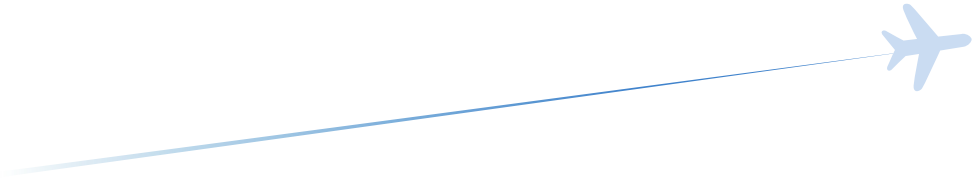

GOF2.0 D2.4 – Appendix B Operation Plan Service Specification

Deliverable ID:	**D2.4-B**
Dissemination Level:	**PU**
Project Acronym:	**GOF 2.0**
Grant:	**101017689**
Call:	**H2020-SESAR-2020-1 VLD Open 2**
Topic:	**U-space capabilities and services to enable Urban Air Mobility**
Consortium Coordinator:	**Lennuliiklusteeninduse Aktsiaselts (EANS)**
Edition Date:	**31 October 2022**
Edition:	**00.00.04**
Template Edition:	**02.00.02**

Founding Members

EUROPEAN UNION EUROCONTROL

SESAR JOINT UNDERTAKING

Authoring & Approval

Authors of the document

Name/Beneficiary	Position/Title	Date
Shengnan Gao / EHANG	WP2	26.4.2021
Hui Zeng / EHANG	WP2	26.4.2021
Juha Lindstedt / Aviamaps	WP2	26.4.2021
Jose Ignacio Rodriguez / EHANG	WP2	26.4.2021
Ada Lu / EHANG	WP2	26.4.2021
Yuhang Yun / EHANG	WP2	26.4.2021
Piotr Luboński / PSNC	WP2	26.4.2021
Damian Soliwoda / PSNC	WP2	26.4.2021
Sven Jürgenson / Threod	WP2	26.4.2021
Piotr Dybiec / DroneRadar	WP2	26.4.2021
Thomas Wana / Dimetor	WP2	26.4.2021
Tanel Järvet / CAFA Tech	WP2	26.4.2021
Parmentier Remy / Vaisala	WP2	26.4.2021
Piotr Szymaniak / PSNC	WP2	26.4.2021
Pawel Korzec / DroneRadar	WP2	26.4.2021
Lukasz Gorny-Zajac / Droneradar	WP2	26.4.2021
Gokul Srinivasan / Robots Expert	WP2	26.4.2021
Thomas Lutz / Frequentis	WP2 Lead	26.4.2021
Hubert Künig / Frequentis	WP2	26.4.2021
Peter Cornelius / Frequentis	WP2	26.4.2021
Gregor Mogeritsch / Frequentis	WP2	26.4.2021

Reviewers internal to the project

Name/Beneficiary	Position/Title	Date
Jonas Stjernberg / Robots Expert	WP3 Lead	03.11.2022
Maria Tamm / EANS	Project Coordinator	03.11.2022
Pawel Korzec / DroneRadar	WP2	03.11.2022

Approved for submission to the SJU By - Representatives of beneficiaries involved in the project

Name/Beneficiary	Position/Title	Date

Rejected By - Representatives of beneficiaries involved in the project

Name/Beneficiary	Position/Title	Date

Document History

Edition	Date	Status	Author	Justification
00.00.01	2019		GOF 1 Project Partners (Sebastian Babiarz / Airmap, Teodor Todorov / Airmap, Rupert Benbrook / Altitude Angel, Phil Binks / Altitude Angel, Chris Forster / Altitude Angel, Simon Wynn-Mackenzie/ Altitude, Angel Alkula Sami / Ansfinland, Tanel Jarvet / Cafatech, Vello Müürsepp / EANS, Heidi Himmanen / Ficora, Dan Davies / Fleetonomy, Peter Cornelius / Frequentis, Thomas Lutz / Frequentis, Harald Milchrahm / Frequentis, Jonas Stjernberg / Robots Experts, Charlotte Kegelaers / Unifly, Ronni Winkler Østergaard / Unifly, Andres Van Swalm / Unifly	
00.00.02	18.03.2021	draft	WP2 Partners	Enhance and update
00.00.03	26.04.2021	released	WP2 Partners	Release
00.00.04	03.11.2022	released	WP2 Partners	Update

GOF 2.0

GOF2.0 INTEGRATED URBAN AIRSPACE VLD

This Updated Service Specification is part of a project that has received funding from the SESAR Joint Undertaking under grant agreement No 101017689 under European Union's Horizon 2020 research and innovation programme.

1.1 Abstract

This specification introduces a service of a Common Information Service (CIS) which ensures interoperability and hence transparent and reliable information flow between the stakeholders in an operational U-space environment. In accordance with ICAO SWIM, represents an Information Exchange Service.

This document describes one of these Bridge Services, the Operation Plan Exchange service in a logical, technology-independent manner.

Table of Contents

List of Tables

List of Figures

2 Introduction

2.1 Purpose of the document

In accordance with according to the guidelines given in [3], this document describes the Operation Plan Exchange service for the GOF USPACE project on a logical technology-independent manner, that is:

- the operational and business context of the service
  - requirements for the service (e.g., information exchange requirements)
  - involved nodes: which operational components provide/consume the service
  - operational activities supported by the service
  - relation of the service to other services
- the service description
  - service interface definitions
  - service interface operations
  - service payload definition
  - service dynamic behaviour description
- service provision and validation aspects

Furthermore, this document clearly defines the version of the service.

2.2 Scope

This document describes the Operation Plan Exchange service for the GOF USPACE project.

The Operation Plan Exchange service provides a means for the operational nodes of the GOF USPACE project to share their UTM operation plans and make them available for further processing.

The Operation Plan Exchange service furthermore provides a means for the operational nodes of the GOF USPACE project to consume UTM operation plans from the U-space participants for further processing.

Finally, the Operation Plan Exchange service provides an interface allowing operational nodes to manipulate drone operation plans by interacting with the service provider in order to request authorization, activation, etc.

2.3 Intended readership

This service specification is intended to be read by service architects, system engineers and developers in charge of designing and developing an instance of the Operation Plan Exchange service.

Furthermore, this service specification is intended to be read by enterprise architects, service architects, information architects, system engineers and developers in pursuing architecting, design and development activities of other related services.

2.4 Background

2.4.1 EUROCONTROL Concept of Operations for U-space (CORUS)

EUROCONTROL CORUS [4] elaborates in 5.1 Flight planning and derived services reporting as follows.

"U2 brings flight planning and many derived services. As mentioned in section 4, for airspaces where flight plans are required, the aim is to have a flight plan from every aircraft, drone or manned. This will allow a something like a complete picture to be built in advance. "Something like" because flight plans for drones, like for manned aircraft, will contain some elements of uncertainty, or "flexibility". U2 brings flight planning and many derived services. As mentioned in section 4, for airspaces where flight plans are required, the aim is to have a flight plan from every aircraft, drone or manned. This will allow a something like a complete picture to be built in advance. "Something like" because flight plans for drones, like for manned aircraft, will contain some elements of uncertainty, or "flexibility".

[…]
Drone flight plans will not be the same as the flight plan defined in ICAO doc 4444 used by manned aviation today. Several factors push for this. There are rules associated with the ICAO flight plan that simply cannot work for small drones, such as a requirement to fly via published points. The ICAO flight plan does a poor job of describing a 4D trajectory and there is no reason to inflict this problem on ourselves in the brave new world of drone flight planning.
[…]
The basic contents of the drone flight plan will be the same as the ICAO flight plan: who flies what, where and when. Like the ICAO flight plan, the drone flight plan is a legal commitment to fly and stakes a claim on a limited public good – the airspace. Submitting a flight plan is also a commitment by the drone operator to meet the obligations associated with flying, which would include factors like having a safe aircraft and a competent pilot and committing to operate the flight safely, but also committing to meet the costs (if any) associated with flying, to have public liability insurance and so on.
There will need to be a nominated actor to whom drone flight plans are sent, probably with a mandate from the civil aviation authority for the country. Flight plans will be sent electronically, ideally from a tool used by a drone operator that makes flight plan creation easy and integrates into the task of creating the plan to be loaded into the drone itself. The basic process will consist in its simplest form as:

- *Drone operator submits flight plan*
- *Flight plan is checked by U-space*
- *Drone operator gets a response*

- *If valid, the flight plan data is added to the set of current flight plans in the flight planning*
- *system and the operator can load it into the drone*
- *If invalid, the drone operator reads the remarks, changes the plan and resubmits.*

The 'check' of the flight plan by the U-space includes:

- *Some parts of Pre-tactical Geo-fencing*
- *Strategic Conflict Resolution*
- *In U3, Dynamic Capacity Management*

The flight plan management service supports:

- *Flight plan preparation / optimisation*
- *Pre-tactical Geo-fencing and Tactical Geo-fencing*
- *Strategic Conflict Resolution*
- *Procedural interface with ATC*
- *Collaborative interface with ATC*
- *Tracking*
- *Monitoring*
- *Traffic Information*
- *Emergency Management*
- *Legal Recording*
- *Incident / Accident reporting*
- *Operations Management*
- *Safety Management*
- *Dynamic Capacity Management*
- *Digital Logbook"*

2.4.2 SESAR-JU

The European Commission identifies an increasing demand for a non-segregated use of airspace which is being driven by a rapidly growing market of Very-Low-Level (VLL) airspace users, most of which are expected to be drones.

Via the Roadmap for the safe integration of drones into all classes of airspace [11], within the European ATM Masterplan [12], the European Commission seeks to ensure that this rapid growth of airspace use happens in a safe and controlled manner.

SESAR develops the required concepts and demonstrations for this process to happen. The roadmap [1], in alignment with ICAO recommendations, identifies three phases for the integration, from which SESAR derives the four U-space service blocks presented in the U-space blueprint [13],

U1 U-space foundation services,

U2 U-space initial services,

U3 U-space advanced services, and

U4 U-space full services.

These stages reflect the anticipated quick growth of demand for U-space services. The state of the art is being validated throughout Europe via several Very Large Demonstrator (VLD) projects such as the GOF USPACE project.

During the U1 phases, SESAR expects drones capable to supply their position via telemetry. The U1 and U2 is anticipated to provide tracking capabilities and services.

2.4.3 1.1.8 Efficient, safe and sustainable traffic at sea (EfficienSea2)

The design method and terminology builds on experience from the EfficienSea2 project [14], [15].

2.5 Glossary of terms

Term	Definition
External Data Model	Describes the semantics of the domain (or a significant part thereof) by defining data structures and their relations. This could be at logical level (e.g., in UML) or at physical level (e.g., in XSD schema definitions), as for example standard data models.
Message Exchange Pattern	Describes the principles how two different parts of a message passing system (in our case: the service provider and the service consumer) interact and communicate with each other. Examples: In the Request/Response MEP, the service consumer sends a request to the service provider in order to obtain certain information; the service provider provides the requested information in a dedicated response. In the Publish/Subscribe MEP, the service consumer establishes a subscription with the service provider in order to obtain certain information; the service provider publishes information (either in regular intervals or upon change) to all subscribed service consumers.
Operational Activity	An activity performed by an operational node. Examples of operational activities are: Route Planning, Route Optimization, Logistics, Safety, Weather Forecast Provision, …
Operational Model	A structure of operational nodes and associated operational activities and their inter-relations in a process model.
Operational Node	A logical entity that performs activities. Note: nodes are specified independently of any physical realisation. Examples of operational nodes are: Control Center, Authority, Weather Information Provider, …

Service	The provision of something (a non-physical object), by one, for the use of one or more others, regulated by formal definitions and mutual agreements. Services involve interactions between providers and consumers, which may be performed in a digital form (data exchanges) or through voice communication or written processes and procedures.
Service Consumer	A service consumer uses service instances provided by service providers.
Service Data Model	Formal description of one dedicated service at logical level. The service data model is part of the service specification. Is typically defined in UML and/or XSD. If an external data model exists (e.g., a standard data model), then the service data model shall refer to it: each data item of the service data model shall be mapped to a data item defined in the external data model.
Service Design Description	Documents the details of a service technical design (most likely documented by the service implementer). The service design description includes (but is not limited to) a service physical data model and describes the used technology, transport mechanism, quality of service, etc.
Service Implementation	The provider side implementation of a dedicated service technical design (i.e., implementation of a dedicated service in a dedicated technology).
Service Implementer	Implementers of services from the service provider side and/or the service consumer side.
Service Instance	One service implementation may be deployed at several places by same or different service providers; each such deployment represents a different service instance, being accessible via different URLs.
Service Instance Description	Documents the details of a service implementation (most likely documented by the service implementer) and deployment (most likely documented by the service provider). The service instance description includes (but is not limited to) service technical design reference, service provider reference, service access information, service coverage information, etc.
Service Interface	The communication mechanism of the service, i.e., interaction mechanism between service provider and service consumer. A service interface is characterised by a message exchange pattern and consists of service operations that are either allocated to the provider or the consumer of the service.
Service Operation	Functions or procedure which enables programmatic communication with a service via a service interface.

Service Physical Data Model	Describes the realisation of a dedicated service data model in a dedicated technology. This includes a detailed description of the data payload to be exchanged using the chosen technology. The actual format of the service physical data model depends on the chosen technology. Examples may be WSDL and XSD files (e.g., for SOAP services) or swagger (Open API) specifications (e.g., for REST services). If an external data model exists (e.g., a standard data model), then the service physical data model shall refer to it: each data item of the service physical data model shall be mapped to a data item defined in the external data model. In order to prove correct implementation of the service specification, there shall exist a mapping between the service physical data model and the service data model. This means, each data item used in the service physical data model shall be mapped to a corresponding data item of the service data model. (In case of existing mappings to a common external (standard) data model from both the service data model and the service physical data model, such a mapping is implicitly given.)
Service Provider	A service provider provides instances of services according to a service specification and service instance description. All users within the domain can be service providers, e.g., authorities, organizations (e.g., meteorological), commercial service providers, etc.
Service Specification	Describes one dedicated service at logical level. The Service Specification is technology-agnostic. The Service Specification includes (but is not limited to) a description of the Service Interfaces and Service Operations with their data payload. The data payload description may be formally defined by a Service Data Model.
Service Specification Producer	Producers of service specifications in accordance with the service documentation guidelines.
Service Technical Design	The technical design of a dedicated service in a dedicated technology. One service specification may result in several technical service designs, realising the service with different or same technologies.
Service Technology Catalogue	List and specifications of allowed technologies for service implementations. Currently, SOAP and REST are envisaged to be allowed service technologies. The service technology catalogue shall describe in detail the allowed service profiles, e.g., by listing communication standards, security standards, stacks, bindings, etc.

Spatial Exclusiveness	A service specification is characterised as “spatially exclusive”, if in any geographical region just one service instance of that specification is allowed to be registered per technology. The decision, which service instance (out of a number of available spatially exclusive services) shall be registered for a certain geographical region, is a governance issue.

Table 1: Glossary of terms

2.6 List of Acronyms

Acronym	Definition
API	Application Programming Interface
MEP	Message Exchange Pattern
NAF	NATO Architectural Framework
REST	Representational State Transfer
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SSD	Service Specification Document
UML	Unified Modelling Language
URL	Uniform Resource Locator
WSDL	Web Service Definition Language
XML	Extendible Mark-up Language
XSD	XML Schema Definition

Table 2: List of acronyms

3 Service Identification

The purpose of this chapter is to provide a unique identification of the service and describe where the service is in terms of the engineering lifecycle.

Name	**OperationPlanExchange Service**
ID	**urn:gof:services:OperationPlanExchangeService**
Version	**2.0**
Description	**An information exchange service which provides operation plan information**
Keywords	**Operation Plan, Mission Planning, Contingency Plan, Operation Volume, Operation Trajectory**
Architect(s)	**2021-today The GOF 2.0 Project Consortium** **2020-2021 The Frequentis Group** **2018-2020 The GOF U-Space Project Consortium**
Status	**Provisional**

Table 3: Service Identification

4 Operational Context

This section describes the context of the service from an operational perspective.

4.1 Functional and Non-functional Requirements

The table below lists applicable existing requirements for the Operation Plan Exchange service.

Requirement Id	Requirement Name	Requirement Text	References
[R-1]	Common Situational Awareness	At all times, all U-space participants shall operate on the same common set of data, during pre-flight planning stages as well as during all stages of flight operations.	CORUS [4], 4.1.1.2 Amber airspace;B1-RPAS [9];CEF-SESAR-2018-1 [1], Objective O5
[R-2]	Basis for Open Market	The U-space concept shall be designed such as to ensure a well-established line of authority while at the same time ensuring that an open market for VLL services may develop	SESAR Drone Roadmap [11], Foreword, 4.1 and 4.2;U-space Blueprint [13], Benefits to European society and economy;CEF-SESAR-2018-1 [1], Table 8 – Key Challenges
[R-3]	Interoperability	There shall be an implementation of a Flight Information Management System (FIMS) which ensures that, at all times, emerging unmanned traffic management systems and existing technologies from manned operations can exchange any data required to support such common situational awareness, be it for drone operations in areas where established ATC procedures apply, or in zones outside established ATC.	ICAO Doc 10039 [2];[R-2];CEF-SESAR-2018-1 [1], Objective O6;CEF-SESAR-2018-1 [1], Table 8 – Key Challenges

[R-4]	Standard Protocols	Standard communication protocols shall hence be used where available, and such standard protocols be developed otherwise, in order to ensure the lowest level of obstruction for an open VLL airspace use market to develop.	[R-2];SESAR Drone Roadmap [11], 3.5, section 'Standards';CEF-SESAR-2018-1 [1], Table 8 – Key Challenges
[R-5]	Open Interfaces	Any interface and protocol hence must be openly defined and its definition be freely accessible in order to ensure the lowest level of obstruction for an open VLL airspace use market to develop.	[R-2];CEF-SESAR-2018-1 [1], Table 8 – Key Challenges
[R-6]	SWIM	The implementation of a Flight Information Management System (FIMS) shall be based on an ICAO SWIM-compliant architecture.	[R-3];CEF-SESAR-2018-1 [1], 5.3.4 Overall approach and methodology

[R-7]	Latency	Under no operational circumstance, the processing of position data may add significant latency to the overall detection-to-display latency of position data. In particular, The processing latency added by the processing of positional data shall never exceed 10 per cent of the maximum value of the corresponding value permitted for the entire ATM automation system. The processing latency and delay added by the processing of positional data should not exceed 1 per cent of the maximum value of the corresponding value permitted for the entire ATM automation system. The maximum value for latency and delay is the minimum of the values defined by the ATM system performance requirements by EUROCONTOL and the FAA; for a 3 NM minimal separation, this is 2.2 s, for a 5 NM separation, 2.5 s.	[17], tables in the Executive Summary, [16], 3N_C-R8 and 5N_C-R8

Table 4: Requirements for the Traffic/Telementry Service

4.2 Other Constraints

4.2.1 Relevant Industrial Standards

4.2.1.1 ICAO SWIM

The System Wide Information Management (SWIM, [2]) complements human-to-human with machine-to-machine communication, and improves data distribution and accessibility in terms of quality of the data exchanged. The SWIM Concept addresses the challenge of creating an "interoperability environment" which allows the SWIM IT systems to cope with the full complexity of operational information exchanges. The SWIM environment shifts the ATM information architecture paradigm from point-to-point data exchanges to system-wide interoperability.

4.2.2 Operational Nodes

A typical U-space flight goes through several stages, starting strategic-tactically, pre-flight, from Strategic Planning, over to Pre-Tactical Planning, to Tactical Planning. Then, tactical-operationally it enters into the actual in-flight stages from Departure, over to In-Flight, and, finally Arrival. Further post-flight stages may evaluate the results from the data produced during the prior stages.

The Operation Plan Exchange service primarily is relevant during the Pre-Tactical planning stages of a U-space flight during which the relevant stakeholders generate flight planning information data as well as requests for authorization and activation, which we convey via the Operation Plan Exchange service.

Operational nodes which may provide data for the Operation Plan Exchange service include the following ones.

Operational Node	Remarks
USSP	
CISP	
UTM Service Provider	
Flight Information Management System	

Table 5: Operational Nodes providing the Operation Plan Exchange service

Operational nodes which may consume the Operation Plan Exchange service include the following ones.

Operational Node	Remarks
Flight Information Management System	
Information Display	
Drone Operator	
ATM Service Provider	
Legal Recorder	

Table 6: Operational Nodes consuming the Operation Plan Exchange service

4.2.3 Operational Activities

Operational activities supported by the Operation Plan Exchange service include the following ones.

Phase	Operational Activity	Remarks

Pre-flight	Plan	This is the main phase where the Operation Plan Exchange service is used.
In-Flight	Cruise	In the in-flight phase the Operation Plan Exchange service is enhanced by the Drone Flight Exchange service.
Post-Flight	Report	

Table 7: Operational Activities supported by the Operation Plan Exchange service

4.3 Service Interfaces

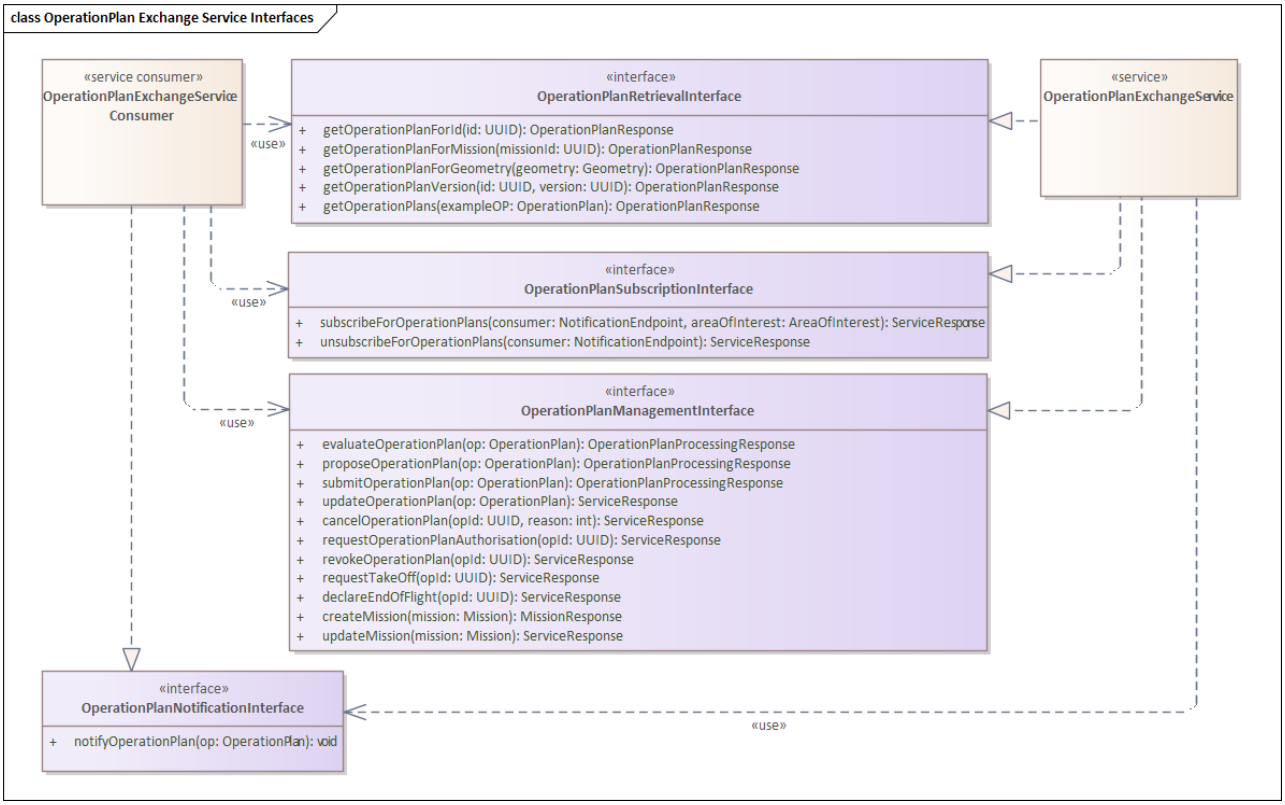

Figure 1: OperationPlanExchangeService Interface Definition diagram

ServiceInterface	Role (from service provider point of view)	ServiceOperation
OperationPlanRetrievalInterface	Provided	getOperationPlanForId getOperationPlanVersion getOperationPlansForMission getOperationPlansForGeometry getOperationPlans

OperationPlanSubscriptionInterface	Provided	subscribeForOperationPlans unSubscribeForOperationPlans
OperationPlanNotificationInterface	Required	notifyOperationPlan
OperationPlanManagementInterface	Provided	evaluateOperationPlan proposeOperationPlan submitOperationPlan updateOperationPlan cancelOperationPlan requestOperationPlanAuthorisation revokeOperationPlan requestTakeOff declareEndoOfFlight createMission updateMission

Table 8: Service Interfaces

5 Service Data Model

This section describes the information model, i.e., the logical data structures to be exchanged between providers and consumers of the service.

5.1 Overview

The OperationPlan exchange service transfers information about Operation Plans and associated data. The central part of the data model for this service is the OperationPlan structure, which collects various data items of various categories:

- identification;
- information about geographical and timely dimension of the operation plan:
  - rough trajectory by a sequence of operation volumes including timely validity of these volumes;
  - optionally a more finegrained 4D trajectory, defined by a list of waypoints (trajectory elements).
- information about contingency plans for the case of unplanned events:
  - alternative actions;
  - alternative landing or loitering areas;
  - alternative volumes for emergency landings;
  - expected endurance
- information about the current state of the operation plan:
  - current state;
- history of the operation plan:
  - history of past updates to the operation plans;
  - history of state transitions.
- priority declarations;
- information about used devices: drone identification and registration information;
- flight related information;
- administrative information, such as contact details, public name and description, comments, etc.;
- additional information about the operation mission.

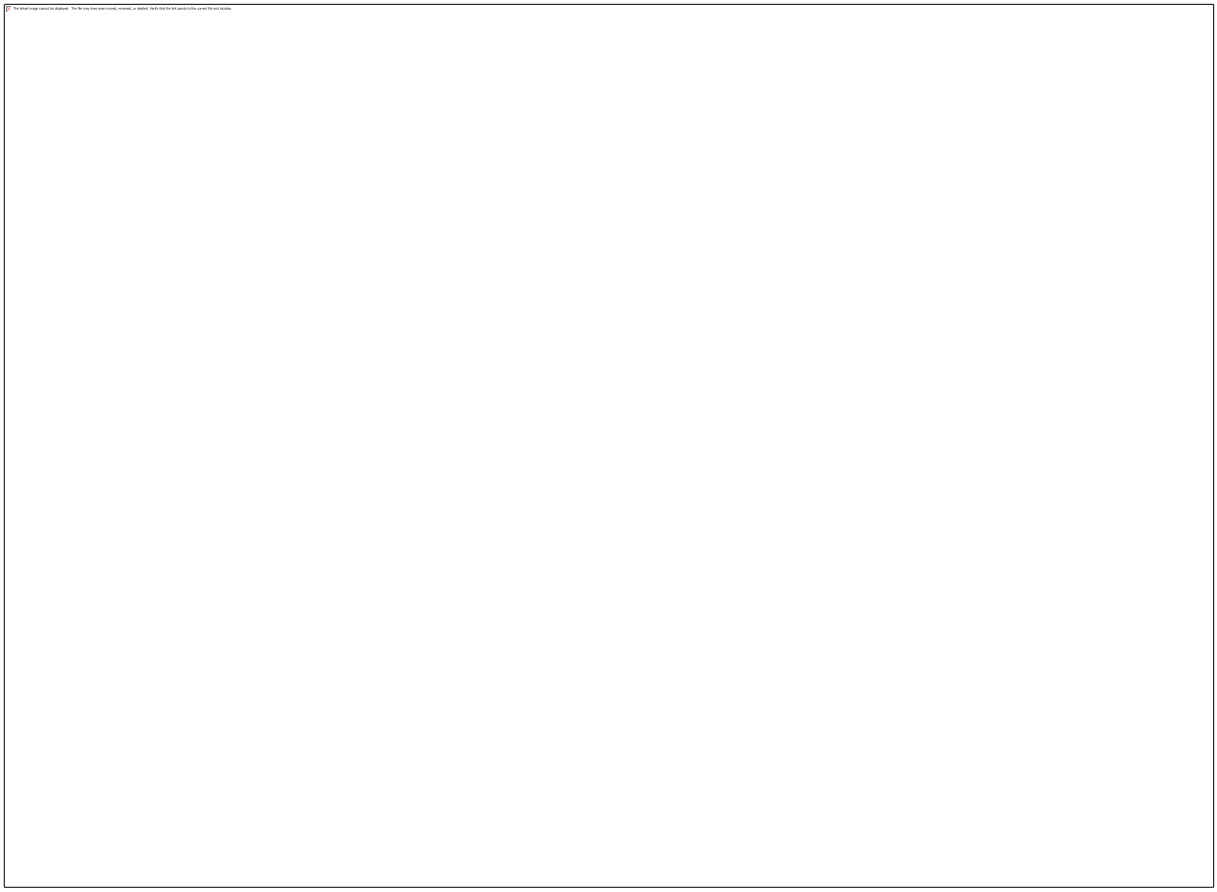

Figure 2: Operation Plan Services - Base Data Model diagram

5.2 The OperationPlan Data Structure

OperationPlan is the central part of the OperationPlanExchange service data model. As such, it acts as a composite of plenty other structures.

Property	Type	Multiplicity	Description	Note
operationPlanId	UUID	1	Globally unique identifier of this operation plan.	
version	UUID	1	Unique identifier of the version of this operation plan.	This is needed for the history of the OP. See **previousVersion** reference .

previousVersion	Reference to OperationPlan	0..*	Reference to the previous version of this operation plan.	This reference builds up the history of the OP. Upon every change, a new version of the OP shall be created; this reference points to the previous version. This reference can simply be realized by storing the version UUID of the previous version. Alternatively, it could be realized by storing an ordered linked list of UUIDs, including the whole history.
aircraftComment	String	0..1	Informative text about the aircraft. Not used by the UTM System. Only for human stakeholders.	
state	OperationPlanState	1	The current state of the operation. Must be maintained by the USS.	
operator	String	1	Operator identification.	
submitTime	DateTime	1	Time that this operation plan was first announced to the USS Network in any way.	The submitTime value MUST remain constant for each recipient of the announcement since this value is potentially part of a signature of the operation plan in some cases.

updateTime	DateTime	1	A timestamp set by the USS any time the state of the operation plan is updated within the USS Network. An update may be minor or major, but if/when the operation plan is shared in the USS Network, updateTime must reflect the time that update was provided. The updateTime value MUST be constant for each update data exchange. This field is set and maintained by the USS managing the operation and is communicated to other USSs.	When the operation plan is announced for the first time, updateTime MUST be equal to submitTime. When an operation plan is modified (updated), updateTime MUST be greater than submitTime.
typeOfOperation	OperationType	1	Indicates the type of operation.	

swarmSize	int	1	Number of drones flying as a swarm. If >1, this indicates that this operation plan represents a swarm.	Corus: "U-space considers formation flights and swarms as being collections of aircraft that do not need to be separated by U-space." "A swarm is considered by U-space to be a single, solid object. U-space will not attempt to pass another flight through a swarm.... A swarm will have a single operation plan and this plan will include dimensions for the swarm. Swarms may be prohibited in some volumes."
formationId	string	0..1	Designator for a formation flight. This string is supposed to have the same value for all OPs taking part in a formation flight.	Corus: "Drone formation flights are individual operation plans that are linked, rather than single plans for multiple aircraft.
formationOpIds	UUID	0..*	References to other OPs belonging to the same formation.	Corus: "Drone formation flights are individual operation plans that are linked, rather than single plans for multiple aircraft.
minContOpTime	TimeDuration	0..1	Minimum continuous operation time.	Minimum acceptable time of the continuous operation to recognize the operation as succesfull.
atsInstruction	string	0..1	Optional instructions provided by ATS during approval process.	
closureReason	EnumClosureReasonType	0..1	Indication of the reason for closing the operation plan.	

operationVolumes	OperationVolume	0..*	The actual geographical information for the operation, defined by at least one operation volume.	Each operation volume MUST have non-zero 4D volume (i.e. each of the 4 dimensions much have a valid value). Volume intersection must pass the following checks: a. When ordered by ordinal values, a succeeding operation volume must have a 2D or 3D intersection in 3D space with its immediately preceding operation volume. Note that a 2D intersection in 3D space implies two volumes that "touch" and the intersection has 2D area. Sharing just an edge would not qualify. 3D volumes that don't touch at all would not qualify, even if they would intersect when projected into 2D space (e.g. if "looking down" on the two volumes). b. When ordered by ordinal values, a succeeding operation volume must have a non-negative temporal intersection with its immediately preceding operation volume (Note we'd calculate this by t1-t2 where t1 is the preceding operation volume end time and t2 is the succeeding operation volume start time.). c. When ordered by ordinal values, a succeeding operation volume must have either a non-zero volume (3D) intersection OR a positive temporal intersection. (Note this is a logical "OR" so it may have both intersection types...

				i.e. it is not an "exclusive OR"). Each spatial dimension of an operation volume's bounding box must have length less than 6000ft (value TBD). This is a sanity check against excessively large volumes. Need to be careful here as there may be legitimate use cases wherein large volumes are required/allowed, but we want to encourage efficient planning and protect against misuse of the shared airspace. The planned duration of an operation volume must be less than 120 minutes (value TBD). Again, need to be careful to not damage legitimate use cases, but need to protect against misuse/poor planning. For long duration missions, it may be reasonable to have them replan or to have volumes with the same geography and long time values that slightly intersect. The start time of an operation volume other than the first in the array must be greater than or equal to the start time of its immediately preceding operation volume. operation volumes may by omitted if an **operationTrajectory** is provided.

operationTrajectory	OperationTrajectory	0..1	Detailed planned trajectory, including uncertainties.	If no **operationTrajectory** is provided, then the **operationVolumes** are mandatory.
contingencyPlans	ContingencyPlan	0..*	An array of contingency plans wherein this operation may land if needed/required during operation. Aids in planning and communication during the execution of a contingency.	
controllerLocation	Geometry	1	Two dimensional location (geographical point) of the controller. This is the actual drone pilot location.	
gcsLocation	Geometry	0..1	Two dimensional location (geographical point) of the ground control station. This is the optional control station location for the (potential) cases where the pilot is not co-located with the GCS.	
takeoffLocation	Geometry	0..1	Two dimensional location (geographical point) of the takeoff point.	

landingLocation	Geometry	0..1	Two dimensional location (geographical point) of the landing target.	
uasRegistrations	UasRegistration	1..*	The registration data for the vehicle(s) to be used in this Operation. Note that this is an array to allow for future operations involving multiple vehicles (e.g. 'swarms' or tandem inspections).	The uasRegistrations array MUST NOT be used as a list of potential vehicles for this Operation. If the vehicle data changes prior to an Operation, an update to the plan may be submitted with the updated vehicle information. Providing multiple uasRegistrations in this manner implies that all vehicles will conform to the provided operation volumes.
flightDetails	FlightDetails	0..*	More details related to the flight.	
priority	Priority	1	describes the priority of this operation operation.	If necessary, this priority may be overruled by a priority definition in the OperationVolume (for example, if the high prio operation is limited to a subset of the OP's OperationVolumes).
contactDetails	ContactDetails	0..1	Contact information for this operation.	
publicInfo	PublicInformation	0..1	Additional information for public disclosure.	

mission	Mission reference	0..1	Reference to a mission.	A mission allows collecting common information for flights related to each other. E.g., a series of flights to be executed for one common goal, e.g., power line inspections flights for one customer at one specific day.

Table 9: The OperationPlan data structure

5.3 The OperationPlanResponse Data Structure

OperationPlanResponse is used to carry the result of query-operations asking for o peration plans.

Depending on the operation result, it may contain zero, one or several operation plans.

Property	Type	Multiplicity	Description	Note
operationPlans	OperationPlan	0..*	Operation plan(s).	
< inherited >			**All properties inherited from ServiceResponse.**	**See common data types.**

Table 10: The OperationPlanResponse data structure

5.4 The OperationVolume Data Structure

OperationVolume is used to describe a portion of the actual geographical information for the operation. It includes a definition of the three-dimensional boundaries of a portion of airspace, associated with the time constraints when this portion is planned to be used. Additional flags indicate whether the portion of airspace is beyond the visual line of sight, or if there is a structure (e.g., building) in near vicinity.

Property	Type	Multiplicity	Description	Note
alias	String	0..1	Optional descripitve text.	

timeBegin	DateTime	1	Earliest time the operation will use the operation volume. It must be less than timeEnd.	timeBegin < timeEnd MUST be true.
timeEnd	DateTime	1	Latest time the operation will done with the operation volume. It must be greater than timeBegin.	timeBegin < timeEnd MUST be true.
actualTimeEnd	DateTime	0..1	Time that the operational volume was freed for use by other operations. Should be populated and stored by the USS.	actualTimeEnd MUST satisfy: actualTimeEnd > timeBegin whenever actualTimeEnd is not null.
isBVLOS	Boolean	1	Describes whether any portion of the operation volume is beyond the visual line of sight of the RPIC.	
isNearStructure	Boolean	0..1	Is this operation volume within 400' (value TBD) of a structure?	

ordinal	Integer	1	This integer represents the ordering of the operation volume within the set of operation volumes.	Need not be consecutive integers.
operationGeometry	Geometry	1	Three-dimensional geometry defining the operation volume.	The type of Geometry, in this case, must be a three-dimensional shape.
operationGeometryMinimumSeparation	Geometry	0..1	Three-dimensional geometry defining minimum separation around the operation volume.	
priority	Priority	0..1	Priority indication for the operation within this operation volume.	This priority overrules the priority definition in the OperationPlan.

Table 11: The OperationVolume data structure

5.5 The ContingencyPlan Data Structure

ContingencyPlan is the

Property	Type	Multiplicity	Description	Note
id	UUID	1	An identifier unique amongst the set of Contingencies for this operation.	

causes	ContingencyCauseType	1..*	Describes the cause(s) leading to this contingency plan.	
locationDescr	ContingenyLocationDescriptionType	1	Indication on where the contingency plan is described.	
responseAction	ContingencyResponseType	1	The type of contingency response.	LANDING: targeting the contingencyGeography, optionally heading through the contingencyExtraVolume. LOITERING: loiter at the contingencyGeography at the specified loiterAltitude. RETURN_TO_BASE: return to base as specified by the contingencyGeography, optionally heading through the contingencyExtraVolume. The USS may issue an update to the operation plan to support this maneuver. OTHER: details given in freeText property.
validTimeBegin	DateTime	1	Time that this location is expected to be first available.	

validTimeEnd	DateTime	1	Time that this location is expected to become unavailable.	
freeText	String	0..1	To be used for additional comments as needed.	For human use, not for automating any process.
relativePreference	Float	0..1	Optional numerical value that can be used in ranking the preference of this Contingency versus any other within the set of Contingency for this operation.	This may be thought of as a ranking of the potential landing sites with all other factors being held equal, though dynamic conditions will likely play a role in adjusting this ranking in real time by the USS or Operator. For example, one Contingency may be significantly further from the operation at a given time and, thus, would be less preferred than it might be otherwise. Further interpretation of this field is left to the operator and USS.
relevantOperationVolumes	Integer	1..*	This is an indicator that this particular ContingencyPlan is valid for use when the operation is active in any of the particular noted OperationVolumes.	In the planning stage of an operation, this array may be populated with ordinals that correspond to the ordinal values supplied with each OperationVolume.

endurance	TimeDuration	1	Supplementary endurance time.	This is the maximum remaining flight time for this contingency plan.
contingencyGeometry	Geometry	1	The geographical area for this contingency.	If responseAction is LANDING or RETURN_TO_BASE, then this is the area on the ground that the UA will be targeting for a landing. The polygon should be large enough to provide high confidence (for some TBD value of "high") that the vehicle will land within it, but hopefully be no larger (for some TBD value of hopefully). If responseAction is LOITERING, this is the 3D-volume the UA will stay within during loitering.
contingencyExtraVolume	OperationVolume	0..*	Describes an extra portion of airspace, which might be needed in the contingecy plan if the emergency landing shall take place outside the original operation trajectory.	This extra 4D-volume shall be adjacent/overlapping to the referred relevantOperationVolumes in the same manner as subsequent OperationVolumes must be adjacient/overlapping (see descriptive Notes on the operationVolumes property of the OperationPlan).

Table 12: The ContingencyPlan data structure

5.6 The UasRegistration Data Structure

UasRegistration is the collection of registration data for UAVs.

Property	Type	Multiplicity	Description	Note
droneId	String	0..1	Drone identifier.	
registrationId	UUID	0..1	Unique identifier referring to a registration.	
registrationLocation	String	1	An Internet-reachable URL for the registration authority.	More details to come, however, it is thought that this should be an endpoint allowing an unauthenticated GET to obtain metadata about the registrar.

Table 13: The UasRegistration data structure

5.7 The FlightDetails Data Structure

FlightDetails provides some flight related data of interest for an operation plan.

Property	Type	Multiplicity	Description	Note
flightNumber	String	0..1	Optional. For use by USS for identification purposes.	
flightType	String	0..1	Optional. Allows to distinguish different types of flight.	
flightComment	String	0..1	Optional informative text about the operation.	Not used by the UTM System. Only for human stakeholders.
maxFlightSpeedKnots	Double	0..1	Maximum flight speed in knots.	

Table 14: The FlightDetails data structure

5.8 The Priority Data Structure

The Priority data structure describes the priority of an operation plan (or of an operation volume). On of the two optional priorityLevel attributes must be provided.

Property	Type	Multiplicity	Description	Note
priorityText	String	0..1	Free text description of the priority classification.	

priorityLevelSimple	PriorityLevelSimple	0..1	Priority may be expressed in this simple way, allowing to classify the OP as Low-, Medium-, or High-Priority flight.	
priorityLevelCorus	PriorityLevelCorus	0..1	Alternatively, the priority may be expressed as one of the 8 priority levels defined in Corus.	

Table 15: The Priority data structure

5.9 The OperationTrajectory Data Structure

An OperationTrajectory defines a 4D trajectory to be flown, including the uncertainty.

Property	Type	Multiplicity	Description	Note
positionCRS	EnumCRSType	1	Coordinate Reference System used for the horizontal dimensions of the trajectory elements.	
altitudeCRS	EnumCRSType	1	Coordinate Reference System used for the vertical dimension of the trajectory elements.	
positionUncertainty	float	1	Maximum allowed horizonal deviation from individual trajectory elements.	
altitudeUncertainty	float	1	Maximum allowed vertical deviation from individual trajectory elements.	
timingUncertainty	Duration	1	Maximum allowed timing deviation from individual trajectory elements.	
altitudeType	EnumAltitudeType	1	Indicates how the vertical dimension is defined in the trajectory elements.	
trajectoryElements	TrajectoryElement	1..*	Ordered list of 4D points	

Table 16: The OperationTrajectory data structure

5.10The TrajectoryElement Data Structure

A Trajectoryelement defines a single 4D position to be used in an OperationTrajectory.

Reference coordinate system as well as allowed tolerances (uncertainties) are defined in the OperationTrajectory, valid for all TrajectoryElements.

Property	Type	Multiplicity	Description	Note

latitude	float	1	latitude value.	
longitude	float	1	longitued value.	
altitude	float	1	vertical position value.	Altitude or height value, depending on the altitudeType given in the OperationTrajectory.
time	DateTime	1	time value.	

Table 17: The TrajectoryElement data structure

5.11The Mission Data Structure

A Mission is an administrative object, allowing to provide common information about flights that are related to each other, e.g., as they are defined to achieve a common goal.

Property	Type	Multiplicity	Description	Note
missionId	UUID	1	Globally unique identification of a mission.	
beginTime	DateTime	0..1	Time stamp indicating the begin of the mission.	
endTime	DateTime	0..1	Time stamp indicating the end of the mission.	
description	String	0..1	Mission description.	
contactDetails	ContactDetails	0..1	Contact details for the mission.	
publicInfo	PublicInformation	0..1	Additional descriptive information of the mission.	
operationPlans	Reference to OperationPlan	0..*	Reference to operation plans sharing this mission.	

Table 18: The Mission data structure

5.12The PublicInformation Data Structure

PublicInformation allows to provide additional information allowed for public disclosure.

Property	Type	Multiplicity	Description	Note

title	String	0..1	Short public title for the operation plan.	
description	String	0..1	Public description of the operation plan.	

Table 19: The PublicInformation data structure

5.13The OperationPlanState Enumeration

The OperationPlanState enumeration type specifies the possible states of an operation plan.

Property	Description	Note
PROPOSED	Initial state of the operation plan.	This operation is not yet approved. It may be awaiting information from the operator, it may be in conflict with another operation and undergoing a negotiation process, or for some other reason it is not yet able to be declared PERMITTED.
PERMITTED	Authority has given permission to proceed (Certification Processes, SORA, ...)	
AUTHORIZED	This operation has been deemed approved by the supporting USS. This implies that the operation meets the requirements for operating in the airspace based on the type of operation submitted.	Authorization of an OP may include the approval by multiple stakeholders. ATM may be one such stakeholder. In some cases an OP may be AUTHORIZED without the approval of ATM (in cases where no ATM airspace is involved.
ACTIVATED	Operation is cleared for takeoff.	The **OperationPlanState** remains ACTIVATED even when the drone **flight** gets ACTIVE or INACTIVE (this is indicated by the **DroneFlightState**; see DroneFlightExchange service).
CLOSED	This operation is closed. It is not airborne and will not become airborne again.	If the UAS and the crew will fly again, it would need to be submitted as a new operation. A USS may announce the closure of any operation, but is not required to announce unless the operation was ROGUE or NONCONFORMING. The closure reason is noted in the operation plan attribute closureReason.

Table 20: The OperationPlanState enumeration

5.14The EnumClosureReason Enumeration

The EnumClosureReason enumeration type specifies the possible reasons for closing an operation plan

Property	Description	Note
NOMINAL	Operation completed nominally.	
REJECTED	OP has been rejected by USSP, CISP or authority.	
REVOKED	OP has been revoked by USSP, CISP or authority.	

Table 21: The PriorityLevelSimple enumeration

5.15The ContingencyCauseType Enumeration

The ContingencyCauseType enumeration type specifies the possible causes for contingency operations.

Property	Description	Note
LOST_C2_UPLINK	Command and control connection from controller to drone was lost.	
LOST_C2_DOWNLINK	Command and control connection from drone to controller was lost.	
LOST_NAV	Navigation equipment lost.	
LOST_SAA		
LOW_FUEL	Not enough energy.	
NO_FUEL	Energy lost.	
MECHANICAL_PROBLEM	Mechanical problem.	
SOFTWARE_PROBLEM	Software problem.	
ENVIRONMENTAL	Environmental issues.	
SECURITY	Security issues.	
TRAFFIC	Other traffic.	
LOST_USS		
OTHER		

ANY		

Table 22: The ContingencyCauseType enumeration

5.16The ContingencyLocationDescriptionType Enumeration

The ContingencyLocationDescriptionType enumeration type specifies the possible ways to describe a contingency plan.

Property	Description	Note
PREPROGRAMMED	Contingency location that is determined prior to launch and programmed onto the UA.	
OPERATOR_UPDATED	Contingency location that is (or will be) updated during operation by operator (e.g., sent to UA).	
UA_IDENTIFIED	Contingency location that is identified to be safe to land by the UA itself.	
OTHER	Contingency location does not fit any of the defined categories.	

Table 23: The ContingencyLocationDescriptionType enumeration

5.17The ContingencyResponseType Enumeration

The ContingencyResponseType enumeration type specifies the possible kinds of contingency response actions.

Property	Description	Note
LANDING	The contingency operation will be landing.	
LOITERING	The operation will loitering.	
RETURN_TO_BASE	The operation will return to base.	
OTHER	Additional details should be provided in freeText.	If this gets used often for similar events, Enumeration will be updated with new value.

Table 24: The ContingencyResponseType enumeration

5.18The PriorityLevelSimple Enumeration

The PriorityLevelSimple enumeration type specifies three simple priority levels.

Property	Description	Note

PRIO_LOW	low priority	
PRIO_MEDIUM	medium priority	
PRIO_HIGH	high prioroty	

Table 25: The PriorityLevelSimple enumeration

5.19The PriorityLevelCorus Enumeration

The PriorityLevelCorus enumeration type specifies priority levels defined in the Corus Conops.

Property	Description	Note
PRIO_1_EMERGENCY		
PRIO_2_HOSPITAL		
PRIO_3_AUTHORITIES		
PRIO_4_URGENT_TRANSPORT		
PRIO_5_INDUSTRY		
PRIO_6_TRANSPORT		
PRIO_7_FILMING		
PRIO_8_LEISURE		

Table 26: The PriorityLevelCorus enumeration

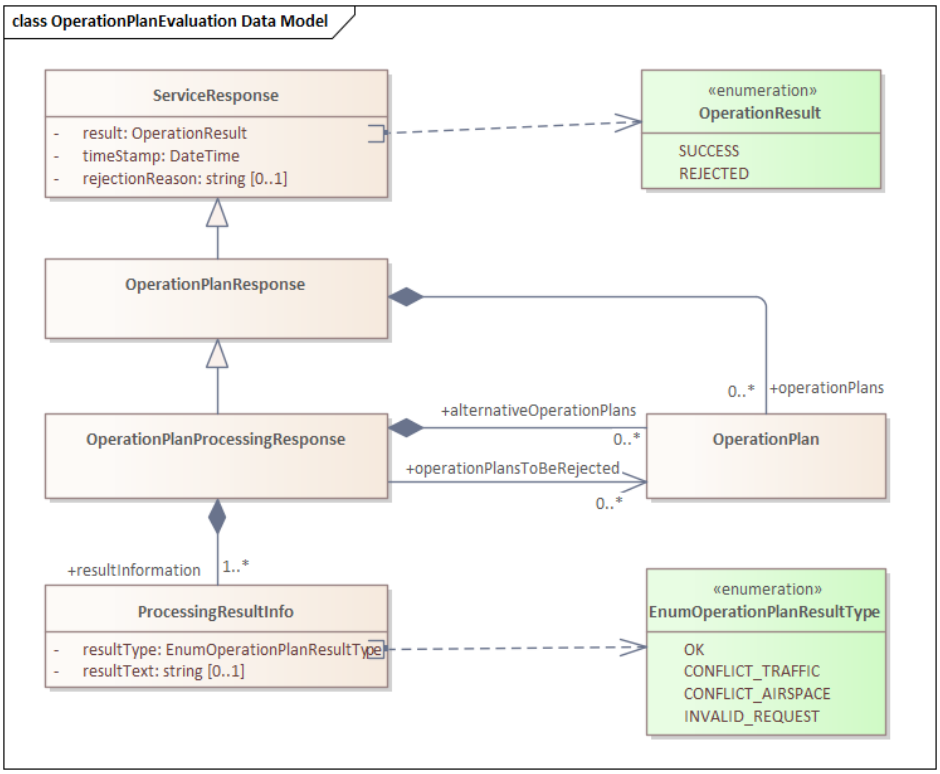

Figure 3: Operation Plan Services - Operation Plan Processing Response data structure diagram

5.20The OperationPlanProcessingResponse Data Structure

OperationPlanProcessingResponse is used to carry the result of evaluation or processing requests for operation plans.

In case of a negative result, it may contain one or several alternative operation plans.

Property	Type	Multiplicity	Description	Note
resultInformation	ProcessingResultInfo	1..*	Indicates the result of evaluating or processing an operation plan. In case of negative result, more than one **resultInformation** may be added.	

alternativeOperationPlans	OperationPlan	0..*	Optional alternative Operation Plans that may be proposed instead of the evaluated operation plan, in case the resultInformation.resultType is other than OK. In case of resultInformation.resultType=OK, this list shall be empty.	see section on Basic Operation Plan data structures for a description of OperationPlan.
operationPlansToBeRejected	UUID	0..*	Operation plans to be rejected in order to be able to accept the given Operation Plan. In case of conflicting operation plans, the service provider may use this property to indicate that the given OP could be accepted, if other OPs (listed in this property) would be rejected.	
< inherited > operationPlans	OperationPlan	0..*	In this property, the **OperationPlanProcessingResponse** may contain the OperationPlan for which the evaluation/submission was requested. **(inherited from OperationPlanResponse data structure)**	see section on Basic Operation Plan data structures for a description of **OperationPlanResponse** .
< inherited > ...			**All properties inherited from ServiceResponse.**	**See common data types.**

Table 27: The OperationPlanProcessingResponse data structure

5.21The ProcessingResultInfo Data Structure

ProcessingResultInfo is used to carry information about the result of evaluation or processing requests for operation plans.

Property	Type	Multiplicity	Description	Note
resultType	EnumOperationPlanResultType	1	Indicates the result of evaluating an operation plan.	

resultText	String	0..1	Textual description of the reason result. For example, a textual description of the rejection reason. In case of a positive resultType, this field may be empty or omitted.	

Table 28: The ProcessingResultInfo data structure

5.22The EnumOperationPlanResult Enumeration

The **EnumOperationPlanResult** enumeration type specifies potential results of an Operation Plan processing.

Property	Description	Note
OK	The processing result is OK, i.e., the Operation Plan can be accepted.	
CONFLICT_TRAFFIC	The Operation Plan cannot be accepted due to conflicting traffic.	
CONFLICT_AIRSPACE	The Operation Plan cannot be accepted due to a conflict related to airspace restriction.	
INVALID_REQUEST	The Operation plan cannot be accepted due to invalid data. E.g. invalid version, forbidden state transfer etc.	

Table 29: The EnumOperationPlanResult enumeration

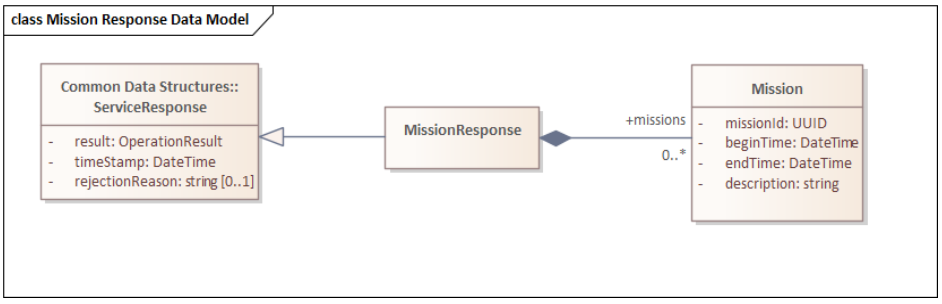

Figure 4: Operation Plan Services - Mission Response data structure diagram

5.23The MissionResponse Data Structure

MissionResponse is used to carry the result of query-operations asking for Missions.

Depending on the operation result, it may contain zero, one or several Mission objects.

Property	Type	Multiplicity	Description	Note
missions	Mission	0..*	Mission data structure(s).	see section on Basic Operation Plan data structures for a description of Mission.
< inherited >			**All properties inherited from ServiceResponse.**	**See common data types.**

Table 30: The MissionResponse data structure

5.24Common Data Structures Used in UTM Service Specifications

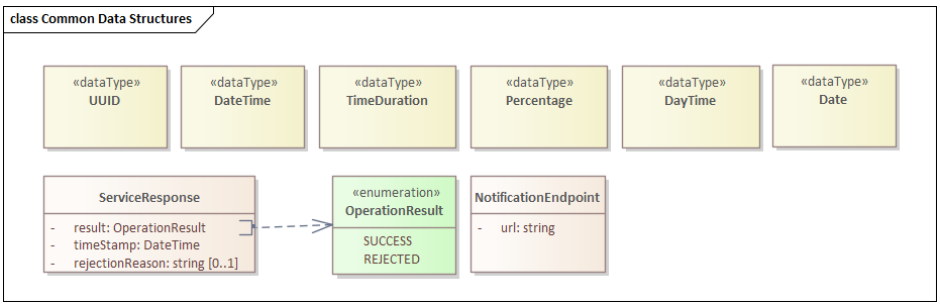

Figure 5: Common Data Types Used in UTM Service Specifications

5.24.1NotificationEndpoint **Data Structure**

NotificationEndpoint is used in subscription and un-subscription operations to show the receiver of notifications as a result of the subscription.

Property	Type	Multiplicity	Description	Note
URL	**String**	1	Endpoint capable of receiving notifications	

Table 31: NotificationEndpoint Data Structure

5.24.2ServiceResponse **Data Structure**

ServiceResponse is the generic response provided by each service operation. In some cases, this basic data structure may be extended by inheritance.

Property	Type	Multiplicity	Description	Note

result	**OperationResult**	1	Indicates the result of the request to the service	
rejectReason	**String**	0..1	Optional additional information to be provided in case of negative result	
timeStamp	DateTime	1		

Table 32: ServiceResponse Data Structure

5.24.3 OperationResult **Enumeration**

The **OperationResult** enumeration type specifies the possible outcomes of calling an operation.

Property	Description	Note
SUCCESS	Operation was successfully executed.	
REJECTED	Operation could not be executed.	

Table 33: OperationResult Enumeration

5.25 Common Geometry Data Structures Used in UTM Service Specifications

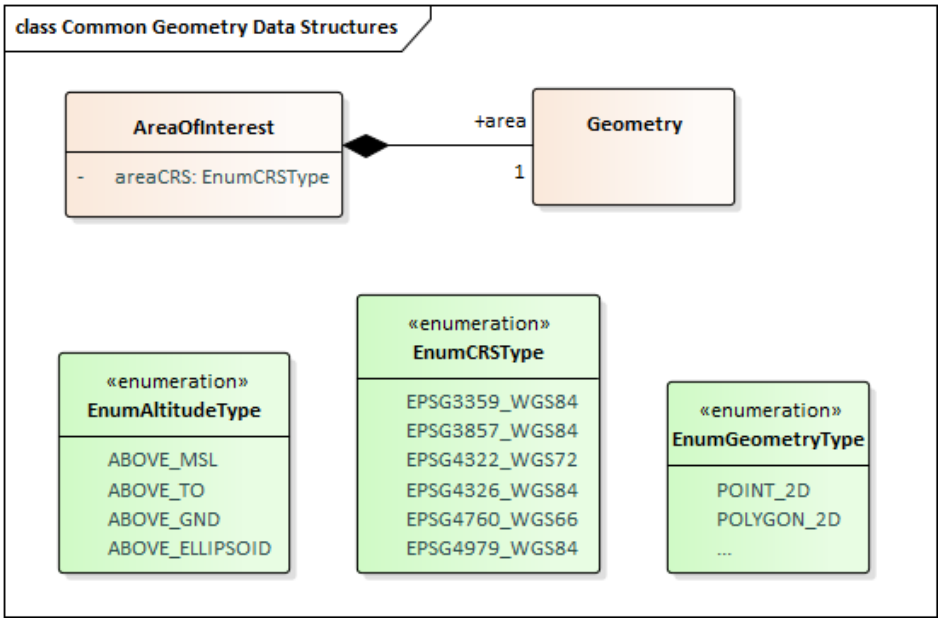

Figure 6: Common Geometry Data Types Used in UTM Service Specifications

5.25.1AreaOfInterest **Data Structure**

AreaOfInterest is used in subscription operations to provide an indication of the geographic area for which the subscriber is interested to receive notifications.

Property	Type	Multiplicity	Description	Note
area	**Geometry**	1	A geometric description of a geographic area.	Should be a 2-dimensional geometry in this case.
areaCRS	**EnumCRSType**	1	Coordinate reference system used (WGS-84, EPSG:4979)	

Table 34: AreaOfInterest Data Structure

5.25.2Geometry **Data Structure**

Geometry describes a geometrical shape of one, two or three dimensions.

The **Geometry** data structure is not further detailed in this service specification. One example of how a generic Geometry structure could be realized is sketched in the table below:

Property	Type	Multiplicity	Description	Note
coordinates	**Double**	2..*	Collection of the coordinates, describing the geometry.	
geometryType	**GeometryType**	1	Type of geometry being described by the coordinates.	Examples: Point, Polygon, Polyhedron, etc.

Table 35: Geometry Data Structure

5.25.3EnumAltitudeType **Enumeration**

The **EnumAltitudeType** enumeration type specifies the possible ways to express an altitude/height.

Property	Description	Note
ABOVE_MSL	Altitude above mean-sea-level. Same as orthometric height; same as height above the earth geoid.	
ABOVE_TO	Altitude above take-off location.	
ABOVE_GND	Height above ground surface.	
ABOVE_ELLIPSOID	Altitude above the WGS-84 ellipsoid; value delivered by GPS.	

Table 36: EnumAltitudeType Enumeration

5.25.4EnumCRSType **Enumeration**

The **EnumCRSType** enumeration type specifies the possible ways to express a coordinate reference system. The most common values used are noted in bold letters.

Property	Description	Note
EPSG3395_WGS84	World Mercator Geodetic CRS: WGS 84; Coordinate System: Cartesian CS. Axes: easting, northing (E, N). Orientations: east, north. UoM: metre.	Euro-centric view of world excluding polar areas.
EPSG3857_WGS84	Pseudo-Mercator -- Spherical Mercator, Google Maps, OpenStreetMap, Bing, ArcGIS, ESRI Geodetic CRS: WGS 84; Coordinate System: Cartesian CS. Axes: easting, northing (X, Y). Orientations: east, north. UoM: metre.	Uses spherical development of ellipsoidal coordinates. Relative to WGS 84 / World Mercator (CRS code 3395) errors of 0.7 percent in scale and differences in northing of up to 43km in the map (equivalent to 21km on the ground) may arise.
EPSG4322_WGS72	Geodetic CRS: WGS 72; Coordinate System: Ellipsoidal 2D CS. Axes: latitude, longitude. Orientations: north, east. UoM: degree.	Uses Historic World Geodetic System 1972. Horizontal component of 3D system.

EPSG4326_WGS84	WGS84 - World Geodetic System 1984, used in GPS Geodetic CRS: WGS 84; Coordinate System: Ellipsoidal 2D CS. Axes: latitude, longitude. Orientations: north, east. UoM: degree.	Horizontal component of 3D system. Used by the GPS satellite navigation system and for NATO military geodetic surveying.
EPSG4760_WGS66	Geodetic CRS: WGS 66; Coordinate System: Ellipsoidal 2D CS. Axes: latitude, longitude. Orientations: north, east. UoM: degree.	Uses Historic World Geodetic System 1966. Horizontal component of 3D system.
EPSG4979_WGS84	Geodetic CRS: WGS 84; Coordinate System: Ellipsoidal 3D CS. Axes: latitude, longitude, ellipsoidal height. Orientations: north, east, up. UoM: degree, degree, metre.	Used by the GPS satellite navigation system.

Table 37: EnumCRSType Enumeration

5.25.5EnumGeometryType **Enumeration**

The **EnumGeometryType** enumeration type specifies possible geometrical shapes.

Property	Description	Note
POINT	Single point.	
POLYGON	Polygon.	
...		

Table 38: EnumGeometryType Enumeration

5.26 Common Address Data Structures Used in UTM Service Specifications

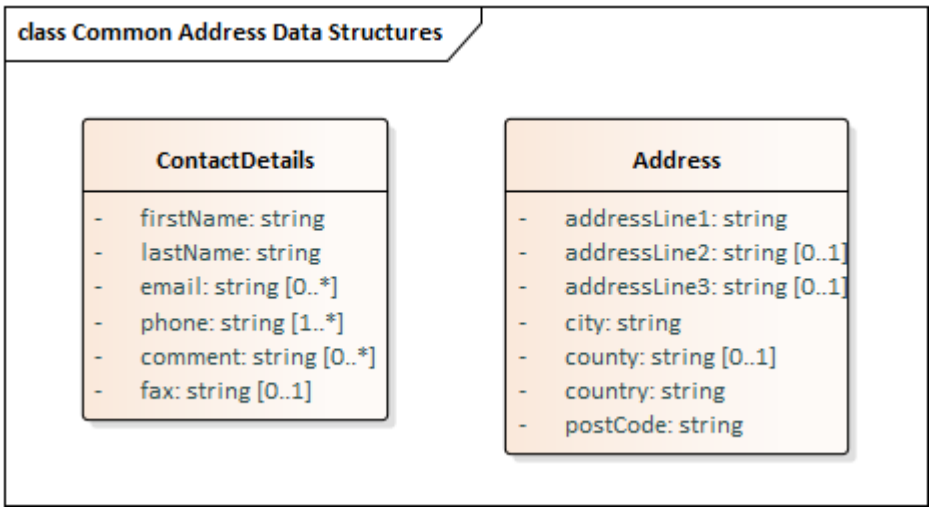

Figure 7: Common Address Data Types Used in UTM Service Specifications

5.26.1 Address Data Structure

Address is used to represent an address of a person or an organization.

Property	Type	Multiplicity	Description	Note
addressLine1	**string**	1	First address line, typically the street name and house number.	
addressLine2	**string**	0..1	Optional second address line, typically the unit, floor or apartment number.	
addressLine3	**string**	0..1	Optional third address line for country specific information.	
city	**string**	1	The name of the city.	
county	**string**	0..1	Optionally, the name of the county.	
country	**string**	1	The name of the country.	
postCode	**string**	1	Post code.	

Table 39: Address Data Structure

5.26.2ContactDetails **Data Structure**

The ContactDetails data structure is used to collect contact information. A contact may be a Person, State, Organisation, Authority, aircraft operating agency, handling agency etc.

Property	Type	Multiplicity	Description	Note
firstName	**string**	1	First name of the contact.	
lastName	**string**	1	Last name of the contact.	
email	**string**	0..*	An optional array of email addresses.	To establish best practices, the order of the email addresses in the array should indicate the order that they should be used. The responsibility is on the USS providing the email address to ensure it is valid and operational.
phone	**string**	1..*	An array of at least one phone number.	To establish best practices, the order of the phone numbers in the array should indicate the order that they should be used.
fax	**string**	0..1	Optional fax number to reach the contact	
comment	**string**	0..*	Any additional comments related to contact information.	

Table 40: ContactDetails Data Structure

6 Service Interface Specifications

This chapter describes the details of each service interface. One sub-chapter is provided for each Service Interface.

The Service Interface specification covers only the static design description while the dynamic design (behaviour) is described later.

6.1 Service Interface OperationPlanRetrievalInterface

The service provider offers this interface to allow consumers to query operation plan data.

6.1.1 Operation getOperationPlanForId

6.1.1.1 Operation Functionality

A consumer calls this operation to explicitly request an operation plan by submitting the known id.

6.1.1.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
id	Input	UUID	Identifier of an operation plan.
response	Return	OperationPlanResponse	Query response, including the operation plan data, if the request was successful.

Table 41: Payload description of getOperationPlanForId operation

6.1.2 Operation getOperationPlanVersion

6.1.2.1 Operation Functionality

A consumer calls this operation to explicitly request a certain version of an operation plan by submitting the known ids.

6.1.2.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
id	Input	UUID	Identifier of an operation plan.
version	Input	UUID	Version identifier.
response	Return	OperationPlanResponse	Query response, including the operation plan data, if the request was successful.

Table 42: Payload description of getOperationPlanVersion operation

6.1.3 Operation getOperationPlanForMission

6.1.3.1 Operation Functionality

A consumer calls this operation to explicitly request operation plan data belonging to a certain mission.

6.1.3.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
missionId	Input	UUID	Identifier of a mission.
response	Return	OperationPlanResponse	Query response, including the operation plan data, if the request was successful. The OperationPlanResponse may contain a list of OperationPlans: all OPs belonging to the mission.

Table 43: Payload description of getOperationPlanForMission operation

6.1.4 Operation getOperationPlanForGeometry

6.1.4.1 Operation Functionality

A consumer calls this operation to explicitly request operation plan data for a certain geographical area.

6.1.4.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
geometry	Input	Geometry	Geographical area of interest.
response	Return	OperationPlanResponse	Query response, including the operation plan data, if the request was successful. The OperationPlanResponse may contain a (potentially empty) list of OperationPlans: all OPs for the area of interest.

Table 44: Payload description of getOperationPlanForGeometry operation

6.1.5 Operation getOperationPlans

6.1.5.1 Operation Functionality

A consumer calls this operation to explicitly request operation plan data matching various criteria

6.1.5.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
exampleOP	Input	OperationPlan	Subset of an operation plan. This allows to filter e.g., all OPs with a given contactDetails, or all OPs for a certain drone registration id.
response	Return	OperationPlanResponse	Query response, including the operation plan data, if the request was successful. The OperationPlanResponse may contain a list of OperationPlans: all OPs matching the data given in the exampleOP.

Table 45: Payload description of getOperationPlans operation

6.2 Service Interface OperationPlanSubscriptionInterface

The service provider offers this interface to allow consumers to subscribe/unsubscribe for operation plan data.

6.2.1 Operation subscribeForOperationPlan

6.2.1.1 Operation Functionality

A consumer calls this operation to subscribe to receive operation plan data.

6.2.1.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
consumer	Input	NotificationEndpoint	Which endpoint shall be notified in case of new OperationPlan data.
areaOfInterest	Input	AreaOfInterest	Area of interest to the consumer
response	Return	ServiceResponse	Provide status information on subscription

Table 46: Payload description of subscribeForOperationPlans operation

6.2.2 Operation unsubscribe

6.2.2.1 Operation Functionality

A consumer calls this operation at the provider to unsubscribe from operation plan or no flight zone data.

6.2.2.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
consumer	Input	NotificationEndpoint	Which endpoint shall not be notified (anymore) in case of new OperationPlans or No Flight Zones.
response	Return	ServiceResponse	Provide status information on subscription

Table 47: Payload description of unsubscribe operation

6.3 Service Interface OperationPlanNotificationInterface

Once and while subscribed, consumer receives operation plan data via this interface.

6.3.1 Operation notifyOperationPlan

6.3.1.1 Operation Functionality

The service provider uses this logical operation (implemented by the consumer) to publish operation plan data.

6.3.1.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
op	Input	OperationPlan	An operation plan matching the filter criteria provided in the subscription

Table 48: Payload description of notifyOperationPlan operation

6.4 Service Interface OperationPlanManagementInterface

The service provider offers this interface to allow service consumers to manage operation plans. This includes initial creation as well as modification of operation plans and missions; furthermore it includes the life cycle management for operation plans (request for authorisation, request for takeoff, cancellation, revocation, etc.)

6.4.1 Operation evaluateOperationPlan

This operation allows to evaluate an operation plan without actually submitting it for approval.

6.4.1.1 Operation Functionality

The service provider investigates the input operation plan and evaluates whether it can be accepted as is, or whether there exist conflicts with other (planned) operations or with airspace restrictions.

If the operation plan is acceptable, this fact is simply given back as a result to the invoker, but the operation plan is not pre-noted in any way.

If the operation plan is not acceptable, this fact is given back as a result to the invoker, and alternative, acceptable operation plans may be added to the result (e.g., slightly modified in time or space).

6.4.1.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
op	Input	OperationPlan	The "op" parameter specifies an OperationPlan that shall be evaluated.
<none>	Return	OperationPlanProcessingResponse	The return value provides the evaluation result. In case of a negative result, the return value may contain one or several alternative OperationPlans, if the service provider is able to make such proposals.

Table 49: Payload description of evaluateOperationPlan operation

6.4.2 Operation proposeOperationPlan

This operation allows to evaluate an operation plan and proposing it (i.e., scheduling it as a proposed plan).

6.4.2.1 Operation Functionality

The service provider investigates the input operation plan and evaluates whether it can be accepted as is, or whether there exist conflicts with other (planned) operations or with airspace restrictions.

If the operation plan is acceptable, this fact is given back as a result to the invoker, and the operation plan is pre-noted and published as a proposed plan to any subscribers.

If the operation plan is not acceptable, this fact is given back as a result to the invoker, and alternative, acceptable operation plans may be added to the result (e.g., slightly modified in time or space). Such alternatives should be pre-noted as proposed plans for a short amount of time.

6.4.2.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
op	Input	OperationPlan	The "op" parameter specifies an OperationPlan that shall be evaluated and proposed.

<none>	Return	OperationPlanProcessingResponse	The return value provides the evaluation result. In case of a negative result, the return value may contain one or several alternative OperationPlans, if the service provider is able to make such proposals.

Table 50: Payload description of proposeOperationPlan operation

6.4.3 Operation submitOperationPlan

This operation allows to propose an operation plan and submitting it (i.e., requesting for authorisation) in one shot.

6.4.3.1 Operation Functionality

The service provider investigates the input operation plan and evaluates whether it can be accepted as is, or whether there exist conflicts with other (planned) operations or with airspace restrictions. Furthermore, if there are no conflicts, the service provider initiates all necessary approval steps for the operation plan.

If the operation plan is acceptable and can be approved, this fact is given back as a result to the invoker, and the operation plan is published as an approved plan to any subscribers.

If the operation plan is not acceptable or cannot be approved, this fact is given back as a result to the invoker.

6.4.3.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
op	Input	OperationPlan	The "op" parameter specifies an OperationPlan that shall be evaluated and authorised.
<none>	Return	OperationPlanProcessingResponse	The return value provides the evaluation result. In case of a negative result, the return value may contain one or several alternative OperationPlans, if the service provider is able to make such proposals.

Table 51: Payload description of submitOperationPlan operation

6.4.4 Operation updateOperationPlan

This operation allows to modify an operation plan by providing the updated operation plan data.

6.4.4.1 Operation Functionality

The service provider looks up the operation plans in order to find one which's identifier matches the input operation plan. If there is a matching OP, the service provider tries to create a new version of the OP, taking into account the values given in the input OP.

If the updates to the operation plan are acceptable, this fact is given back as a result to the invoker, and the updated operation plan is published to any subscribers.

If the operation plan is not acceptable, this fact is given back as a result to the invoker, and no publication takes place.

6.4.4.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
op	Input	OperationPlan	The "op" parameter specifies an updated version of an operation plan.
<none>	Return	ServiceResponse	The return value provides the operation result.

Table 52: Payload description of updateOperationPlan operation

6.4.5 Operation cancelOperationPlan

This operation allows to cancel an operation plan by providing the operation plan identifier.

6.4.5.1 Operation Functionality

The service provider looks up the operation plans in order to find one which's identifier matches the input opId. If there is a matching OP, the service provider changes the state of this OP to "CLOSED" and the closureReason to "CANCELLED".

If the operation was successful, this fact is given back as a result to the invoker, and the cancelled operation plan is published to any subscribers.

If the operation plan is not successful, this fact is given back as a result to the invoker, and no publication takes place.

6.4.5.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
opId	Input	UUID	Identifier of an operation plan that shall be cancelled.
<none>	Return	ServiceResponse	The return value provides the operation result.

Table 53: Payload description of cancelOperationPlan operation

6.4.6 Operation requestOperationPlanAuthorisation

This operation allows to request the necessary approval steps for a proposed operation plan by providing the operation plan identifier.

6.4.6.1 Operation Functionality

The service provider looks up the operation plans in order to find one which's identifier matches the input opId. If there is a matching OP, the service provider initiates the authorisation process for this OP.

Note that authorisation may include several steps and involvement of several stakeholders and therefore may take some time to compete.

If the operation was successful, this fact is given back as a result to the invoker, and the authorized operation plan is published to any subscribers, as soon as the authorisation is complete.

If the operation is not successful, this fact is given back as a result to the invoker, and no publication takes place.

6.4.6.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
opId	Input	UUID	Identifier of an operation plan that shall be authorised.
<none>	Return	ServiceResponse	The return value provides the operation result.

Table 54: Payload description of requestOperationPlanAuthorisation operation

6.4.7 Operation revokeOperationPlan

This operation allows to revoke an operation plan by providing the operation plan identifier.

6.4.7.1 Operation Functionality

The service provider looks up the operation plans in order to find one which's identifier matches the input opId. If there is a matching OP, the service provider changes the state of this OP to "CLOSED" and the closureReason to "REVOKED".

If the operation was successful, this fact is given back as a result to the invoker, and the revoked operation plan is published to any subscribers.

If the operation plan is not successful, this fact is given back as a result to the invoker, and no publication takes place.

6.4.7.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
opId	Input	UUID	Identifier of an operation plan that shall be revoked.

<none>	Return	ServiceResponse	The return value provides the operation result.

Table 55: Payload description of revokeOperationPlan operation

6.4.8 Operation requestTakeOff

This operation allows to request take-off permission for an authorized operation plan by providing the operation plan identifier.

6.4.8.1 Operation Functionality

The service provider looks up the operation plans in order to find one which's identifier matches the input opId. If there is a matching OP, and if the OP is in state AUTHORIZED, the service provider changes the state of this OP to "TAKEOFFGRANTED".

If the operation was successful, this fact is given back as a result to the invoker, and the operation plan is published to any subscribers.

If the operation plan is not successful, this fact is given back as a result to the invoker, and no publication takes place.

6.4.8.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
opId	Input	UUID	Identifier of an operation plan for which takeoff is requested.
<none>	Return	ServiceResponse	The return value provides the operation result.

Table 56: Payload description of requestTakeOff operation

6.4.9 Operation declareEndOfFlight

This operation allows to declare the end of flight for an active operation plan by providing the operation plan identifier.

6.4.9.1 Operation Functionality

The service provider looks up the operation plans in order to find one which's identifier matches the input opId. If there is a matching OP, and if the OP is in state TAKEOFFGRANTED, the service provider changes the state of this OP to "CLOSED" and the closureReason to "NOMINAL".

If the operation was successful, this fact is given back as a result to the invoker, and the closed operation plan is published to any subscribers.

If the operation plan is not successful, this fact is given back as a result to the invoker, and no publication takes place.

6.4.9.2 Operation Parameters

Parameter Name	Direction	Data Type	Description
opId	Input	UUID	Identifier of an operation plan that shall be declared finished.
<none>	Return	ServiceResponse	The return value provides the operation result.

Table 57: Payload description of declareEndOfFlight operation

6.4.10Operation createMission

This operation allows to initiate a mission.

TBD

6.4.11Operation updateMission

This operation allows update a mission.

TBD

7 Service Dynamic Behaviour

7.1 Sequence of events, cooperation with other services

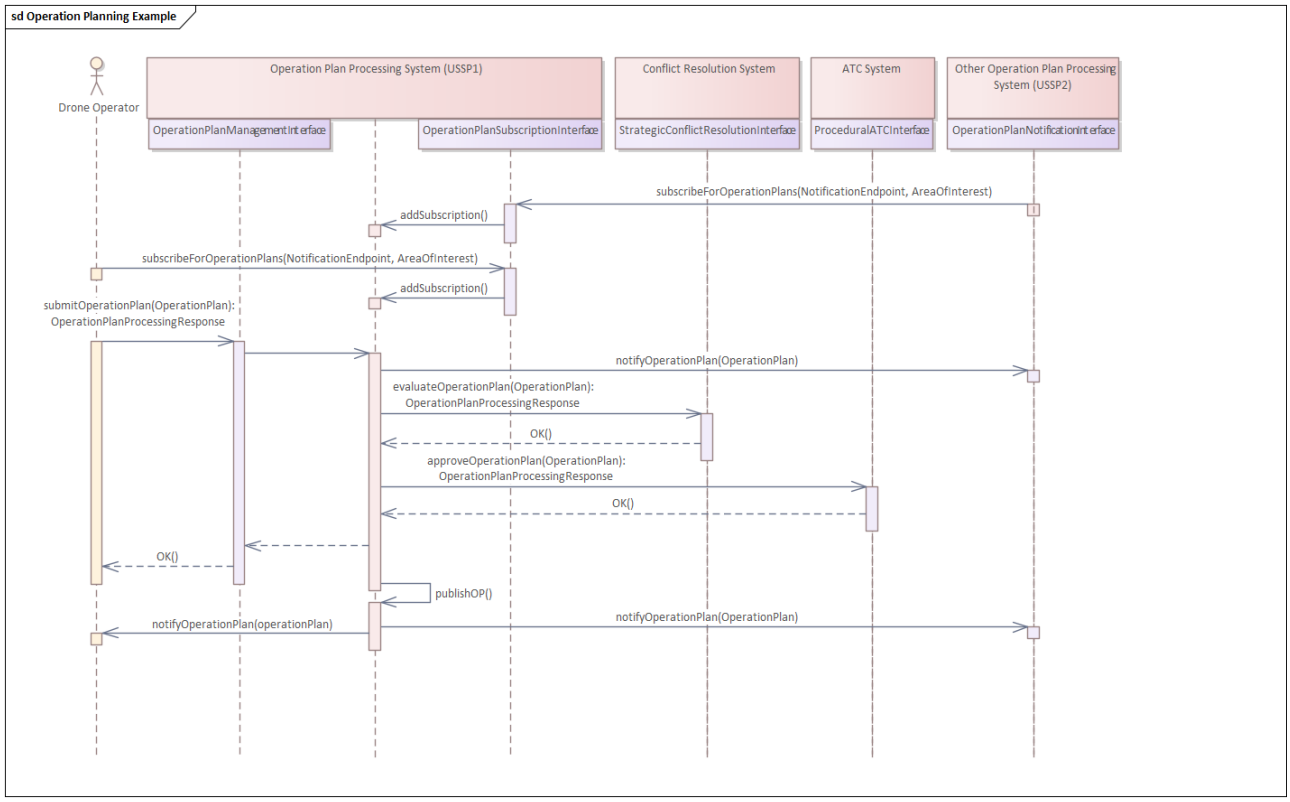

Figure 8: OperationPlanExchange service operation sequence diagram

Note:
In order to illustrate the service operations in a realistic context, this Sequence Diagram contains additional operations from other services, not only OperationPlanExchange service operations.

The figure above provides an example scenario for the OperationPlanExchange service. The scenario assumes an Operation Plan Processing system at USSP1 providing the OperationPlanExchange service. Another Operation Plan Processing system at a second USSP2 in this scenario plays the role of a consumer of the OperationPlanExchange service, by subscribing to the first USSP. In addition, the scenario includes a Drone Operator (in the role of a consumer of the OperationPlanExchange service of USSP1). Furthermore, the scenario finally illustrates an example Conflict Resolution system (providing the StrategicConflictResolution service) and an example ATC System (providing the ProceduralATCInterface service).

- The scenario starts with the consumers of the OperationPlanExchange service subscribing for operation plans:

- USSP2 subscribes at USSP1 in order to receive all operation plans (under the assumption that USSP2 provides services in the same or overlapping geographical area as USSP1)
- Drone Operator subscribes at USSP1 in order to get informed about operation plans in a certain area.

- Drone Operator submits a tentative Operation Plan by using the OperationPlanManagmentInterface of the OperationPlanExchange service
- USSP1 immediately publishes the (tentative) operation plan to all subscribers. So the second USSP gets informed about the ongoing planning performed in the first USSP in an early stage.
- USSP1 may need to contact a Conflict Resolution System (via StrategicConflictReslution service) in order to check the requested operation plans for conflicts with existing plans.
- USSP1 may need to contact the ATC System (via ProceduralATCInterface service) in order to get an approval from ATC.
- As soon as all checks are done and the operation plan is approved, USSP1 will provide the response to Drone Operator.
- At the same time, USSP1 will publish the OperationPlan to all subscribers.

7.2 Operation Plan State Machine

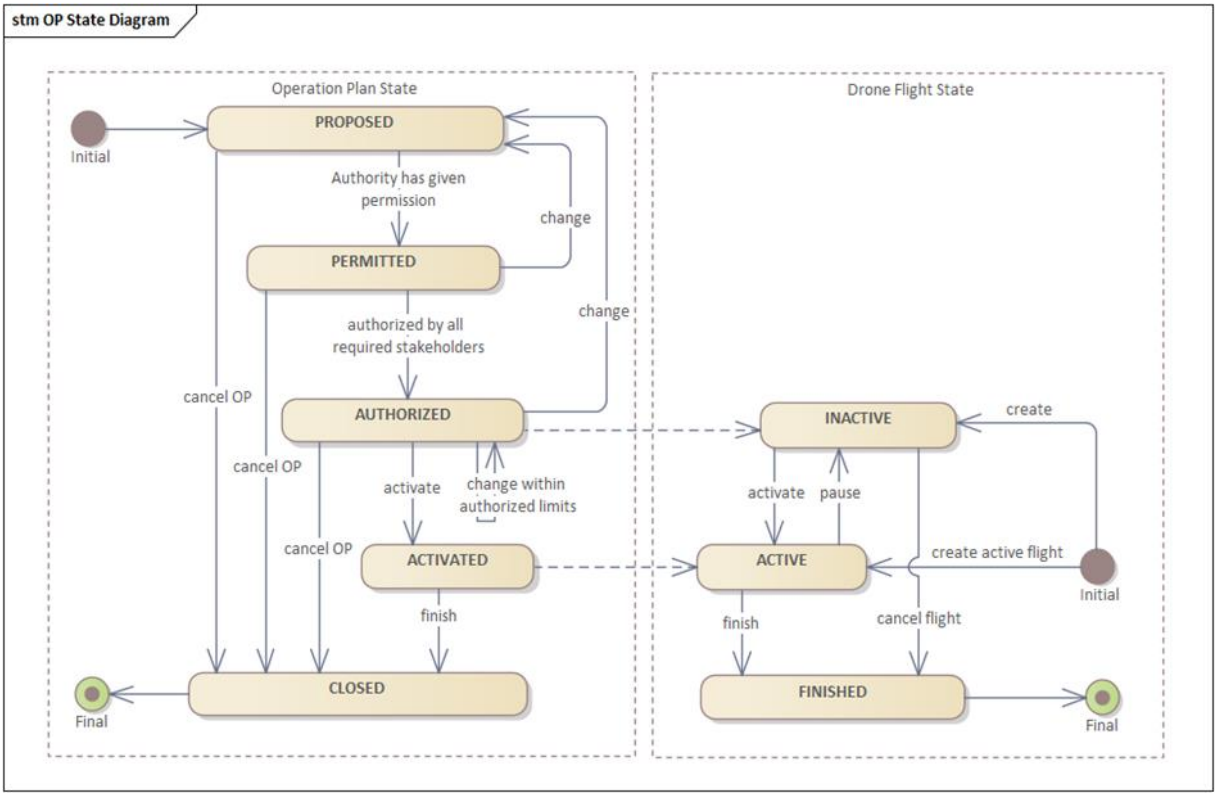

Figure 9: Operation Plan states - state transition diagram

8 References

This section identifies the documents (name, reference, source project) the Study Plan has **to comply to or to be used as additional inputs.**

Nr.	Version	Reference
[1]	n/a	CFP Reference CEF-SESAR-2018-1, “Finnish-Estonian "Gulf of Finland" Very Large U-Space Demonstration”
[2]	Advanced Edition (unedited)	ICAO Doc 10039, Manual on System Wide Information Management (SWIM) Concept
[3]	00.05.00	SESAR 2020 GOF USPACE FIMS Design and Architecture, Appendix A Service Description Templates, document SESAR 2020 GOF USPACE Service Documentation Guidelines
[4]	Ed. 00.02.RC1, 1 March 2019	EUROCONTROL Concept of Operations for U-space (CORUS), D6.2, Grant Ref. 763551, Call Ref. 2016 SESAR 2020 RPAS Exploratory Research Call (H2020 – SESAR -2016-1), Release Candidate 1
[5]	n/a	Global UTM Association (GUTMA) Flight Logging Protocol, https://github.com/gutma-org/flight-logging-protocol/blob/master/Flight_logging_protocol.md
[6]	n/a	Global UTM Association (GUTMA) Air Traffic Protocol, https://github.com/hrishiballal/airtraffic-data-protocol-development
[7]	V1.0	Federal Aviation Administration NextGEN Concept of Operations, Foundational Principles, Roles and Responsibilities, Use Cases and Operational Threads, Unmanned Aircraft System (UAS), Traffic Management (UTM)
[8]	1.0	Federal Office of Civil Aviation (FOCA), Swiss U-Space ConOps, U-Space Program Management, 31.10.2018, FOCA muo / 042.2-00002/00001/00005/00021/00003
[9]	5th Ed. - 2016	ICAO Doc. 9750-AN/963, Global Air Navigation Plan (GANP) 2016-2030
[10]	0.61.1	Intel Corporation, Open Drone ID Message Specification, Draft Specification, November 13, 2018
[11]	n/a	SESAR, European ATM Master Plan: Roadmap for the safe integration of drones into all classes of airspace
[12]	n/a	SESAR, eATM PORTAL, European ATM Master Plan, https://www.atmmasterplan.eu/
[13]	2017	SESAR-JU, U-space Blueprint, https://www.sesarju.eu/u-space-blueprint
[14]	n/a	Efficient, safe and sustainable traffic at sea (EfficienSea2), a Horizon 2020 Project, Grant Agreement No 636329 https://efficiensea2.org https://efficiensea2.org/wp-content/uploads/2018/04/Deliverable-3.6.Standard-proposal-for-Maritime-Cloud-service-specification.pdf

[15]	n/a	IALA specification for e-navigation technical services https://www.iala-aism.org/product/g1128-specification-e-navigation-technical-services
[16]	Ed. 1.0	EUROCONTROL Specification for ATM Surveillance System Performance, EUROCONTROL-SPEC-0147, https://www.eurocontrol.int/publications/eurocontrol-specification-atm-surveillance-system-performance
[17]	1 November 2006	Federal Aviation Administration, Project Report ATC-323, Required Surveillance Performance Accuracy to Support 3-Mile and 5-Mile Separation in the National Airspace System

Table 58: List of References