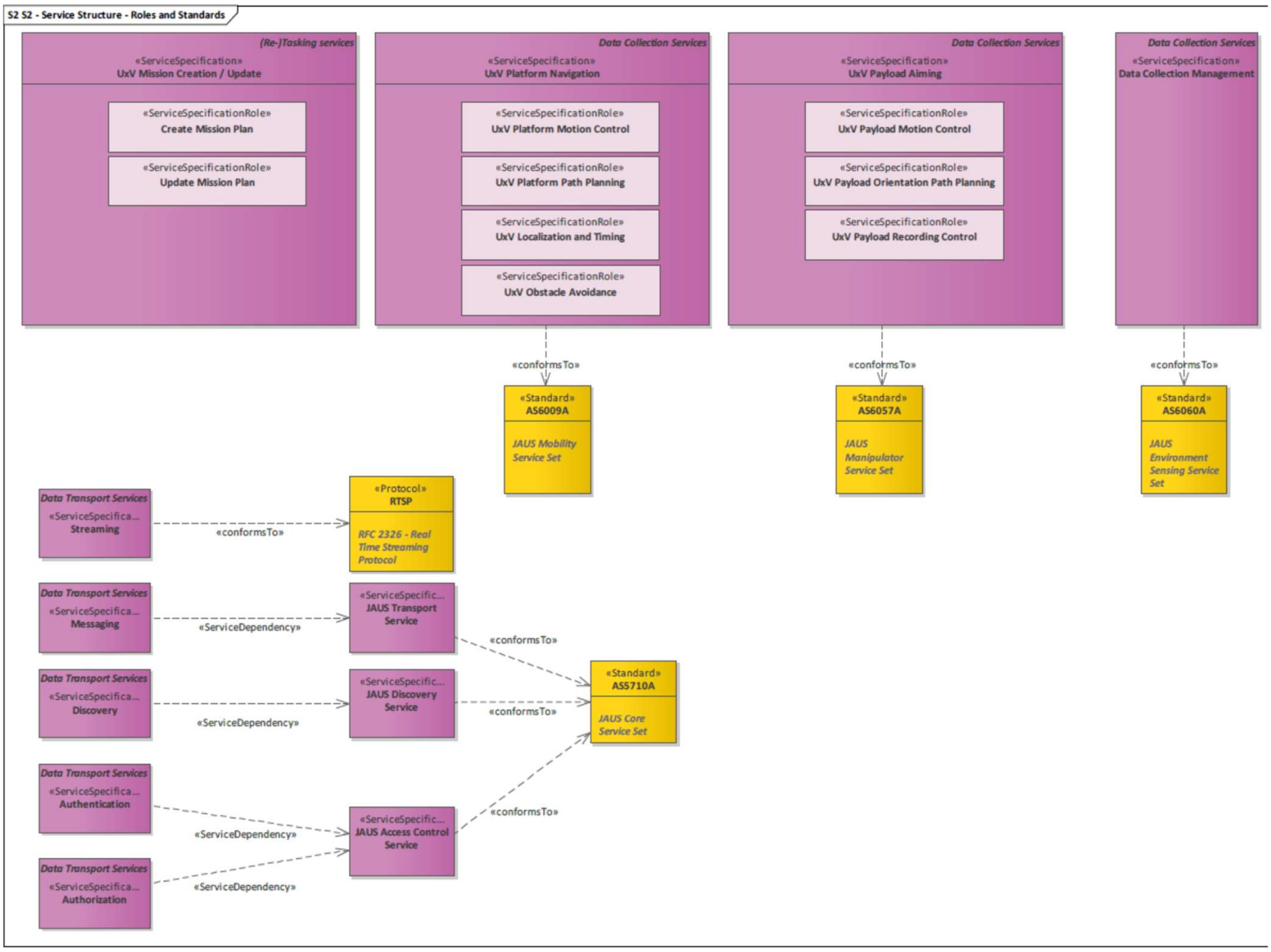

Figure 7: S2 – Service Structure – Roles and Standards

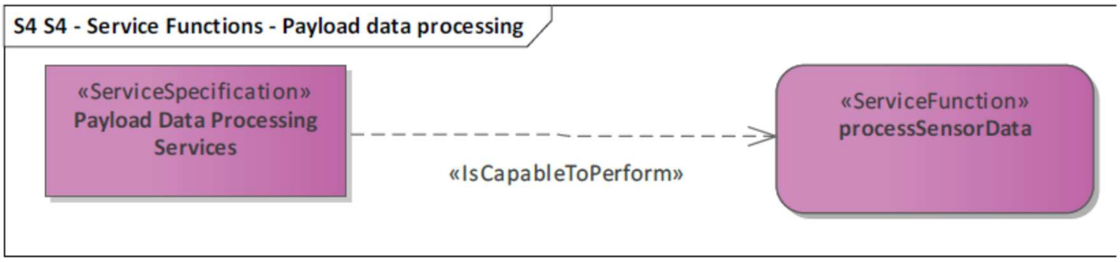

Figure 8: S2 – Service Functions – Payload data processing

V. Conclusion and Future Work

The paper presents the architecture framework used for developing an Interoperability Open Architecture to enable the use of heterogeneous unmanned systems in a variety of combinations, including their usage as an autonomously acting team, the Capability views of the Concepts subject and the Service Specification views.

The full architecture, including also the views of the other subjects, Logical Specifications, Physical Resource Specifications, and Architecture Foundation, as highlighted in Figure 1 have been also developed and documented in [19].

The developed architecture will enable manufacturers of unmanned systems to develop those in accordance with respective standards and ensuring interoperability of payloads and carriers as well as control stations, capable of simultaneously controlling multiple heterogeneous UxS as well as responding flexibly to changes and requirements in relation to other control units.

Future work will focus on a demonstration of the effectiveness of the Interoperability Open Architecture within a demonstration and simulated scenario and the development of a standardisation roadmap for future UxS standards.

Acknowledgment

This project has received funding from the European Union's Preparatory Action on Defence Research under grant agreement No 884270 INTERACT PADR-US-03-2019. The paper reflects only the author's view. The European Defence Agency is not responsible for any use that may be made of the information it contains.

REFERENCES

[1] P. Chrobocinski, E. Makri, N. Zotos, C. Stergiopoulos and G. Bogdos, "'DARIUS project: Deployable SAR integrated chain with unmanned systems'", 2012 Int. Conf. Telecommun. Multimedia TEMU 2012, pp. 220-226, 2012

[2] Mario Monteiro Marques, R. Parreira, V. Lobo, A. Martins, A. Matos, N. Cruz, et al., "Use of multi-domain robots in search and rescue operations-Contributions of the ICARUS team to the euRathlon 2015 challenge" in MTS/IEEE OCEANS 2016, Shangai, pp. 1-7, 2016.

[3] Rodolfo Santos Carapau, Alexandre Valério Rodrigues, Mario Monteiro Marques, Victor Lobo and Fernando Coito, "Interoperability of Unmanned Systems in Military Maritime Operations: Developing a controller for unmanned aerial systems operating in maritime environments", *MTS/IEEE OCEANS 2017*, pp. 1-7, 2017.

[4] Daniel Serrano, "Introduction to JAUS for Unmanned Systems Interoperability – Joint Architecture for Unmanned Systems –", Department of Intelligent Systems, ASCAMM Technology Center, Parc Tecnològic del Vallès, Av. Universitat Autònoma, 23, 08290 Cerdanyola del Vallès SPAIN.

[5] Riccardo Costanzi et al., "Interoperability Among Unmanned Maritime Vehicles: Review and First In-field Experimentation", Frontiers in Robotics and AI, Vol. 7, 2020

[6] DeVito, Marco J., Wood, Erik and Frazier, Tim. "Standardization and Interoperability of Small Unmanned Aircraft Systems (sUAS) for Disaster Management" Journal of Homeland Security and Emergency Management, vol. 19, no. 2, 2022, pp. 175-203. https://doi.org/10.1515/jhsem-2021-0028

[7] Curtis L. Blais, "Unmanned Systems Interoperability Standards", Naval Postgraduate School, Monterey, California, 2016, https://api.core.ac.uk/oai/oai:calhoun.nps.edu:10945/50386

[8] Mathiassen, K.,Schneider, F.E., Bounker, P., Tiderko, A., De Cubber, G., Baksaas, M.,Glówka, J., Kozik, R., Nussbaumer, T., Röning, J., Pellenz, J. and Volk, A. (2021) "Demonstrating interoperability between unmanned ground systemsand command and control systems", Int. J. Intelligent Defence SupportSystems, Vol. 6, No. 2, pp.100–129 (2021) doi: 10.1504/IJIDSS.2021.10037444.

[9] STANAG 4586. Standard Interfaces of UA Control System (UCS) for NATO UA Interoperability, Edition 4, April 2017, AEP-84, Edition A.

[10] J. T. Platts, M. L. Cummings and R. J. Kerr; "Applicability of STANAG 4586 to Future Unmanned Aerial Vehicles", AIAA 2007-2753, doi: 10.2514/6.2007-2753

[11] A. Koubâa, A. Allouch, M. Alajlan, Y. Javed, A. Belghith and M. Khalgui, "Micro Air Vehicle Link (MAVlink) in a Nutshell: A Survey," in IEEE Access, vol. 7, pp. 87658-87680, 2019, doi: 10.1109/ACCESS.2019.2924410

[12] Marty, Joseph A. "Vulnerability analysis of the mavlink protocol for command and control of unmanned aircraft". AIR FORCE INSTITUTE OF TECHNOLOGY WRIGHT-PATTERSON AFB OH GRADUATE SCHOOL OF ENGINEERING AND MANAGEMENT, 2013

[13] Robot Operating System ROS. https://www.ros.org/about-ros

[14] D. Bore, A. Rana, N. Kolhare and U. Shinde, "Automated Guided Vehicle Using Robot Operating Systems," 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019, pp. 819-822, doi: 10.1109/ICOEI.2019.8862716

[15] ROS Military ROS-M. https://rosmilitary.org

[16] NATO Architecture Framework v4 Document Version 2020.09 - https://www.nato.int/nato_static_fl2014/assets/pdf/2021/1/pdf/NAFv4_2020.09.pdf

[17] NATO Architecture Framework v4 Modeling Guidelines for use of the UAF DMM – https://www.nato.int/nato_static_fl2014/assets/pdf/pdf_2019_04/20190404_AC_322-N-201-0046_ENG-UAF-Guide.pdf

[18] NAFv4 MDG Extensions for Enterprise Architect, Example, Documentation and Presentation Slides – https://owncloud.sparxsystems.eu/index.php/s/DWnoo208xikYjr5

[19] INTERACT Deliverable D3.4 Interoperability Open Architecture. Internal Report.