|  |  |
| --- | --- |
|  | **FAA Office of Information Technology (AIT)** |

REST Service Description Document (RSDD)

for

Low Altitude Authorization and Notification Capability (LAANC)

and Third Party Providers

**Version 1.0**

**4/7/2017**

REST Service Description Document

LAANC

Approval Signatures

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Organization | Signature | Date Signed |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

REST Service Description Document

LAANC

Revision Record

|  |  |  |  |
| --- | --- | --- | --- |
| Revision  Letter | Description | Revision Date | Entered By |
| .1 | Initial version | 1/27/2017 | W.Larson |
| .2 | FIXM added | 2/23/2017 | A.Bacarra |
| .3 | Webhooks added | 2/27/2017 | A.Bacarra |
| .4 | Removed data model and data dictionary (will be in separate files) | 3/1/2017 | A.Bacarra |
| .5 | Added OAuth and TLS | 4/7/2017 | A.Bacarra |

Table of Contents

**Low Altitude Authorization and Notification Capability (LAANC)** i

**and Third Party Providers** i

1 Scope 1

1.1 Background 1

2 Applicable Documents 2

2.1 Government Documents 2

2.2 Non-Government Standards and Other Publications 2

3 Definitions 2

3.1 Terms and Definitions 2

3.2 Acronyms 2

4 Service Properties and Capabilities 3

4.1 Service Profile 3

4.1.1 Service Properties 3

4.1.2 Service Provider 3

4.1.2.1 Service Points of Contact 4

4.1.3 Service Consumers 4

4.1.4 Service Functionality 4

4.1.5 Service Security 4

4.1.6 API Protocol & Webhooks 4

4.1.6.1 API Protocol 4

4.1.6.2 Webhooks 5

4.1.6.3 Authentication 5

4.1.6.4 Communication security 5

4.1.7 Message Queuing Protocol 6

4.1.8 Message Format 6

4.1.8.1 UASs in FIXM 6

4.1.8.1.1 Data already covered in FIXM 7

4.1.8.1.1.1 Operator contact information 7

4.1.8.1.1.2 Flight plan 7

4.1.8.1.1.2.1 lat/lon 7

4.1.8.1.1.2.2 Time 8

4.1.8.1.1.2.3 Altitude 8

4.1.8.1.1.2.4 Proposal submission date 8

4.1.8.1.1.2.5 Proposal (FAA) response time 9

4.1.8.1.1.2.6 Closest airport 9

4.1.8.1.1.2.7 Flight Plan Version 10

4.1.8.1.1.2.8 Proposed Speed 10

4.1.8.1.1.3 ATC Authorization 10

4.1.8.1.1.3.1 planningStatus 10

4.1.8.1.1.3.2 operationalAcceptability 11

4.1.8.1.2 5.2. Data not yet covered in FIXM 11

4.1.8.1.2.1 Potential Extensions to FIXM 11

4.1.8.1.2.1.1 Permissions 11

4.1.8.1.2.1.2 UAS operator 12

4.1.8.1.2.1.3 airClass 13

4.1.8.1.2.1.4 Flight Radius 13

4.1.8.1.2.1.5 VLOS Indicator 13

4.1.8.1.2.1.6 Operation over people indicator 13

4.1.9 Qualities of Service 14

4.2 Service Interfaces & Data Model 14

4.2.1 Resources 14

4.2.2 Methods 16

4.2.3 Resource Representations 17

4.2.3.1 <Resource Name> 17

4.2.4 Sample Payloads 17

4.2.4.1 <Sample Payload 1> 17

4.3 Service Implementation 19

4.3.1 Endpoints 19

4.3.1.1 <Endpoint 1 Name> 20

4.3.1.1.1 Supported Data Formats 20

4.3.1.1.2 Network Address 20

4.3.1.1.3 End Point-Specific Qualities of Service 20

5 Appendix A Example FIXM messages 20

5.1 A.1 Operator to TPP: Proposed Flight Plan 20

5.2 A.2 TPP to Operator: Accept/Reject. 25

5.3 A.3 TPP to ATC: Notification 26

5.4 A.4 TPP to ATC: Authorization Request 27

5.5 A.5 ATC to TPP: Accept – Authorization Granted 28

5.6 A.6 ATC to TPP: Reject with Minor Modifications 29

5.7 A.7 ATC to TPP: Reject 30

6 Appendix B Message Collections 30

# Scope

This REST Service Description Document (RSDD) provides information to describe and document the LAANC service, which has been designed using the Representational State Transfer (REST) architectural style for services. In this style, requests and responses between clients and servers focus on the transfer of representations of resources. A resource can be any entity identifiable by a Uniform Resource Identifier (URI), about which information can be exchanged. A representation of a resource is typically a document that captures the current or intended state of a resource.

This document provides basic information about the service and its provider, and describes in detail the resources that the service can access and the RESTful Hypertext Transfer Protocol (HTTP) methods that the service supports for each resource.

## Background

The Federal Aviation Administration (FAA) has been tasked with implementing notification and authorization (N&A) processes as defined in the Federal Aviation Regulations, Parts 101 and 107 respectively. These new regulations provide the needed rules to govern the use of both model aircraft and unmanned aircraft system (UAS) flight operations (unmanned aviation activities) throughout the airspace governed by the FAA.

The FAA is in the process of determining its approach and business plan to integrate UAS into the National Airspace System (NAS). As part of that approach, the FAA is dedicated to ensuring safety requirements are met for integration of UAS into the NAS, where UAS are able to operate safely in the same airspace with manned aircraft. To this end, the FAA must ensure that integrated UAS operations meet appropriate performance standards and access requirements (e.g., SC-228, Minimum Operational Performance Standards for Unmanned Aircraft Systems). The FAA seeks to ensure reduced barriers to access and to avoid monopolization of public resources. The FAA challenge is to foster equitable access for all users and providers while ensuring critical air traffic control (ATC) technical and safety requirements are met for NAS operations. In addition, FAA seeks to foster a competitive environment for providers of UAS and related services. As the FAA and industry move toward integration of all types of UAS into the NAS, two rules have recently been introduced governing the requirements for small UAS (sUAS), defined as UAS that weigh less than 55 pounds.

The development of a fully functioning and streamlined, user friendly N&A process is complex and subject to a variety of inputs and coordination points across the UAS community landscape. This document will give stakeholders and leadership the necessary background and contextual information to understand and provide input on the FAA’s LAANC demonstration planned in 2017. It is expected that this document will provide stakeholders an understanding of the FAA’s approach to working with industry to develop streamlined processes for model aircraft and UAS stakeholders to comply with the law and conduct operations safely.

The LAANC Third Party Providers (TPP) will send authorization, notification, re-consideration requests, re-consideration answers, and denial records to the FAA for FAA display. The TPP will manage Part 107 operator secondary requests for authorization once automatic denial has been provided.

# Applicable Documents

The documents or information sources in the following sub-sections are relevant to this document:

* Federal Aviation Administration, Advisory Circular: 107-2, Small Unmanned Aircraft Systems (sUAS), 21 June 2016
* Federal Aviation Administration, *“Integration of Unmanned Aircraft Systems into the National Airspace System, Concept of Operations v2.0”*, September, 2012
* U.S. Government Publishing Office, Code of Federal Regulations, Title 14, Part 107, Small Unmanned Aircraft Systems, Web 29 December 2016
* Airman’s Information Manual (AIM)

## Government Documents

* <SLA document(s) for the Service, when completed>
* FAA-STD-066, Web Service Taxonomies

## Non-Government Standards and Other Publications

* Authentication Protocol: <https://oauth.net/2/>
* Access Protocol: <https://wadl.java.net/>
* Message Queuing Protocol: <https://www.amqp.org/>
* Message Format: <http://www.json.org/>

# Definitions

## Terms and Definitions

| Name | Definition |
| --- | --- |
| REST Service | A RESTful web service (also known as a RESTful web Application Programming Interface (API)) is a service implemented using HTTP and REST principles. |
| Resource | A resource is an individual data entity that is identifiable by a URI. |

## Acronyms

| Name | Full Spelling |
| --- | --- |
| AIT | Office of Information Technology |
| API | Application Programming Interface |
| FAA | Federal Aviation Administration |
| HTTP | Hypertext Transfer Protocol |
| JSON | JavaScript Object Notation |
| REST | Representational State Transfer |
| RSDD | REST Service Description Document |
| SDLC | System Development Lifecycle |
| SLA | Service Level Agreement |
| SOA | Service Oriented Architecture |
| SSA | Service Security Agreement |
| URI | Uniform Resource Identifier |

# Service Properties and Capabilities

## Service Profile

### Service Properties

| Profile Item | Value |
| --- | --- |
| Service Name |  |
| Registered FAA Namespace |  |
| Description |  |
| Version |  |
| Service Category (per FAA-STD-066) |  |
| Lifecycle Stage (per FAA-STD-066) |  |
| Criticality for the Service (per FAA-STD-066) |  |

### Service Provider

|  |  |
| --- | --- |
| Organization Name | FAA ADE-320 |
| Description | FAA, Solution Delivery |
| Web Page URL |  |

#### Service Points of Contact

| POC Function | Name | Org | Phone | Email |
| --- | --- | --- | --- | --- |
| AIT SOA lead | Jill Longenecker | ADE-320 |  | Jill.Longenecker@faa.gov |
| Manager, EIM | Robert Fernandez | ADE-220 | 301.427.5085 | Robert.Fernandez@faa.gov |
| EIM | Wayne Larson | ADE-220 | 202.267.7210 | Wayne.Larson@faa.gov |

### Service Consumers

Specific consumers of this service are managed by the AIT SOA team and its associated infrastructure and governance processes. Contact the AIT SOA team for more information about the current consumers of this service.

### Service Functionality

| Business Function | Practical Effect |
| --- | --- |
| UAS Flight Notification | * FAA receives record of a voluntary notification from TPP (class G airspace) * FAA receives record of a notification from TPP (class C airspace) |
| UAS Flight Authorization | * FAA receives record of a authorization from TPP (class B and below ATC preapproved altitude) |
| UAS Flight Re-Consideration | * FAA receives record from TPP of a reconsideration request that was made from a UAS operator (class B above pre-approved sUAS altitude |
| UAS Flight Re-Consideration response | * FAA receives record from TPP of a reconsideration response given from ATC |
| UAS Flight Denial | * FAA receives denial record from TPP (for all denied flights) |

### Service Security

Application authentication through OAuth-2.0 (<https://oauth.net/2/>)

### API Protocol & Webhooks

#### API Protocol

APIs shall be REST APIs that support Web Application Description Language (WADL) (https://wadl.java.net/) documentation

#### Webhooks

Webhooks shall play an important role within the system. Below is a description of Webhooks.

A webhook is an event triggered mechanism of a web application to deliver data. Also called a web callback or HTTP push API, a webhook is a way for an app to deliver data to other applications as events occur. An example of an event that triggers a webhook is a user submitting a web form. Therefore, webhooks provide the ability to immediately react respond to given events. Webhooks are intended to primarily receive data and respond accordingly or receive data, process it and pass it on.

A Webhook is a Post request (HTTP callback) that is submitted to a URL. The designated URL is configured to receive the Post and process it in a specific way. When a web application implements webhooks, users are given the ability to create their own URLs to integrate their app with the web application implementing the webhook. Data from webhooks are either in JSON or XML format or transmitted as form data.

APIs are an alternate mechanism to webhooks for application interaction. API excel at exchanging data between separate applications. However, APIs are more cumbersome when change notifications are involved. That is where webhooks come into play. Webhooks are better suited to react to system changes. API require constant monitoring of a system in order to react systems change. This can involve more computing resources than is necessary. Because webhooks operate with an event-based mechanism, webhooks are simpler and more efficient in this case. The key is the configuration of the webhook. Webhooks are ideal with the handling of real-time notifications of events.

#### Authentication

OAuth 2.0 will be used as the protocol for authorization in LAANC.

#### Communication security

TLS 1.2 will be used as the protocol for communication security in LAANC.

### Message Queuing Protocol

APIs shall support the Advanced Message Queuing Protocol (AMQP) (https://www.amqp.org/) message queuing protocol

### Message Format

APIs shall format data as XML (<https://www.w3.org/XML/Core/#Publications)>

#### UASs in FIXM

Within this document, we discuss the data which is required to be transmitted to and from air traffic control and UAS operators. It is based upon the Low Altitude Authorization and Notification Capability (LAANC) data model and the LAANC Concept of Operations document. Specifically, we outline those data which may be expressed using the core of the FAA standard messaging framework (Flight Information eXchange Model; FIXM). For the data which cannot be expressed with the FIXM core framework, we propose extensions to current FIXM data classes which may encompass necessary UAS data.

UASs are not regular, commercial aircraft. Nor are they model aircraft. Rule 107 encompasses proposed regulations on UAS operation for commercial and recreational use. The nature of UASs require data not yet present within current FIXM schema. We follow the LAANC proposed data model, and discuss parts of the data model which fit into the existing FIXM core schema and parts of the data model which will require extensions.

Most of the data needed to track and regulate UASs exists already in FIXM. UASs are piloted aircraft, after all. As such, a flight plan may be submitted either to notify flights nearby or to request authorization to fly in certain regions.

The operator must provide their contact information: postal address, phone number, email.

Any proposed flight plan will consist of one or more points, for which the geographical location (latitude and longitude), the proposed time of the flight, the altitude, and the flight radius needs to be given. A flight may only be performed within visible line of sight during daytime and twilight hours (with proper lights), so a flight proposal class will have permissions or indicators. LAANC includes both the proposed altitude of the flight and the flight radius. **Note: it is advised to maintain using decimal degrees to specify the latitude/longitude location. The sexagesimal standard has limited resolution (0.1 km), which may create conflicts between multiple UAS flights. Otherwise, when specifying the nearest airport, we require both a distance AND a bearing, to pinpoint the location.**

The TPP should send a message to the operator via the front end interface. The TPP should also generate a FIXM-formatted message containing the same information to be recorded by ATC for reference by other operators/ATC. The FIXM format is required for consistency, to ease any subsequent analysis of the messages (require less code to input all the messages) and to boost readability.

##### Data already covered in FIXM

###### Operator contact information

<contact>

<address>

…

</address>

<onlineContact>

…

</onlineContact>

</contact>

Most of the contact information necessary is already covered in FIXM. The XML fields listed above cover two contact methods. Phone/fax number may also be given. For automated response messages, the email should be used, or else a third party provider should take the response and send a notification to the operator via email or mobile message/SMS (whichever option is covered and preferred). If the TPP has a mobile app or web interface, notifications can be sent this way.

###### Flight plan

We cover latitude, longitude, time, altitude, proposal submission datetime, accept/reject datetime, closest airport, flight plan version, and proposed speed. In the following, fb refers to the FIXM base namespace, and fx refers to the FIXM flight namespace. Setting these references can be found in the appendices, where (ns2 = fb) and (ns3 = fx).

lat/lon

Specify a flight route as a single segment with a single point of reference

<fx:point>

    <fb:location srsName="urn:ogc:def:crs:EPSG::4326">

        <fb:pos> </fb:pos>

    </fb:location>

</fx:point>

The srsName is the coordinate reference system. The FIXM documentation states, FIXM uses only "urn:ogc:def:crs:EPSG::4326", which refers to the WGS84 standard. If coordinates are input in a different coordinate reference system, the third party provider must convert to WGS84 coordinates for the FIXM message. However, this may be modified in the future to facilitate further coordinate updates and international communication.

There are several ways of specifying location. E.g. referencePoint or significantPoint, which may be used to specify a location within the flight plan or the location of the departure/arrival aerodrome. The point4D or trajectoryPoint4D. The 4D point specifies the latitude, longitude, altitude, and time. These data may also be specified elsewhere.

Time

Covers planned launch/landing times.

<fx:estimatedDepartureTime>dateTime</fx:estimatedDepartureTime>

<fx:estimatedArrivalTime>dateTime</fx:estimatedArrivalTime>

Inputs are dateTime objects in Greenwich Mean Time.

Altitude

<fx:level>

    <fx:altitude ref="SFC" uom="FT">200</fx:altitude>

</fx:level>

The ref (reference: SFC = above ground level, MSL = above mean sea level, STD = barometric altitude from standard 1 atm pressure, and W84 = height above WGS84 ellipsoid) and uom (Unit Of Measure; can be “FT” or “M”) attributes are required.

Proposal submission date

When the proposal was submitted. The accepted class is a receipt that the ATC received the proposal.

<fx:filing>

    <fx:accepted></fx:accepted>

    <fx:filingTime></fx:filingTime>

</fx:filing>

Accepted field may be “ACCEPTED”, “REJECTED”, or “MANUAL”. The last choice represents a negotiation between the operator and air traffic control. The filingTime field must be a datetime object.

Proposal (FAA) response time

<fx:flightPlanNegotiationStatus>

    <fx:operationalAcceptability status="" statusReason=""></…>

    <fx:planningStatus status=”” statusReason=””></fx:planningStatus>

</fx:flightPlanNegotiationStatus>

Under operationalAcceptability, the status may be “Acceptable” or “Not Acceptable”, and the statusReason is a textual description for the decision. The planningStatus status attribute may be “CONCUR”, “NON\_CONCUR”, or “NEGOTIATE”, and the statusReason attribute is a textual description of the decision.

These decisions may be automated, based upon regulation as determined by air traffic control or a third party provider. If so, then statusReason may simply quote the regulations violated by the flight plan, and the result will require the operator to submit a revised flight plan.

Closest airport

Use the nearest airport as a reference point in departure aerodrome. We assume arrival aerodrome is the same as the departure; only the time will differ.

<fb:otherReference iataDesignator="" name="">

    <fb:referencePoint>

        <fb:distance uom:”FT”></fb:distance>

        <fb:pos></fb:pos>

    </fb:referencePoint>

</fb:otherReference>

UASs flying within 5 miles of an airport require notification (not necessarily authorization). May also be used to determine the message recipient.

Flight Plan Version

The flight plan version, which may go under the "flight" data class, details the version of the flight plan under consideration. This allows for a more extensive negotiation related to the flight plan, if the ATC determines that slight modifications to the flight plan are necessary to grant a waiver or authorization. The operator, TPP, and ATC all require this data.

<version>

    <aspFlightPlanVersion><!-- Unique ID for flight plan. Given by ASP. --> </aspFlightPlanVersion>

    <auFlightPlanVersion><!-- Unique ID for flight plan. Given by airspace user (operator). --> </auFlightPlanVersion>

</version>

ASP = ?, AU = airspace user. Only one or the other need to be given; the TPP may be considered an ASP for these purposes when relaying messages to/from ATC and the operator.

Proposed Speed

The proposed (maximum) speed of the flight.

<fx:airspeed uom="KM\_H">10</ns3:airspeed>

This specified the maximum or proposed average airspeed of the flight. It is called within a single segment. The the case of a flight specified by waypoints, each segment (the space connecting each waypoint) has a proposed speed, or else the speed applies equally to each segment. The units may be KM\_H for kilometers per hour, KT for knots, or MACH for mach number.

###### ATC Authorization

Latitude, longitude, time, altitude, flight start, flight end (times), time of decision, and flight plan version. These data in the FAA authorization message are covered above in the flight plan data. ATC authorization will mostly send flightPlanNegotiationStatus. This data class allows ATC to give a full-text explanation for acceptance/rejection. Within flightPlanNegotiationStatus, there are objects for planningStatus and operationalAcceptability, which give reasons for the status of the proposed route and the status of the message, respectively.

planningStatus

This class is optional, and outlines the status of the message. This is depreciated in favor of 5.1.3b.

<operationalAcceptability status="" statusReason=""/>

status is either 'Ack' (acknowledge), 'Accept', or 'Reject'. The statusReason is a string of indeterminate length explaining the status. It will be necessary for a 'Reject' status. A reason may or may not be given for an acknowledge/accept message.

operationalAcceptability

This class is for a response to a proposed flight plan and outlines the status of the proposed flight plan.

<planningStatus status="" statusReason=""/>

The status attribute may be 'Ack', 'Accept', or 'Reject'. The statusReason is a string of indeterminate length explaining the status. It will be necessary for a 'Reject' status. A reason may or may not be given for an acknowledge/accept message.

##### 5.2. Data not yet covered in FIXM

* Operator: UAS class/code, descriptions
* Aircraft: make, model, registration number
* Flight plan: flight radius, waivers (over people indicator, airspeed, altitude, etc.), description, airspace class, nearest airport
* FAA Authorization: waivers (airspace, time, etc.), “ATC approved specific conditions”, authorization acknowledgement & time.
* Airspace class (needs to be attached to the flight plan, not the operator; add into point, region classes?). And airspace permission is for a single flight.
* VLOS and operation over people indicators.

###### Potential Extensions to FIXM

In this document, we do not give the XML schema which will define these data types. However, we give sample XML code and potential inputs which help clarify the necessary data types and their relationships.

Permissions

Class, within operator details -- specifies previous permissions held. This is an optional class, and there can be any number of permissions given for a particular operator. The waiver indicators should default to False (0), and the airspaceClassPermission should default to G, prior to air traffic control response and modification.

<fb:permissions>

   <fb:airspaceClassPermission>G</fb:airspaceClassPermission>

   <fb:permissionWaiverID>\_\_\_\_</fb:permissionWaiverID>

   <fb:permissionWaiverSignatory>\_\_\_\_</fb:permissionWaiverSignatory>

   <fb:permissionAcceptDate></fb:permissionAcceptDate>

   <fb:permissionStartDate></fb:permissionStartDate>

   <fb:permissionEndDate></fb:permissionEndDate>

    <fx:operationOverPeople>True</fx:operationOverPeople>

   <fx:altitudeWaiver>True/False</fx:altitudeWaiver>

   <fx:VLOSwaiver>True/False</fx:VLOSwaiver>

   <fx:speedWaiver>True/False</fx:speedWaiver>

   <!-- add whichever other waivers are necessary -->

</fb:permissions>

Required:

permissionWaiverID, permissionWaiverSignatory (ID of who authorized waiver, characterType), permissionAcceptDate (date of authorization, datetime type), permissionStartDate (date the authorization begins, datetime type), permissionEndDate (date the authorization expires, datetime type)

Optional:

The authorization indicators are optional. The airspace class permission defaults to G, and every other permission indicator defaults to False. The following indicate which rules the authorization cover

airspaceClassPermission (character: B, C, D, E, G), overPeople (boolean), altitudeWaiver (boolean), VLOSwaiver (boolean), speedWaiver (boolean).

UAS operator

Class: covers the operator class code and operator type, specified in LAANC data model.

<fb:UASOperator>

   <fb:classCode ns2:uasCode="">Description</fb:classCode>

   <fb:uasOperatorType ns2:uasType="">Description</fb:uasOperatorType>

</fb:UASOperator>

Both are required for commercial flights and accepts all characters (from CharacterStringType). UAS operator type may be COMMERCIAL, PUBLIC, HOBBYIST, or OTHER. The OTHER would require a description, COMMERCIAL would also require a description, and PUBLIC and HOBBYIST would not require a description.

airClass

Attribute, within the region class. Points may have FAA designations to help identify where the aircraft is. Now, each point also has an airspace classification. This proposed attribute is optional, and mostly applicable to UASs.

<fx:region airClass="G"></fx:region>

airClass (optional) = {‘B’, ‘C’, ‘D’, ‘E’, ‘G’, ‘R’, ‘T’}, where ‘R’ = restricted, ‘T’ temporarily unrestricted for certain types of activity (after checking the NOTAMs). Input is optional, and may be a ICAO location designation (e.g., airport code) and accepts CharacterStringType.

Flight Radius

Class. A proposed flight may only fly up to 400 ft (without a waiver), however, the average person can technically resolve a 2 ft object at more than half a mile. An operator must maintain enough visual resolution of the aircraft in order to maintain operations (including seeing orientation [yaw, pitch, roll]). It is not clear how far this can be, but let us be more generally allow for a flight ellipsoid: define two distances (altitude [vertical] & flight radius [horizontal]).

<fx:flightRadius uom="FT">200</fx:flightRadius>

Unit of measure = feet or meters, specify. Input is a lengthType.

VLOS Indicator

A proposed flight is required to operate within visible line-of-sight of the operator. This data is a boolean, indicating whether the operator will maintain visible line-of-sight for the duration of the flight. If FALSE, then the ATC may grant a waiver after clarifications.

<fx:VLOSIndicator>True/False</ns3:VLOSIndicator>

Operation over people indicator

A proposed flight is not allowed to operate over people (though I do not know what the actual definition is). The datum is a boolean, indicating whether the UAS is going to operate over people. A waiver may be granted by ATC (e.g., news casting).

<fx:operationOverPeople>True/False</fx:operationOverPeople>

### Qualities of Service

Qualities of Service (QoS) is addressed by Service Level Agreements (SLAs) developed for this service.

## Service Interfaces & Data Model

REST is a style of software architecture that provides a convenient and consistent approach to requesting and modifying data. In the context of FAA REST services, it refers to using HTTP verbs to retrieve and modify representations of data stored by FAA.

In a RESTful system, resources are stored in a data store; a client sends a request that the server perform a particular action (such as creating, retrieving, updating, or deleting a resource), and the server performs the action and sends a response, often in the form of a representation of the specified resource.

In RESTful services, the client specifies an action using an HTTP verb such as POST, GET, PUT, or DELETE. It specifies a resource by a globally-unique URI of the following form:

http://{ServiceHost}.faa.gov/{serviceName}/{serviceVersion}/{resourcePath}?{parameters}

### Resources

A resource is an individual data entity with a unique identifier. A REST service can operate on one or more types of resources based on that service’s data model.

A service data model can be based on groups of resources, called collections.

|  |  |
| --- | --- |
| **Class Name** | **Class Description** |
| Airport | Vendor created table containing all Airport information obtained from FAA. |
| Airspace | Vendor created table   containing all Airspace information   obtained from FAA.  Class B, C, D, E & G Airspace    The TPP planning tool would also depict features relevant to UAS flight such as classes of airspace, any active temporary flight restriction (TFR), obstacles, or other restricted airspace (e.g., public utilities).     Each TPP will use authorized FAA airport, SUA, and locality map data to automatically provide, where feasible, confirmations of notification and authorizations to UAS operators.   Class E Airspace  TFR Airspace  Restricted Airspace |
| Airspace Authorization | Airspace Authorization Association  The Assumption is that an Authorized UAS operation may occur in multiple Airspaces.  Either transitioning from one Airspace to another or operating in overlapping Airspaces. |
| Authorization | Authorizations are the result of data sent from ATC to an operator regarding a specific request received asking permission to operate in a particular airspace. Authorizations in the context of LAANC shall not be confused with ATC permissions provided in-flight via radio between a pilot in command and ATC to enter airspace requiring two-way communication with ATC. |
| Proposed Operation | Flight Specifics of Proposed Operation |
| Proposed Operation Airspace | Proposed Operation Airspace Association  The Assumption is that a proposed UAS operation may occur in multiple Airspaces.  Either transitioning from one Airspace to another or operating in overlapping Airspaces. |
| Reference Request Type | Reference Request Type |
| Reference Timeframe | Reference Timeframe |
| Request | A request is the result of data sent from a UAS operator to ATC providing key parameters about an operation which must be approved or denied.   A Request may also be for Notification purposes only where no authorization is required. |
| UAS Operator Class | UAS Operator Class |
| UAS Operator Type | UAS Operator Type |

### Methods

The following table lists the HTTP methods that are generally applicable to the LAANC RESTful services.

| Method | Description | REST HTTP Mappings |
| --- | --- | --- |
| Get | Gets a specific resource or lists a specified set of resources | GET on resource URI |
| Put | Updates a specific resource | PUT on resource URI, where you pass in data for the updated resource |
| Post | Creates a resource |  |
| Delete | At present, there is no intent to delete in this system |  |

Of the candidate methods in the table above, the following table indicates which HTTP methods are supported for each resource.

| Resource Name | Supported Methods | | | |
| --- | --- | --- | --- | --- |
|  | Get | Post | Put | Delete |
| Airport | Yes | Yes | Yes | No |
| Airspace | Yes | Yes | Yes | No |
| Airspace Authorization | Yes | Yes | Yes | No |
| Authorization | Yes | Yes | Yes | No |
| Proposed Operation | Yes | Yes | Yes | No |
| Proposed Operation Airspace | Yes | Yes | Yes | No |
| Reference Request Type | Yes | Yes | Yes | No |
| Reference Timeframe | Yes | Yes | Yes | No |
|  |  |  |  |  |

### Resource Representations

This section define how each resource is represented.

<For each resource for the service, include a subsection below describing the resource representation.>

#### <Resource Name>

The data dictionary is now a separate document in the FAA LAANC Github repository. “LAANC (Class) Data Dictionary 2017-02-27.xlsx”

### Sample Payloads

<For each data format supported, include a subsection below (entitled, for example, “Sample XML Payload”, “Sample JSON Payload”) with one or more sample payloads in each format. If the payloads differ significantly across resources, include samples for all such resources.>

#### <Sample Payload 1>

<Sample Payload for the Service in format 1.>

<Request>

<Phone\_Number></Phone\_Number>

<Request\_Type\_Code></Request\_Type\_Code>

<Request\_Status\_Acknowledged\_Indicator></Request\_Status\_Acknowledged\_Indicator>

<Request\_Status\_Acknowledged\_Timestamp></Request\_Status\_Acknowledged\_Timestamp>

<Request\_Status\_Authorized\_Indicator></Request\_Status\_Authorized\_Indicator>

<Request\_Status\_Authorized\_Timestamp></Request\_Status\_Authorized\_Timestamp>

<Request\_Status\_Denied\_Indicator></Request\_Status\_Denied\_Indicator>

<Request\_Status\_Denied\_Timestamp></Request\_Status\_Denied\_Timestamp>

<ATC\_Denied\_Comments></ATC\_Denied\_Comments>

<Request\_Status\_Terminated\_Indicator></Request\_Status\_Terminated\_Indicator>

<Request\_Status\_Terminated\_Timestamp></Request\_Status\_Terminated\_Timestamp>

<Request\_Hazard\_Indicator></Request\_Hazard\_Indicator>

<Notification\_Required\_Indicator></Notification\_Required\_Indicator>

<Authorization\_Required\_Indicator></Authorization\_Required\_Indicator>

<Prior\_Waiver\_Number></Prior\_Waiver\_Number>

<Prior\_Authorization\_Number></Prior\_Authorization\_Number>

<Prior\_Letters\_Of\_Agreement\_Number></Prior\_Letters\_Of\_Agreement\_Number>

<UAS\_Operator\_Class\_Code></UAS\_Operator\_Class\_Code>

<UAS\_Operator\_Type\_Code></UAS\_Operator\_Type\_Code>

<Notification\_Acknowledgement\_Indicator></Notification\_Acknowledgement\_Indicator>

<Nearest\_Airport></Nearest\_Airport>

<UAV\_Registration\_Number></UAV\_Registration\_Number>

<Third\_Party\_Provider\_Key></Third\_Party\_Provider\_Key>

<Create\_Timestamp></Create\_Timestamp>

</Request>

<Proposed\_Operation>

<Operation\_Description></Operation\_Description>

<Timeframe\_Code></Timeframe\_Code>

<Maximum\_Altitude></Maximum\_Altitude>

<Latitude\_Degrees></Latitude\_Degrees>

<Latitude\_Minutes></Latitude\_Minutes>

<Latitude\_Seconds></Latitude\_Seconds>

<Latitude\_Direction></Latitude\_Direction>

<Longitude\_Degrees></Longitude\_Degrees>

<Longitude\_Minutes></Longitude\_Minutes>

<Longitude\_Seconds></Longitude\_Seconds>

<Longitude\_Direction></Longitude\_Direction>

<VLOS\_Indicator></VLOS\_Indicator>

<Operations\_Over\_People\_Indicator></Operations\_Over\_People\_Indicator>

<Flight\_Radius></Flight\_Radius>

<Flight\_Start\_Timestamp></Flight\_Start\_Timestamp>

<Flight\_End\_Timestamp></Flight\_End\_Timestamp>

<Operation\_Status\_Denied\_Indicator></Operation\_Status\_Denied\_Indicator>

<Operation\_Status\_Denied\_Timestamp></Operation\_Status\_Denied\_Timestamp>

<Create\_Timestamp></Create\_Timestamp>

</Proposed\_Operation>

<Authorization>

<Authorization\_FAA\_Acknowledgement\_Indicator></Authorization\_FAA\_Acknowledgement\_Indicator>

<Authorization\_FAA\_Acknowledgement\_Timestamp></Authorization\_FAA\_Acknowledgement\_Timestamp>

<ATC\_Approved\_Specific\_Conditions></ATC\_Approved\_Specific\_Conditions>

<Timeframe\_Code></Timeframe\_Code>

<Maximum\_Altitude></Maximum\_Altitude>

<Latitude\_Degrees></Latitude\_Degrees>

<Latitude\_Minutes></Latitude\_Minutes>

<Latitude\_Seconds></Latitude\_Seconds>

<Latitude\_Direction></Latitude\_Direction>

<Longitude\_Degrees></Longitude\_Degrees>

<Longitude\_Minutes></Longitude\_Minutes>

<Longitude\_Seconds></Longitude\_Seconds>

<Longitude\_Direction></Longitude\_Direction>

<VLOS\_Indicator></VLOS\_Indicator>

<Operations\_Over\_People\_Indicator></Operations\_Over\_People\_Indicator>

<Flight\_Radius></Flight\_Radius>

<Flight\_Start\_Timestamp></Flight\_Start\_Timestamp>

<Flight\_End\_Timestamp></Flight\_End\_Timestamp>

<Create\_Timestamp></Create\_Timestamp>

</Authorization>

## Service Implementation

### Endpoints

The following subsections provide information about the base service endpoints that are used to reference and access the resources.

<For each base service endpoint, include a subsection below describing the endpoint.>

#### <Endpoint 1 Name>

<Brief description of endpoint 1.>

##### Supported Data Formats

|  |  |
| --- | --- |
| Protocol: | XML |
| Description: | The Extensible Markup Language (XML) is a subset of SGML that is completely described in this document. Its goal is to enable generic SGML to be served, received, and processed on the Web in the way that is now possible with HTML. XML has been designed for ease of implementation and for interoperability with both SGML and HTML. |
| Specification Location: | https://www.w3.org/XML/Core/#Publications |

##### Network Address

The network address for this endpoint is:

http://{ServiceHost}.faa.gov/REST/<ServiceName.>Service.svc

##### End Point-Specific Qualities of Service

Please refer to the SLA document(s) for the service for QoS details.

The data model is now a separate document in the FAA LAANC Github repository. “LAANC (Class) 2017-02-27.pdf”

**Data Model**

# Appendix A Example FIXM messages

## A.1 Operator to TPP: Proposed Flight Plan

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<!-- In order to determine the airspace permissions, we need airspace codes and aircraft 4D location: latitude, longitude, altitude, and time. Only comment needs to be:  -->

<message xmlns="<http://www.fixm.aero/messaging/4.0>"  xmlns:xsi="<http://www.w3.org/2001/XMLSchema-instance>"

   xmlns:ns2="<http://www.fixm.aero/base/4.0>" xmlns:ns3="<http://www.fixm.aero/flight/4.0>"

   xsi:schemaLocation="<http://www.w3.org/2001/XMLSchema-instance> <http://www.fixm.aero/fixm/4.0/Fixm.xsd>">

<metadata>

   <gumi>urn:[fixm.aero](http://fixm.aero):asa:ods:20130203T022530:000001</gumi>

</metadata>

 <messageDateTime timeReference="UTC"></messageDateTime>

<flight>

   <ns3:gufi>au.asa.20130203T022530a.000001</ns3:gufi>

   <ns3:activeFlightPlan ns3:flightPlanIdentifier="1">

       <ns3:aircraftIdentity>

           <ns3:acid>FFFF</ns3:acid>

       </ns3:aircraftIdentity>

       <ns3:aircraftDescription>

           <ns3:capabilities ns3:standard="STANDARD">

               <ns3:communication>

                   <ns3:communicationCodes>N</ns3:communicationCodes>

                   <ns3:dataLinkCodes></ns3:dataLinkCodes>

                   <ns3:selectiveCallingCodes></ns3:selectiveCallingCodes>

               </ns3:communication>

                <ns3:surveillance>

                   <ns3:surveillanceCodes>N</ns3:surveillanceCodes>

               </ns3:surveillance>

               <ns3:navigation>

                   <ns3:navigationCodes>N</ns3:navigationCodes>

                   <ns3:performanceBasedCodes></ns3:performanceBasedCodes>

               </ns3:navigation>

           </ns3:capabilities>

           <ns3:registration>\_\_\_</ns3:registration>

               <!-- aircraftCategory is new. The make/model of a UAS must be

                       included in a waiver request. It is provided here under

                       the aircraft description.

               -->

           <ns3:aircraftCategory>

               <ns3:model>\_\_\_</ns3:model>

               <ns3:manufacture>\_\_\_</ns3:manufacture>

           </ns3:aircraftCategory>

       </ns3:aircraftDescription>

       <ns3:formationAircraftCount>1</ns3:formationAircraftCount>

       <ns3:filing>

           <ns3:accepted>ACCEPTED</ns3:accepted>

           <ns3:filingTime>2013-02-03T02:25:30.890Z</ns3:filingTime>

       </ns3:filing>

       <ns3:departure>

       <!-- Already included in FIXM: reference Aerodrome. This includes

               geographical location of reference point (nearest aerodrome)

               and the distance between reference point and operator.

        -->

           <ns2:OtherReference iataDesignator="" name="">

               <ns2:referencePoint>

                   <ns2:distance uom="FT">\_\_</ns2:distance>

                   <ns2:pos>\_\_ \_\_</ns2:pos>

               </ns2:referencePoint>

           </ns2:OtherReference>

          <ns3:estimatedDepartureTime>

               2013-02-02T04:00:00.000Z

           </ns3:estimatedDepartureTime>

       </ns3:departure>

       <ns3:arrival>

           <ns3:estimatedArrivalTime>

               2013-02-02T05:00:00.000Z

           </ns3:estimatedArrivalTime>

       </ns3:arrival>

       <ns3:route>

           <ns3:text>2704N11627W</ns3:text>

           <ns3:segment xsi:type="ns3:BasicSegmentType" xmlns:xsi="<http://www.w3.org/2001/XMLSchema-instance>">

               <ns3:level>

                   <ns3:altitude ref="SFC" uom="FT">200</ns3:altitude>

               </ns3:level>

               <ns3:airspeed uom="KM\_H">10</ns3:airspeed>

               <ns3:airway xsi:nil="true"/>

               <ns3:point>

                   <ns2:location srsName="urn:ogc:def:crs:EPSG::4326">

                       <!-- How many decimal places are valid? Is there a standard? -->

                       <ns2:pos>27.066667 -116.453328</ns2:pos>

                   </ns2:location>

               </ns3:point>

               <!-- Proposed extension specifies the flight radius. When

                       combined with the altitude, this forms an semi-

                       ellipsoid, centered at ground level.

               -->

               <ns3:flightRadius uom="FT">200</ns3:flightRadius>

               <!-- Proposed extension is the "airClass" attribute to the

                       region class. This specifies the airspace classification

                       of the proposed flight, and helps determine the

                       accept/reject status of the proposed flight.

               -->

               <ns3:region airClass="G"></ns3:region>

               <fx:VLOSIndicator>True</ns3:VLOSIndicator>

               <fx:operationOverPeople>True/False</fx:operationOverPeople>

           </ns3:segment>

       </ns3:route>

       <ns3:flightType>SCHEDULED</ns3:flightType>

   </ns3:activeFlightPlan>

   <version>

       <aspFlightPlanVersion><!-- Unique ID for flight plan. Given by ASP. --> </aspFlightPlanVersion>

       <auFlightPlanVersion><!-- Unique ID for flight plan. Given by airspace user (operator). --> </auFlightPlanVersion>

   </version>

</flight>

</message>

<recipient>

     <!-- who gets the message. See contact, under operating Organization. Recipient = TPP →

</recipient>

<messageOriginator>

    <!-- Information about who sent this message. See contact, under operating Organization. Originator = operating organization. →

</messageOriginator>

<ns3:operatingOrganization>

   <!-- UAS Operator class, includes the operator class code and the

           type and the respective descriptions.

   -->

   <ns2:UASOperator>

       <ns2:classCode ns2:uasCode="">Description</ns2:classCode>

       <ns2:uasOperatorType ns2:uasType="">Description</ns2:uasOperatorType>

   </ns2:UASOperator>

   <ns2:contact name="B" title="Dr.">

       <ns2:address>

           <ns2:administrativeArea>STATE/PROVINCE</ns2:administrativeArea>

           <ns2:city>CITY NAME</ns2:city>

           <ns2:countryCode>USA</ns2:countryCode>

           <ns2:countryName>United States of America</ns2:countryName>

           <ns2:deliveryPoint>StreetAddress</ns2:deliveryPoint>

           <ns2:postalCode>10001</ns2:postalCode>

       </ns2:address>

       <ns2:onlineContact>

           <ns2:email>\_\_\_@[ata-llc.com](http://ata-llc.com)</ns2:email>

       </ns2:onlineContact>

       <ns2:phoneFax>

           <ns2:voice>PhoneNumber</ns2:voice>

       </ns2:phoneFax>

   </ns2:contact>

   <!-- Proposed extension to operatingOrganization or organization class

       (or both): permissions. This lists the permissions the organization

       already has. This helps determine the accept/reject status of the

       proposed flight.

   -->

    <!-- If, for example, the operator has a overPeople waiver, the following

                 permission would be granted.

    -->

    <ns2:permissions>

       <ns2:permissionWaiverID>\_\_\_\_</ns2:permissionWaiverID>

       <ns2:permissionWaiverSignatory>\_\_\_\_</ns2:permissionWaiverSignatory>

       <ns2:permissionAcceptDate></ns2:permissionAcceptDate>

       <ns2:permissionStartDate></ns2:permissionStartDate>

       <ns2:permissionEndDate></ns2:permissionEndDate>

       <ns2:operationOverPeople xsi:type="waiverType">True</ns2:operationOverPeople>

   </ns2:permissions>

</ns3:operatingOrganization>

<uniqueMessageIdentifier codeSpace=""/>

</FlightMessage>

## A.2 TPP to Operator: Accept/Reject.

This message is sent from the TPP to the operator once the TPP has decided the status of the flight based upon the rules stated in part 107 OR the ATC has responded to the TPP about authorization. Important information includes the planningStatus data class within the flightPlanNegotiationStatus data class. If the status returns "Rejected" the statusReason will contain information about the rules in violation. Note: with an "Accepted" status, the statusReason may still contain the rules in violation as long as the proper waivers have been obtained. The statusReason will include information about these waivers.

<message xmlns="<http://www.fixm.aero/messaging/4.0>" xmlns:xsi="<http://www.w3.org/2001/XMLSchema-instance>"   
xmlns:ns2="<http://www.fixm.aero/base/4.0>" xmlns:ns3="<http://www.fixm.aero/flight/4.0>"   
xsi:schemaLocation="<http://www.w3.org/2001/XMLSchema-instance> <http://www.fixm.aero/fixm/4.0/Fixm.xsd>">

    <messageDateTime timeReference="UTC"></messageDateTime>

    <ns3:flightPlanNegotiationStatus>

        <ns3:planningStatus status="Accepted" statusReason=""/>

        <!-- OR <ns3:planningStatus status="Rejected" statusReason="part 107, rule 1: airspace class A. part 107 rule \_\_: nighttime operation"/> →

        <!-- OR <ns3:planningStatus stauts="Rejected" statusReason=""/> — this means that no modification will get a waiver. -->

    </ns3:flightPlanNegotiationStatus>

    <flight>

        <!-- Specifically, will contain the flight plan VERSION. No need to reiterate the entire flight plan. This saves space and prevents bloat in order to improve readability. A relational database should have no problems, given the proper message IDs, message collection IDs, and flight plan versions. -->

    </flight>

    <recipient>

        <!-- who gets the message: operator, specified in operating oranization, or the official contact for this flight plan. →

    </recipient>

    <messageOriginator>  <!-- Information about who sent this message: TPP --> </messageOriginator>

    <operatingOrganization>

        <!-- Information about UAS operator ... who is flying the UAS. If the status is Accepted after ATC has granted a waiver, the operatingOrganization field will be updated. An example is as follows:

        -->

        <ns2:permissions>

           <ns2:permissionWaiverID>\_\_\_\_</ns2:permissionWaiverID>

           <ns2:permissionWaiverSignatory>\_\_\_\_</ns2:permissionWaiverSignatory>

           <ns2:permissionAcceptDate></ns2:permissionAcceptDate>

           <ns2:permissionStartDate></ns2:permissionStartDate>

           <ns2:permissionEndDate></ns2:permissionEndDate>

           <ns2:speedWaiver xsi:type="waiverType">True</ns2:speedWaiver>

       </ns2:permissions>

    </operatingOrganization>

    <uniqueMessageIdentifier codeSpace=""/>

</message>

## A.3 TPP to ATC: Notification

This message occur when a proposed flight plan is otherwise authorized, except for operating within 5 miles of an airport. The full flight plan is to be submitted, as the ATC requires the information to render a decision. Notice the planningStatus data class: notification. The important indicator within the flight plan is the airClass attribute (within the region data class) and the distance data class, within the (arrival) aerodrome.

<message xmlns="<http://www.fixm.aero/messaging/4.0>" xmlns:xsi="<http://www.w3.org/2001/XMLSchema-instance>"   
xmlns:ns2="<http://www.fixm.aero/base/4.0>" xmlns:ns3="<http://www.fixm.aero/flight/4.0>"   
xsi:schemaLocation="<http://www.w3.org/2001/XMLSchema-instance> <http://www.fixm.aero/fixm/4.0/Fixm.xsd>">

    <messageDateTime timeReference="UTC"></messageDateTime>

    <!--  And acknowledge status with a notification reason: notifies ATC of proposed flight. This will be sent simultaneously with a message of "accepted" to the operator.

    -->

    <ns3:flightPlanNegotiationStatus>

        <ns3:planningStatus status="Acknowledge" statusReason="Notification"/>

        <!-- OR <ns3:planningStatus status="Rejected" statusReason="part 107, rule 1: airspace class A. part 107 rule \_\_: nighttime operation"> -->

    </ns3:flightPlanNegotiationStatus>

    <flight>

        <!-- Contains information about departure/arrival times/locations and flight radius and location and time: full flight plan. And will contain the flight plan VERSION. Give the proper message IDs, message collection IDs, and flight plan versions.

         -->

    </flight>

    <recipient> <!-- who gets the message: ATC --> </recipient>

    <messageOriginator>  <!-- Information about who sent this message: TPP --> </messageOriginator>

    <operatingOrganization> <!-- Information about UAS operator ... who is flying the UAS -->  </operatingOrganization>

    <uniqueMessageIdentifier codeSpace=""/>

</message>

## A.4 TPP to ATC: Authorization Request

This message is sent from the TPP to ATC if authorization to waive some part 107 rules is required.

<message xmlns="<http://www.fixm.aero/messaging/4.0>" xmlns:xsi="<http://www.w3.org/2001/XMLSchema-instance>"   
xmlns:ns2="<http://www.fixm.aero/base/4.0>" xmlns:ns3="<http://www.fixm.aero/flight/4.0>"   
xsi:schemaLocation="<http://www.w3.org/2001/XMLSchema-instance> <http://www.fixm.aero/fixm/4.0/Fixm.xsd>">

    <messageDateTime timeReference="UTC"></messageDateTime>

    <ns3:flightPlanNegotiationStatus>

        <ns3:planningStatus status="Acknowledge" statusReason="Authorization Request">

        <!-- The following authorizationRequest is a proposed extension to the planning status data class. Add indicators as necessary.  -->

            <ns3:VLOSIndicator requestDateTime="">Boolean</ns3:VLOSIndicator>

            <ns3:overPeopleIndicator requestDateTime="">Boolean, optional</ns3:overPeopleIndicator>

        <!-- ... Add additional indicators (authorization request indicators) as needed ... -->

        </ns3:planningStatus>

    </ns3:flightPlanNegotiationStatus>

    <flight>

        <!-- Contains information about departure/arrival times/locations and flight radius and location and time: full flight plan. And will contain the flight plan version. Give the proper message IDs, message collection IDs, and flight plan versions. This information will be used by ATC to determine authorization status.

         -->

    </flight>

    <recipient> <!-- who gets the message: ATC --> </recipient>

    <messageOriginator>  <!-- Information about who sent this message: TPP --> </messageOriginator>

    <operatingOrganization> <!-- Information about UAS operator ... who is flying the UAS -->  </operatingOrganization>

    <uniqueMessageIdentifier codeSpace=""/>

</message>

## A.5 ATC to TPP: Accept – Authorization Granted

This message is sent from the ATC to the TPP after the ATC has recieved an authorization request. This response includes the waiver ID, the acceptance date, and the start/expiration date for the waiver along with the waiver-type indicator. The full flight plan does not need to be included, as the waiver is granted with no modifications.

<message xmlns="<http://www.fixm.aero/messaging/4.0>" xmlns:xsi="<http://www.w3.org/2001/XMLSchema-instance>"   
xmlns:ns2="<http://www.fixm.aero/base/4.0>" xmlns:ns3="<http://www.fixm.aero/flight/4.0>"   
xsi:schemaLocation="<http://www.w3.org/2001/XMLSchema-instance> <http://www.fixm.aero/fixm/4.0/Fixm.xsd>">

    <messageDateTime timeReference="UTC"></messageDateTime>

    <ns3:flightPlanNegotiationStatus>

        <ns3:planningStatus status="Accept" statusReason="Waiver granted for part 107 rule \_\_\_.">

        </ns3:planningStatus>

    </ns3:flightPlanNegotiationStatus>

    <flight>

        <!-- Full flight plan not returned to TPP, only the flight plan version. Give the proper message IDs, message collection IDs, and flight plan versions.

         -->

    </flight>

    <recipient> <!-- who gets the message: TPP --> </recipient>

    <messageOriginator>  <!-- Information about who sent this message: ATC --> </messageOriginator>

    <operatingOrganization> <!-- Information about UAS operator ... who is flying the UAS. With ATC authorization, the permissions section here can be updated once the TPP notifies the operator in the next step, or the ATC can update the permissions now. The first case is covered in Appendix A.2 -->

           <ns2:permissions>

               <ns2:permissionWaiverID>\_\_\_\_</ns2:permissionWaiverID>

               <ns2:permissionWaiverSignatory>\_\_\_\_</ns2:permissionWaiverSignatory>

               <ns2:permissionAcceptDate></ns2:permissionAcceptDate>

               <ns2:permissionStartDate></ns2:permissionStartDate>

               <ns2:permissionEndDate></ns2:permissionEndDate>

               <ns2:speedWaiver xsi:type="waiverType">True</ns2:speedWaiver>

           </ns2:permissions>

    </operatingOrganization>

    <uniqueMessageIdentifier codeSpace=""/>

</message>

## A.6 ATC to TPP: Reject with Minor Modifications

This message is sent if it is determined that a slight modification to the flight plan will get a waiver or make a waiver unnecessary. Note the words "Modify ..." within the statusReason in the planningStatus data class. In this case, "slight" means lowering maximum altitude or moving the operating location(s) within one (this number can be changed) flight radius. Flight information is not returned; the flight plan version number is given so the operator may reference the correct flight plan proposal. The quantity which needs to be modified is given within the statusReason of the planningStatus (rule number violation), and ATC does not need to determine in which direction to modify the given quantity. That is left to the operator.

<message xmlns="<http://www.fixm.aero/messaging/4.0>" xmlns:xsi="<http://www.w3.org/2001/XMLSchema-instance>"   
xmlns:ns2="<http://www.fixm.aero/base/4.0>" xmlns:ns3="<http://www.fixm.aero/flight/4.0>"   
xsi:schemaLocation="<http://www.w3.org/2001/XMLSchema-instance> <http://www.fixm.aero/fixm/4.0/Fixm.xsd>">

    <messageDateTime timeReference="UTC"></messageDateTime>

    <ns3:flightPlanNegotiationStatus>

        <ns3:planningStatus status="Reject" statusReason="Part 107 rule \_\_ violation. Modify (altitude/location).">

        </ns3:planningStatus>

    </ns3:flightPlanNegotiationStatus>

    <flight>

        <!-- Full flight plan not returned to TPP, only the flight plan version. Give the proper message IDs, message collection IDs, and flight plan versions.

         -->

    </flight>

    <recipient> <!-- who gets the message: ATC --> </recipient>

    <messageOriginator>  <!-- Information about who sent this message: TPP --> </messageOriginator>

    <operatingOrganization> <!-- Information about UAS operator ... who is flying the UAS -->  </operatingOrganization>

    <uniqueMessageIdentifier codeSpace=""/>

</message>

## A.7 ATC to TPP: Reject

The ATC determines that the flight cannot be modified in order to be granted a waiver. E.g., proposal to take real estate photographs within a no-fly zone.

<message xmlns="<http://www.fixm.aero/messaging/4.0>" xmlns:xsi="<http://www.w3.org/2001/XMLSchema-instance>"   
xmlns:ns2="<http://www.fixm.aero/base/4.0>" xmlns:ns3="<http://www.fixm.aero/flight/4.0>"   
xsi:schemaLocation="<http://www.w3.org/2001/XMLSchema-instance> <http://www.fixm.aero/fixm/4.0/Fixm.xsd>">

    <messageDateTime timeReference="UTC"></messageDateTime>

    <ns3:flightPlanNegotiationStatus>

        <ns3:planningStatus status="Reject" statusReason="">

        </ns3:planningStatus>

    </ns3:flightPlanNegotiationStatus>

    <flight>

        <!-- Full flight plan not returned to TPP, only the flight plan version. Give the proper message IDs, message collection IDs, and flight plan versions.

         -->

    </flight>

    <recipient> <!-- who gets the message: ATC --> </recipient>

    <messageOriginator>  <!-- Information about who sent this message: TPP --> </messageOriginator>

    <operatingOrganization> <!-- Information about UAS operator ... who is flying the UAS -->  </operatingOrganization>

    <uniqueMessageIdentifier codeSpace=""/>

</message>

# Appendix B Message Collections

A message collection is an object which contains a number of FIXM messages. The following messages will be grouped based upon UAS flight plan – each flight plan will be a message collection consisting of the proposed flight plan(s) and any response from the third party providers or air traffic control (ATC). If ATC conditionally rejects a flight plan, the operator must file a modified or new flight plan, to be approved by the ATC again. An example abbreviated message collection follows.

<messageCollection xmlns="<http://www.fixm.aero/messaging/4.0>" xmlns:xsi="<http://www.w3.org/2001/XMLSchema-instance>"   
xmlns:ns2="<http://www.fixm.aero/base/4.0>" xmlns:ns3="<http://www.fixm.aero/flight/4.0>"   
xsi:schemaLocation="<http://www.w3.org/2001/XMLSchema-instance> <http://www.fixm.aero/fixm/4.0/Fixm.xsd>">

<uniqueMessageIdentifier codeSpace=""/>

<!-- Message 1: flight plan proposal, see Appendix A.1 -->

<message>

    <messageDateTime timeReference="UTC"></messageDateTime>

    <ns3:flightPlanNegotiationStatus> ... </ns3:flightPlanNegotiationStatus>

    <flight>

        <!-- Contains information about departure/arrival times/locations and flight radius and location and time. -->

    </flight>

    <recipient> <!-- who gets the message --> </recipient>

    <messageOriginator>  <!-- Information about who sent this message --> </messageOriginator>

    <operatingOrganization> <!-- Information about UAS operator ... who is flying the UAS -->  </operatingOrganization>

    <uniqueMessageIdentifier codeSpace=""/>

</message>

<!-- Message 2: notification of flight to FAA/ATC. See Appendix A.3 -->

<message>

    <messageDateTime timeReference="UTC"></messageDateTime>

    <ns3:flightPlanNegotiationStatus> ... </ns3:flightPlanNegotiationStatus>

    <flight>

        <!-- Contains information about departure/arrival times/locations and flight radius and location and time. -->

    </flight>

    <recipient> <!-- who gets the message --> </recipient>

    <messageOriginator>  <!-- Information about who sent this message --> </messageOriginator>

    <operatingOrganization> <!-- Information about UAS operator ... who is flying the UAS -->  </operatingOrganization>

    <uniqueMessageIdentifier codeSpace=""/>

</message>

<!-- Message 3: TPP notification of acceptance of flight plan to operator. See Appendix A.2 -->

<message>

    <messageDateTime timeReference="UTC"></messageDateTime>

    <ns3:flightPlanNegotiationStatus> ... </ns3:flightPlanNegotiationStatus>

    <flight>

        <!-- Contains information about departure/arrival times/locations and flight radius and location and time. -->

    </flight>

    <recipient> <!-- who gets the message --> </recipient>

    <messageOriginator>  <!-- Information about who sent this message --> </messageOriginator>

    <operatingOrganization> <!-- Information about UAS operator ... who is flying the UAS -->  </operatingOrganization>

    <uniqueMessageIdentifier codeSpace=""/>

</message>

<!-- End of message collection. Flight plan approved. -->

</messageCollection>