# D3.3

# ALFA system data model Database

| | |
|---|---|
| **Project number:** | 700002 |
| **Project acronym:** | ALFA |
| **Project title:** | Advanced Low Flying Aircrafts Detection and Tracking |
| **Start date of the project:** | 1st January, 2017 |
| **Duration:** | 36 months |
| **Programme:** | H2020-BES-2015 |

| | |
|---|---|
| **Deliverable type:** | Other |
| **Deliverable reference number:** | BES-700002 / D3.3/ V1.0 |
| **Work package contributing to the deliverable:** | WP 3 |
| **Due date:** | MAR 2019 – M27 |
| **Actual submission date:** | 29th March 2019 |

| | |
|---|---|
| **Responsible organisation:** | ENG |
| **Editor:** | Giuseppe Francaviglia |
| **Dissemination level:** | PU |
| **Revision:** | V 1.0 |

| | |
|---|---|
| **Abstract:** | This document describes the implementation of the ALFA data model and the JSON data structures |
| **Keywords:** | ALFA system data model, JSON data structures |

The ALFA project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 700002.

## Editor

Giuseppe Francaviglia (ENG)

## Contributors (ordered according to beneficiary numbers)

FERNANDES, Luis; CHAVES, Paulo (INOV)

PRÉMEL-CABIC, Gilles (TNL)

FRANCAVIGLIA, Giuseppe; PASQUAZI, Jacopo (ENG)

POL, Johan van de (TNO)

RUIZ LOZANO, Francisco Javier; MARTINEZ-SALIO, José-Ramón (Atos)

KRÜCKEMEIER, Markus (TUBS)

## Disclaimer

# Executive Summary

This report describes the activities performed within the framework of the ALFA project Work Package 3 (WP3) "Components and technologies", Task 3.6 "System Data Model".

The ALFA Work Package 3 objective is to achieve the necessary developments for detection capabilities according to findings of Work Package 1; the applicable components and technologies have been developed according to sensor improvements described in Task 1.5. Also, all the components and the technologies developed in the context of the ALFA framework are compliant with the architectural aspects, as described in Task 2.1 (ALFA architecture for border surveillance) and with the data representation aspects, as described in Task 2.3 (Data model architecture).

The objective of Task 3.6 is the implementation of the JavaScript Object Notation (JSON) data structures required to describe the information exchanged by the ALFA components.

Starting from the conceptual and logical schema defined in Task 2.3, the data structures to represent the following information have been generated:

- The data generated by the ALFA sensors, which detect the air flying objects and need to send this information to the fusion and classification capability of the ALFA system;
- The data generated by the data processing capability of the ALFA system, i.e. the fusion and classification, the threat analysis and the behaviour analysis;
- The data generated by the prediction capability of the ALFA system, which predicts the potential landing sites and drop off, and the probable routes of a target.

Furthermore, during the course of the task, the data structures have been modified according to new system and user requirements to be able to satisfy changes in the data generated by the ALFA components still in development phase.

Finally, are presented the data flow diagrams in order to clarify how the ALFA system components participate in the management of the data.

| | |
|---|---|
| Input from WP(s): | WP1, WP2 |
| Output to WP(s): | WP4, WP5 |
| Deliverables: | D3.3 |
| Duration: | April 1, 2018 to March 31, 2019 |
| WP leader: | TNL |
| Participants: | INOV, TNL, ENG, TNO, Atos, TUBS |

# Contents

# List of Figures



# List of Tables

# Chapter 1 Introduction

## 1.1 Purpose

This report describes the activities performed within the framework of the ALFA project Work Package 3 (WP3) "Components and technologies", Task 3.6 "System Data Model".

The main purpose is to deliver the ALFA common system data model, enabling the representation of the information exchanged in the ALFA system by means of JSON data structures. By means of these data schemas, all the ALFA components are able to receive the right data, to produce their output, and to communicate it in the agreed format. Data schemas are developed according to the logical model defined in Task 2.3 and described in D2.2; actually, the ALFA data model has changed a bit since then in order to meet the last developments of the ALFA components and to run the first integration of the ALFA system. All the updates have been implemented in the existing logical model and have been presented in this document.

According with the data serialization frameworks analysed during the architecture design, the data structures are implemented in JSON.

## 1.2 Document overview

Chapter 2 reports the main updates made to the ALFA data model during the development phase and the first integration phase of the project.

Chapter 3 provides a view on the representation of the Track entity. The data structure to represent a track and all its attributes is presented.

Chapter 4 introduces the data structure used to represent the information that the ALFA sensors are able to detect and send to the ALFA system.

Chapter 5 reports about the data structures used to represent the heartbeat messages of the ALFA sensors.

In Chapter 6 the data flow diagrams are introduced.

Finally, the summary and the conclusion of this document are presented in Chapter 7.

# Chapter 2 ALFA data model updates

This chapter introduces the updates made to the ALFA data model since its first version was released with deliverable D2.2.

The conceptual and the logical schemas of the ALFA data model presented in D2.2 were designed during the requirement analysis phase of the project, labelling so the starting point for the future developments of the model itself and for the implementation of the data structures. That version of the data model was designed in order to define the first generalization of all the entities included in the ALFA domain, with the openness and the awareness that it would evolve during the development of the detection capability, the data processing and the prediction capability of ALFA.

Some updates to the data model were also made as the integration phase of the project started, in order to improve the integration and the performance of the whole system while exchanging and managing the information.

Updates to the data model consist of the representation of new information, changes in some entity attributes and relationships.

Following the Unified Modelling Language (UML) diagrams representing the updated logical schemas are shown.

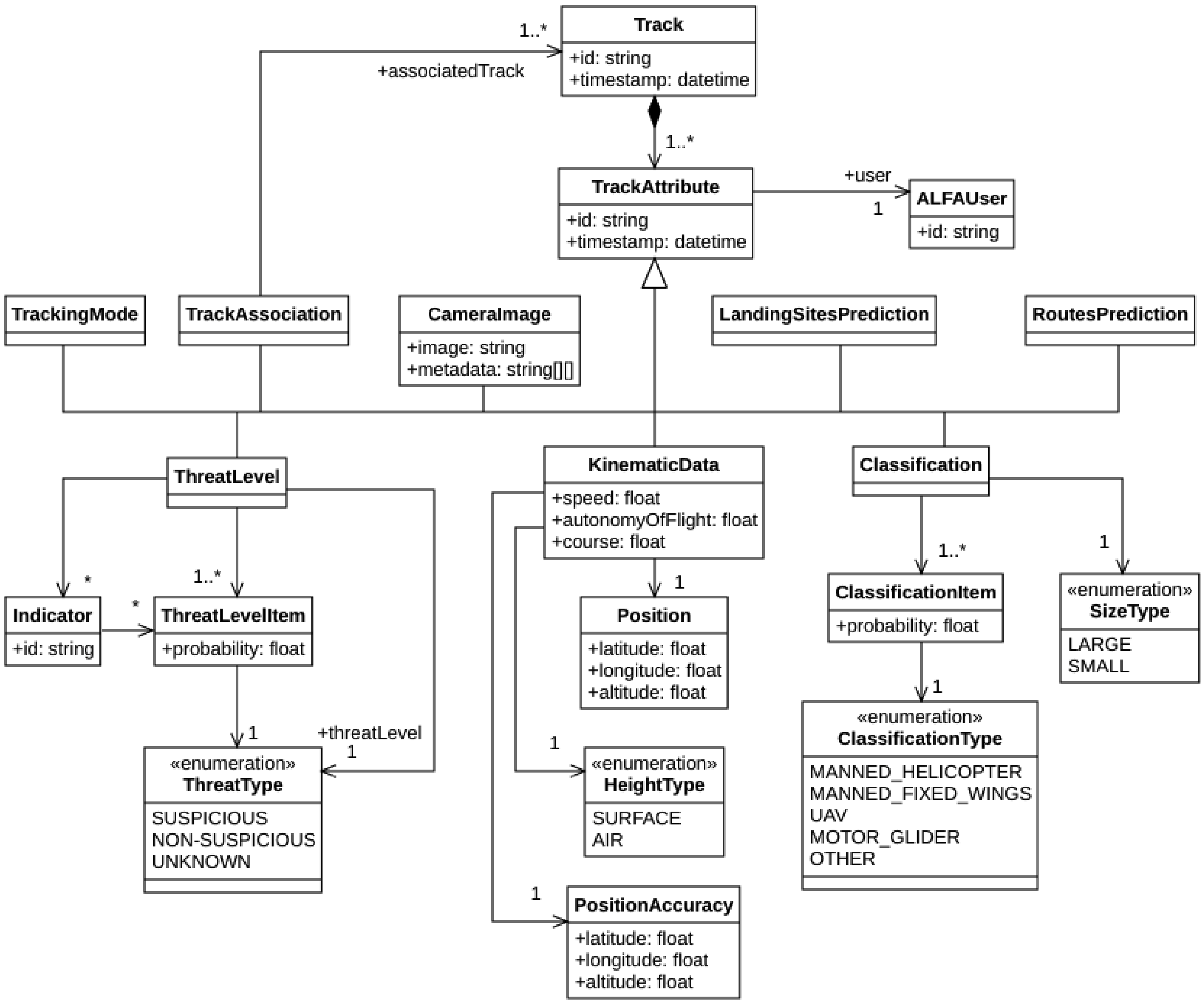


Figure 1: Track logical model

Figure 1 shows the updated logical model of the Track; major updates involve:

- The RoutesPrediction entity has been added as an attribute of the Track;
- The TrackingMode entity has been added as an attribute of the Track;
- The TargetVideo has been removed;
- A direct relationship between the ThreatLevel and the ThreatType has been added;
- The ThreatLevel entity includes now a list of Indicator;
- The KinematicData refers now to only one Position;
- The PositionAccuracy entity has been added.

Figure 2 shows the logical schema of the RoutesPrediction, added to the data model. This entity represents the prediction of the probable route the track could follow and it is calculated by the prediction capability of ALFA; it is an attribute of the track and it is a specialization of the TrackAttribute entity.

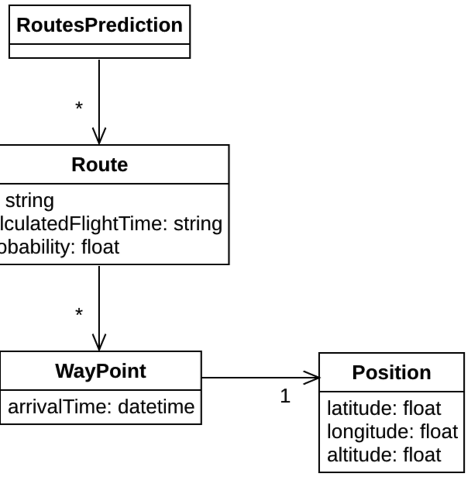


Figure 2: Routes prediction logical model

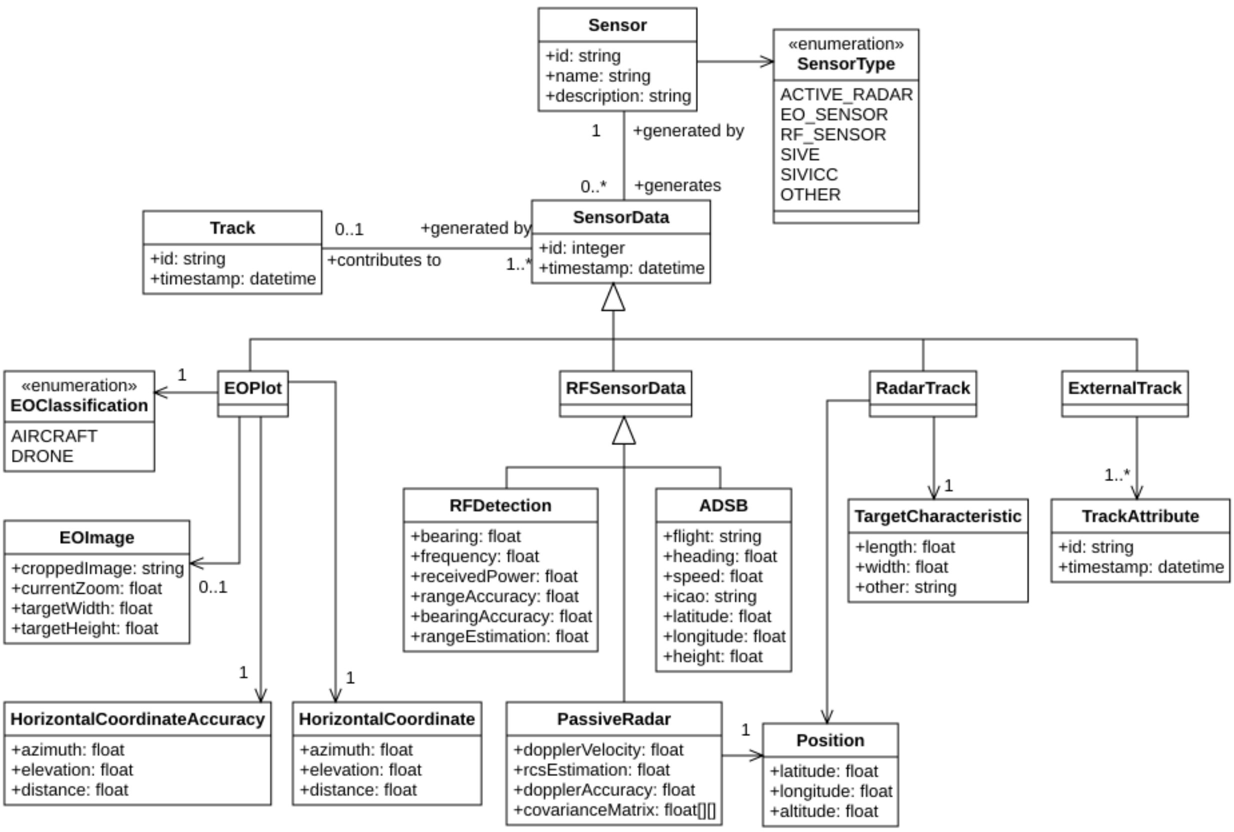


Figure 3: Sensor logical model

Figure 3 shows the updated logical model of the Sensor; major updates involve:

- The relationship between the Sensor and the Position has been removed; this relationship has been moved in the Sensor Configuration logical model (see Figure 4);
- The EOClassification enumeration has been introduced;
- The HorizontalCoordinateAccuracy entity has been added to the Electro Optic (EO) plot;

- The RadioSystemClassification enumeration has been removed;
- The Automatic Dependent Surveillance - Broadcast (ADSB) entity has been added.

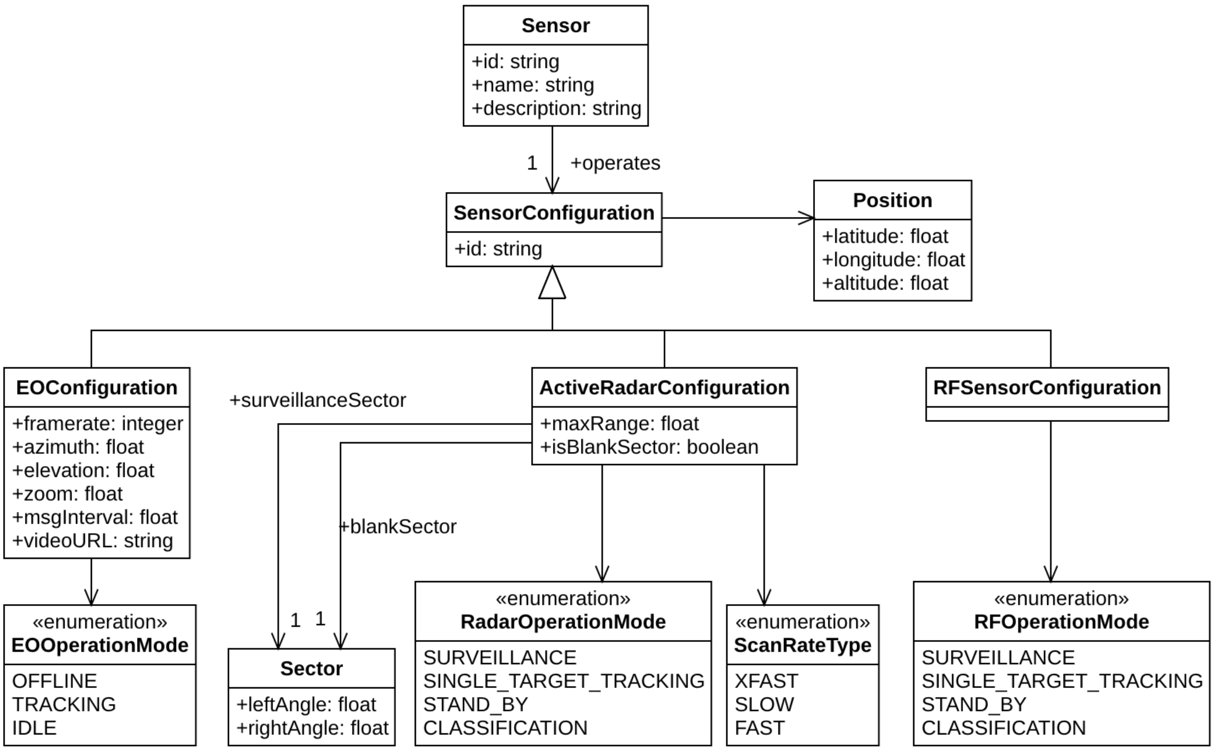


Figure 4: Sensor configuration logical model

Figure 4 shows the updated logical model of the Sensor Configuration; major updates involve:

- The Position entity has been introduced as an attribute of the SensorConfiguration;
- The relationship between the RFSensorConfiguration and the Position entity has been removed; now each SensorConfiguration specifies its position directly (see the previous point);
- The FrequencyNumber enumeration has been removed from the previous model.

# Chapter 3 Track data structure

This chapter reports the data structures used to represent all the information related to a track. As discussed in the ALFA data model [ALFA22], the track is the main entity of the ALFA domain, and it describes all the information of the flying objects detected, tracked and classified by the ALFA system. The following table reports the data structure used to represent a track, and in the following sub sections all the attributes of the track will be further detailed.

| | |
|---|---|
| {<br>"id": "test-station.track1",<br>"timestamp": "2019-02-11T14:00:27.476Z",<br>"trackAttribute": [<br>    { kinematic data },<br>    { classification },<br>    { threat level },<br>    { camera image },<br>    { tracking mode },<br>    { track association },<br>    { landing sites prediction },<br>    { routes prediction }<br>  ]<br>} | **id**, the identifier of this object<br>**timestamp**, the timestamp of the track, represented in Zulu time format<br>**trackAttribute**, a list including the data structures which represent the attributes of the track |

Table 1: Track data structure (JSON representation)

According to the ALFA architecture [ALFA21], in order to improve the performance and the responsiveness of the integrated system, only new tracks and the updates of the existing ones will be propagated via RabbitMQ: that means every time an attribute of a track changes its value, it will propagated as a separate JSON message via RabbitMQ.

The JSON data structures which represent the tracks' attributes have been defined; the following sub sections report the data structure which represent the kinematic data, the classification, the threat level, the camera image, the tracking mode, the track association, the landing sites prediction and the routes prediction.

## 3.1 Kinematic data

This sub section reports the data structure used to represent the kinematic data of a track.

| | |
|---|---|
| {<br>"id": null,<br>"timestamp": "2019-02-11T14:00:27.476Z",<br>"user": {<br>        "id": "control-station.data-generator"<br>        },<br>"speed": 107.85951,<br>"course": 63.07191,<br>"autonomyOfFlight": 0.0, | **id**, the identifier of this object<br>**timestamp**, the timestamp of this object, represented in Zulu time format<br>**user**, includes the identifier of the ALFA component that generated this object<br>**speed**, float, the speed of the flying object in meters per second |

| | |
|---|---|
| "heightType": "AIR",<br>"position": {<br>"latitude": 37.507652,<br>"longitude": -8.460271,<br>"altitude": 999.0<br>},<br>"positionAccuracy": {<br>"latitude": 0.1,<br>"longitude": 0.2,<br>"altitude": 0.4<br>}<br>} | **course**, float, the course of the flying object in degrees from north<br>**autonomyOfFlight**, float, the flight autonomy of the flying object in seconds<br>**heightType**, enumeration, possible values are AIR and SURFACE<br>**position**, the position of the flying object, latitude and longitude in degrees and altitude in meters<br>**positionAccuracy**, the accuracy of the position in meters for all directions |

Table 2: Kinematic data (JSON representation)

## 3.2 Classification

This sub section reports the data structure used to represent the information regarding the classification of a track.

| | |
|---|---|
| {<br>"id": null,<br>"timestamp": "2019-02-11T14:00:27.476Z",<br>"user": {<br>"id": "control-station.data-generator"<br>},<br>"classificationItems": [{<br>"classificationType": "UAV",<br>"probability": 93.0<br>},<br>{<br>"classificationType": "OTHER",<br>"probability": 7.0<br>}],<br>"sizeType": "SMALL"<br>} | **id**, the identifier of this object<br>**timestamp**, the timestamp of this object, represented in Zulu time format<br>**user**, includes the identifier of the user/ALFA component who generated this object<br>**classificationItems**, this list includes the classifications of the track and the related probability<br>**classificationType**: enumeration, possible values "MANNED_HELICOPTER", "UAV", "MANNED_FIXED_WINGS", "MOTOR_GLIDER", "OTHER"<br>**sizeType**: enumeration, possible values are "LARGE", "SMALL" |

Table 3: Classification data structure (JSON representation)

## 3.3 Threat Level

This sub section reports the data structure used to represent the information regarding the suspiciousness/threat level of a track.

| | |
|---|---|
| {<br>"id": null,<br>"timestamp": "2019-02-11T14:00:27.476Z",<br>"user": {<br>"id": "control-station.data-generator"<br>},<br>"threatLevel": "SUSPICIOUS",<br>"threatLevelItems": [{<br>"probability": 0.63,<br>"threatType": "SUSPICIOUS"<br>},<br>{<br>"probability": 0.27,<br>"threatType": "NON_SUSPICIOUS"<br>},<br>{<br>"probability": 0.1,<br>"threatType": "UNKNOWN"<br>}]<br>} | **id**, the identifier of this object<br>**timestamp**, the timestamp of this object, represented in Zulu time format<br>**user**, includes the identifier of the user/ALFA component who generated this object<br>**threatLevel**, specifies the current threat level of the track; see **threatType** for possible values<br>**threatLevelItems**, this list specifies the probability of each suspiciousness level for the track<br>**probability**, float, the probability as a value between 0 and 1<br>**threatType**: enumeration, possible values are SUSPICIOUS, NON_SUSPICIOUS and UNKNOWN |

Table 4: Threat Level data structure (JSON representation)

## 3.4 Camera image

This sub section reports the data structure used to represent the image captured by the EO sensor during the tracking of a track.

| | |
|---|---|
| {<br>"id": "test-station.a12ba33f-c882-3a72-86e2-8509c3e39d5a",<br>"timestamp": "2019-03-07T09:48:17.971Z",<br>"user": {<br>"id": "test-station.fc"<br>},<br>"image": "a string in Base64 encoding",<br>"metadata": {<br>"zoom": "2.5"<br>}<br>} | **id**, the identifier of this object<br>**timestamp**, the timestamp of this object, represented in Zulu time format<br>**user**, includes the identifier of the ALFA component that generated this object<br>**image**, a string representing the image encoded in base64<br>**metadata**, includes the metadata of the image as name-value pairs; e.g. the zoom factor |

Table 5: Camera Image data structure (JSON representation)

## 3.5 Tracking mode

This sub section reports the data structure used to specify the ALFA sensors that are working in tracking mode. If the EO sensor is working in tracking mode, also the url of the camera live feed is specified (if present).

| | |
|---|---|
| {<br>"id": "test-station.a12ba33f-c882-3a72- 86e2-8509c3e39d5a",<br>"timestamp": "2019-03-07T09:48:17.971Z",<br>"user": {"id": "test-station.fc"},<br>"mode": "CAMERA",<br>"sensors": [<br>"test-station.radar1",<br>"test-station.camera"<br>],<br>"videoUrl": " the url "<br>} | **id**, the identifier of this object<br>**timestamp**, the timestamp of this object, represented in Zulu time format<br>**user**, includes the identifier of the ALFA component that generated this object<br>**mode,** specifies what sensor is working in track mode, possible values are RADAR, CAMERA, BOTH and NONE<br>**sensors**, the list including the sensor (the IDs) which are allocated to track the target<br>**videoUrl**, represents the url of the camera live feed if present, otherwise it is null or missing |

Table 6: Tracking mode data structure (JSON representation)

## 3.6 Track Association

This sub section reports the data structure used to represent the association between different tracks.

| | |
|---|---|
| {<br>"id": null,<br>"timestamp": "2019-02-11T14:00:27.476Z",<br>"user": {<br>"id": "control-station.data-generator"<br>},<br>"associatedTrack": [<br>{<br>"id": "test-station.track012"<br>},<br>{<br>"id": "test-station.track234"<br>}]<br>} | **id**, the identifier of this object<br>**timestamp**, the timestamp of this object, represented in Zulu time format<br>**user**, includes the identifier of the ALFA component that generated this object<br>**associatedTrack**, it is a list including the id of the tracks to be associated |

Table 7: Track association data structure (JSON representation)

## 3.7 Landing sites prediction

This sub section reports the data structure used to represent the prediction of the landing sites/drop off.


```
{
"id": "3902832",
"timestamp": "2019-02-11T14:00:27.476Z",
"user": {
        "id": "control-station.data-generator"
        },
"locations": [{
        "type": "it.eng.alfa.dm.landingsite.POI",
        "id": 6464126,
        "probability": 30.0,
        "arrivalTime": "2019-02-11T14:00:27.476Z",
        "description": "POI Drop-off",
        "linearDistance": 53929.758,
        "trajectory": null,
        "locationType": "DROPOFF",
        "position": {
                "latitude": 37.722904,
                "longitude": -7.923643,
                "altitude": 999.0
                }
        },
        {
        "type": "it.eng.alfa.dm.landingsite.Area",
        "id": 2712367,
        "probability": 70.0,
        "arrivalTime": "2019-02-11T14:00:27.476Z",
        "description": "Landing Area",
        "linearDistance": 107859.516,
        "trajectory": null,
        "locationType": "LANDING",
        "positions": [{
                "latitude": 37.91915,
                "longitude": -7.395054,
                "altitude": 0.0
                },
                {
                "latitude": 37.95915,
                "longitude": -7.395054,
                "altitude": 0.0
                },
                {
                "latitude": 37.95915,
                "longitude": -7.355054,
                "altitude": 0.0
```


**id**, the identifier of this object

**timestamp**, the timestamp of this object, represented in Zulu time format

**user**, includes the identifier of the ALFA component that generated this object

**locations**, the list including the predictions; a prediction will be represented as an area; each location could represent a drop off (**locationType** set to DROPOFF) or a landing area (**locationType** set to LANDING).

Each location is also described by:

**probability**, the probability of this prediction as a percentage

**arrivalTime**, the arrival time in Zulu format

**description**, the description of the location

**linearDistance**, the linear distance between the flying object and the area, calculated in meters

**trajectory**, the trajectory of the target, if available

Moreover, the **type** attribute defines if a location is a **POI** or an **Area**: if the location is represented by a POI, then it includes its position defined as a single point on the map; else if the location is defined by an Area, then it includes the position of the points that enclose the Area.

In any case, the **position** is defined by the latitude and longitude in degrees and by the altitude in meters

| | |
|---|---|
| },<br>{<br>"latitude": 37.91915,<br>"longitude": -7.355054,<br>"altitude": 0.0<br>},<br>{<br>"latitude": 37.91915,<br>"longitude": -7.395054,<br>"altitude": 0.0<br>}]<br>}]<br>} | |

Table 8: Landing sites prediction data structure (JSON representation)

## 3.8 Routes prediction

This sub section reports the data structure used to represent the route predictions.

| | |
|---|---|
| {<br>"id": "routePrediction1",<br>"timestamp": "2019-02-11T14:00:27.476Z",<br>"user": {<br>"id": "control-station.data-generator"<br>},<br>"routes": [{<br>"calculatedFlightTime": "300",<br>"id": "route1",<br>"probability": 90.0,<br>"wayPoints": [{<br>"arrivalTime": "2019-02-11T14:00:27.476Z",<br>"position": {<br>"latitude": 45.0,<br>"longitude": 45.0,<br>"altitude": 300.0<br>}<br>},<br>.....,<br>{<br>"arrivalTime": "2019-02-11T14:00:27.476Z",<br>"position": {<br>"latitude": 45.1,<br>"longitude": 45.1,<br>"altitude": 300.0<br>}<br>}]<br>}]<br>} | **id**, the identifier of this object<br><br>**timestamp**, the timestamp of this object, represented in Zulu time format<br><br>**user**, includes the identifier of the ALFA component that generated this object<br><br>**routes**, a list which includes the predicted routes of the track. Each route includes:<br>• **id**, the identifier of the route<br>• **probability**, the likelihood of the route<br>• **calculatedFlightTime**, the flight time of the route measured in seconds<br>• **wayPoints**, the list of waypoints, each described by the position and the arrival time in Zulu format; latitude and longitude in degrees and altitude in meters |

Table 9: Routes prediction data structure (JSON representation)

# Chapter 4 Sensor data structure

This chapter reports the data structures used to represent all the information detected by the ALFA sensors. As defined in the ALFA data model [ALFA22], three type of sensor data have been modelled: the plots provided by the EO sensor, the plots provided by the Radio Frequency (RF) sensor, and the plots provided by the active radar.

The following sub sections report the data structures used to model the data generated by the sensors and transfer this data to the ALFA module in charge of the fusion and classification capability.

## 4.1 Active radar data structure

This sub section reports the data structure used to represent the data generated by the ALFA active radar.

| | |
|---|---|
| { <br>"id": 8775842, <br>"timestamp": "2019-02-11T14:00:27.476Z", <br>"track": null, <br>"targetCharacteristic": { <br>"length": 10.4, <br>"width": 20.0, <br>"other": "some string" <br>}, <br>"position": { <br>"latitude": 10.4, <br>"longitude": 20.0, <br>"altitude": 34.2 <br>} <br>} | **id**: integer, the identifier of this data <br><br>**timestamp**: the timestamp of this data, represented in Zulu time format <br><br>**track**, represents the identifier of the track, if available <br><br>**targetCharacteristic**, the characteristic of the tracked flying object <br><br>**position**, the position of the flying object, latitude and longitude in degrees and altitude in meters |

Table 10: Sensor data, radar data structure (JSON representation)

## 4.2 Passive radar data structure

This sub section reports the data structure used to represent the data generated by the ALFA RF sensor. This sensor generates RF detections, ADSB data and passive radar measurements.

| | |
|---|---|
| { <br><br>"type": "RF_SENSOR", <br><br>"name": "RFSensor", | **type**, the type of sensor, possible values are RF_SENSOR, ACTIVE_RADAR, EO_SENSOR, SIVE, SIVICC, OTHER <br><br>**name**, string, the name of the sensor |

| | |
|---|---|
| "id": "ALFASENSORSTATION1.rfsensor1",<br>"sensordata": [<br>{<br>"type": "ADSB",<br>"flight": "1234",<br>"timestamp": "2019-02-21T06:40:50.250Z",<br>"longitude": 37.1022267733119,<br>"height": 518.8102893890675,<br>"icao": "4400b7",<br>"latitude": -7.473782421221865,<br>"speed": 150,<br>"heading": 314.7128269100888<br>},<br>{<br>"type": "PassiveRadar",<br>"dopplerAccuracy": 9,<br>"timestamp": "2019-02-21T06:40:50.250Z",<br>"dopplerVelocity": 181.5541918387126,<br>"rcsEstimation": 11.724080586458413,<br>"covarianceMatrix": [<br>[ 0.0005, 0.0001, 0.0001 ],<br>[ 0.0001, 0.0005, 0.002 ],<br>[ 0.0001, 0.002, 100 ]<br>],<br>"position": {<br>"latitude": 39.48087669212153,<br>"altitude": 381.04510870473905,<br>"longitude": -7.9658990487282075<br>},<br>},<br>{<br>"type": "RFDetection", | **id**: string, the identifier of this data<br>**sensordata**, a list of sensor data, in this case the data generated by the RF sensor<br><br>The RF sensor generates 3 types of dataset, and this information is represented by the **type** attribute, the possible values are ADSB, PassiveRadar and RFDetection.<br><br>**<u>ADSB data</u>**<br>**flight**, string, the code of the flying object<br>**timestamp**, the timestamp of this object, represented in Zulu time format<br>**longitude**, float, the longitude of the flying object in degrees<br>**height**, float, the height from the ground of the flying object over the<br>**icao**, string, unique aircraft address expressed in hexadecimal form<br>**latitude**, float, the latitude of the flying object in degrees<br>**speed**, float, the current speed of the flying object in meters per second<br>**heading**, float, heading of the flying aircraft with respect to the north<br><br>**<u>PassiveRadar measurements</u>**<br>**dopplerAccuracy**, float, the doppler accuracy<br>**timestamp**, the timestamp of this object, represented in Zulu time format<br>**dopplerVelocity**, float, doppler velocity in meters per second<br>**rcsEstimation**, float, the radar cross section estimation in $m^2$<br>**covarianceMatrix**, float, covariance of the position<br>**position**, the position of the flying object, latitude and longitude in degrees and altitude in |

| | |
|---|---|
| "bearing": 146.00681701561018,<br>"timestamp": "2019-02-21T06:39:09.149Z",<br>"bearingAccuracy": 0,<br>"frequency": 2400003452.9639215,<br>"receivedPower": -77.54307407709099,<br>"rangeAccuracy": 1500,<br>"rangeEstimation": 5538.763624250492<br>}<br>]<br>} | meters<br><br>**<u>RF detections</u>**<br>**bearing**, float, bearing of sensors in degrees respect to the Geographical North<br>**timestamp**, the timestamp of this object, represented in Zulu time format<br>**bearingAccuracy**, float, bearing accuracy in degrees<br>**frequency**, float, Signal frequency in Hz<br>**receivedPower**, float, measure of received power in dB<br>**rangeAccuracy**, float, range accuracy in meters<br>**rangeEstimation**, float, range estimation in meters |

Table 11: Sensor data, RF sensor data structure (JSON representation)

## 4.3 EO sensor data structure

This section reports the data structure used to represent the data provided by the ALFA EO.

| | |
|---|---|
| {<br>"id": 8775842,<br>"timestamp": "2019-02-11T14:00:27.476Z",<br>"track": null,<br>"eoClassification": "AIRCRAFT",<br>"horizontalCoordinate": {<br>"azimuth": 10.4,<br>"elevation": 20.0,<br>"distance": 30.23<br>},<br>"horizontalCoordinateAccuracy": {<br>"azimuth": 1.4,<br>"elevation": 2.0,<br>"distance": 3.3<br>},<br>"eoImage": {<br>"croppedImage": "a string ",<br>"currentZoom": 2.5,<br>"targetWidth": 1.3, | **id**, the identifier of this object<br>**timestamp**, the timestamp of this object, represented in Zulu time format<br>**track**, represents the identifier of the track, if available<br>**eoClassification**, enumeration, possible values are AIRCRAFT, DRONE<br>**horizontalCoordinate**, represents the horizontal coordinate measurement in degrees (azimuth and elevation) and meters (distance)<br>**horizontalCoordinateAccuracy**, represents the standard deviation of the horizontal coordinate measurement<br>**eoImage**, represents the image of the target and some measurements:<br>• **croppedImage**: a string representing the |

| | |
|---|---|
| "targetHeight": 4.5<br>}<br>} | image encoded in base64<br>• **currentZoom**, the current zoom level<br>• **targetWidth**, the approximate horizontal size of the target in meters<br>• **targetHeight**, the approximate vertical size of the target in meters |

Table 12: Sensor data, EO sensor data structure (JSON representation)

# Chapter 5 Heartbeat data structure

This chapter reports the data structures used to represent the heartbeat messages of the ALFA sensors. The heartbeat is used by the ALFA sensors, and in general by all the ALFA components, to register with the framework and to publish their configuration. For example, the ALFA sensors send their heartbeat each second, so the heartbeat of the sensors will be used to know that the sensor are still alive, and to update the map on the ALFA Geographic Information system (GIS), showing the sensors in the right position and their configuration.

The following table shows the data structure to be used by the ALFA components to define their heartbeat message; the "configuration" part will contain the specific configuration parameters of each component.

<table>
<tr>
<td><pre>{
  "version": "1.0",
  "id": "test-station.squire1",
  "type": "sensor",
  "subtype": "active-radar",
  "data": "test-station.squire1.data",
  "inbox": "test-station.squire1.inbox",

  "configuration": {

        }
}</pre></td>
<td>version, the string representing the message version<br><br>id, the string representing the identifier of the component<br><br>type, a string representing the type of the component (sensor, control, algorithm, ...)<br><br>subtype, a string representing the detailed type info of the component<br><br>data, a string with the component data RabbitMQ topic name<br><br>inbox, a string with the component inbox RabbitMQ topic name<br><br>configuration, it include the configuration of the component</td>
</tr>
</table>

Table 13: Heartbeat, general data structure (JSON representation)

The following sub sections report the heartbeat message of each ALFA sensor, and the configuration part of the message is detailed.

## 5.1 Active radar heartbeat

This sub section reports the data structure used to represent the heartbeat message of the ALFA active radar.

<table>
<tr>
<td><pre>{
  "version": "1.0",
  "id": "test-station.squire1",
  "type": "sensor",
  "subtype": "active-radar",</pre></td>
<td>The first part of this data structure has already been described in Table 13.</td>
</tr>
</table>

| | |
|---|---|
| "data": "test-station.squire1.data",<br>"inbox": "test-station.squire1.inbox",<br>"configuration": {<br>"id": "test-station.squire1",<br>"maxRange": 3000.0,<br>"isBlankSector": true,<br>"surveillanceSector": {<br>"leftAngle": 0.5235,<br>"rightAngle": 2.6180<br>},<br>"blankSector": {<br>"leftAngle": 0.5235,<br>"rightAngle": 2.6180<br>},<br>"radarOperationMode": "SURVEILLANCE",<br>"scanRateType": "FAST",<br>"position": {<br>"latitude": 37.177411,<br>"altitude": 10,<br>"longitude": -7.467875<br>}<br>}<br>} | The configuration part consists of:<br>**id**, string<br>**maxRange**, float, maximum range of the sensor in meters<br>**isBlankSector**, boolean<br>**blankSector**, the blank sector, defined by its left and its right angles in radians<br>**radarOperationMode**, enumeration, possible values are SURVEILLANCE, SINGLE_TARGET_TRACKING, STAND_BY, CLASSIFICATION<br>**scanRateType**, enumeration, possible values are XFAST, SLOW, FAST<br>**position**, the position of this sensor, latitude and longitude in degrees and altitude in meters<br>**surveillanceSector**, the surveillance sector, defined by its left and its right angles in radians |

Table 14: Radar, heartbeat data structure (JSON representation)

## 5.2 Passive radar heartbeat

This sub section reports the data structure used to represent the heartbeat message of the ALFA RF sensor.

| | |
|---|---|
| {<br>"version": "1.0",<br>"id": "ALFASENSORSTATION1.rfsensor1",<br>"type": "sensor",<br>"subtype": "passive-rf",<br>"data": "ALFASENSORSTATION1.passive_rf",<br>"inbox": null,<br>"configuration": {<br>"id": null,<br>"position": {<br>"latitude": 37.177411,<br>"altitude": 10,<br>"longitude": -7.467875<br>}<br>}<br>} | The first part of this data structure has already been described in Table 13.<br><br>The configuration part consists of:<br>**id**: string<br>**position**, the position of this sensor, latitude and longitude in degrees and altitude in meters |

Table 15: RF sensor, heartbeat data structure (JSON representation)

## 5.3 EO sensor heartbeat

This sub section reports the data structure used to represent the heartbeat message of the ALFA EO sensor.

<table>
<tr>
<td>
{<br>"version": "0.1",<br>"id": "test-station.eosensor",<br>"type": "sensor",<br>"subtype": "eo",<br>"inbox": "test-station.eosensor.config",<br>"data": "test-station.eosensor.data",<br>"configuration": {<br>"id": "test-station.eosensor ",<br>"name": "Camera",<br>"framerate": 5,<br>"azimuth": 20.0,<br>"elevation": 30.23,<br>"zoom": 30.23,<br>"msgInterval": 30.23,<br>"videoURL": "https://www.google.com",<br>"eoOperationMode": "TRACKING",<br>"position": {<br>"latitude": 45.5,<br>"longitude": 25.5,<br>"altitude": 13.5<br>}<br>}<br>}
</td>
<td>
The first part of this data structure has already been described in Table 13.<br><br>The configuration part consists of:<br>id: string,<br>name, string<br>framerate: integer<br>azimuth: float, direction of the sensor in degrees with respect to the north<br>elevation: float, tilt angle of the sensor in degrees<br>zoom: float<br>msgInterval: float, in seconds<br>videoURL, string<br>eoOperationMode, enumeration, possible values are OFFLINE, TRACKING, IDLE<br>position, the position of this sensor, latitude and longitude in degrees and altitude in meters
</td>
</tr>
</table>

Table 16: EO sensor, heartbeat data structure (JSON representation)

# Chapter 6 Data flow

This chapter introduces the data flow in the ALFA system. The data flow is defined by means of the diagrams in Figure 6 and Figure 7 that show the flow of the data between the main processes of the system.

The diagrams include the following graphical elements, shown in Figure 5:

- Process: is a process or activity in which data is used or generated;
- External: represents an external source, user or repository of the data;
- Data Store: represents an internal physical or electronic repository of data, into and out of which data is stored and retrieved;
- Data Flow (connector): represents how data flows through the system, in physical or electronic form;
- Gate: represents the termination point of incoming and outgoing messages on a lower level diagram (that is, messages to and from processes depicted elsewhere).

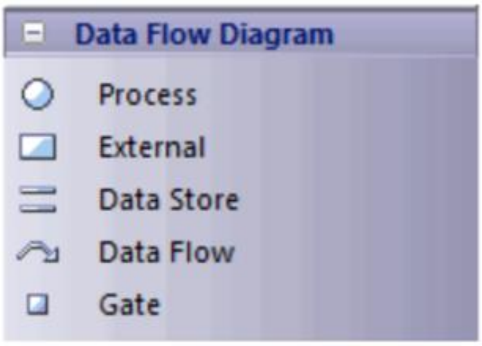


Figure 5: Data flow elements

The first diagram (see Figure 6) focuses on the Track entity and the Sensor data entity of the ALFA data model: the diagram shows all the parties involved in this data flow, where this data is generated and updated.

Instead, the diagram in Figure 7 focuses on the data related the heartbeat message and the configuration of the ALFA components.

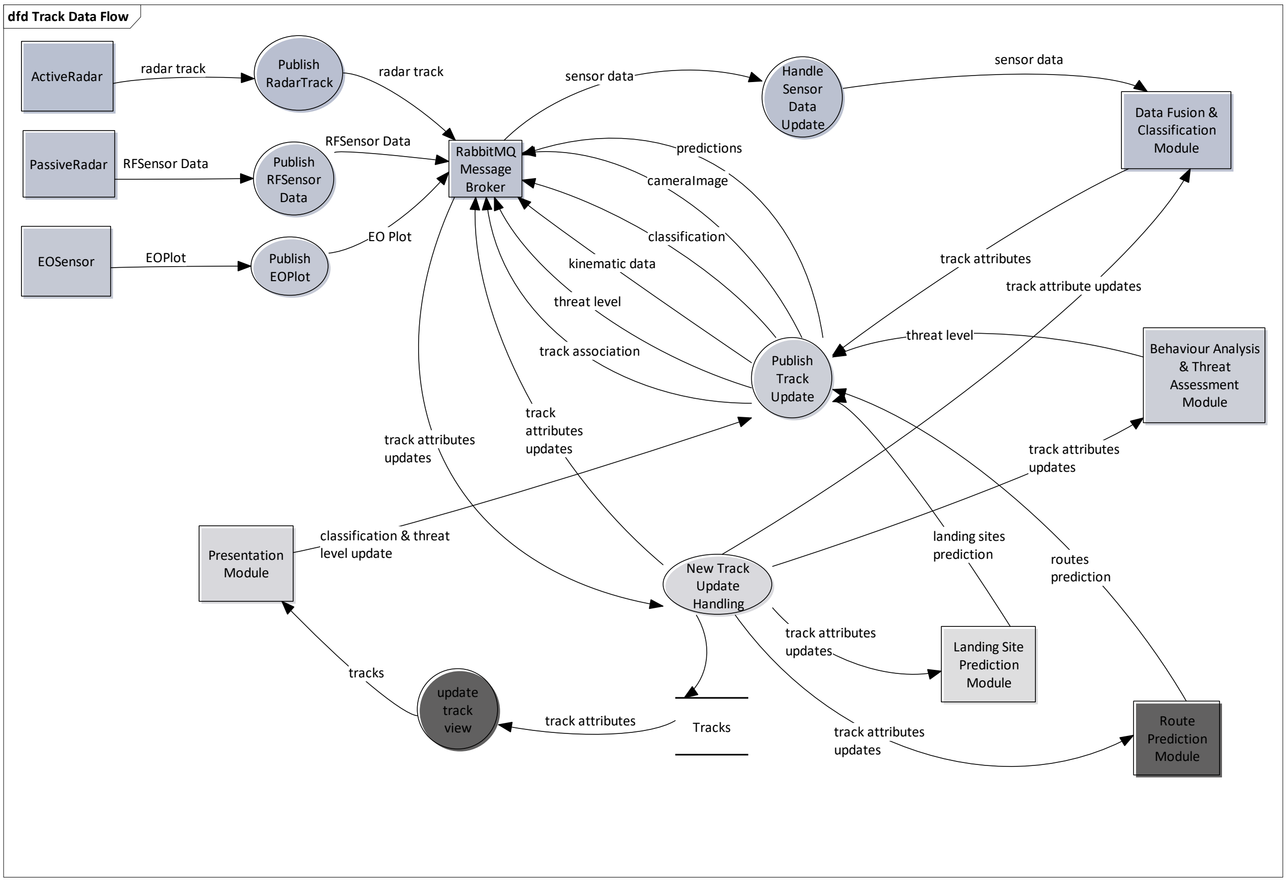


Figure 6: Track data flow

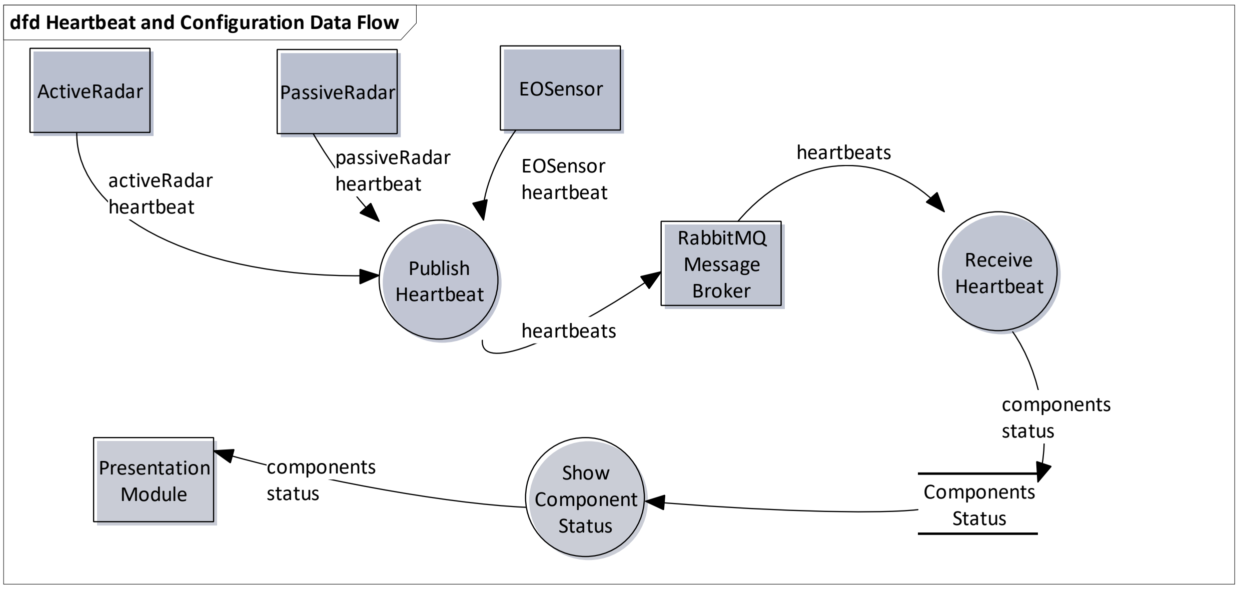


Figure 7: Heartbeat and configuration data flow

# Chapter 7 Summary and Conclusion

This report describes the activities performed within the framework of the ALFA project Work Package 3 (WP3) "Components and technologies", Task 3.6 "System Data Model".

The work in this task aimed at implementing all the JSON data structures required to represent all the information useful to build the ALFA situational awareness. Data structures have been modelled according to the data required and generated from all the components included in the ALFA framework, that are the ALFA sensors, the data processing components (fusion and classification, threat and behavioural analysis), the prediction components (landing site and route prediction) and the Human-Machine Interface (HMI).

The implementation phase of the data model followed an iterative and incremental process, and the data structures were refined according to the changes of the interfaces of the ALFA components, still in development during the execution of Task 3.6.

In Chapter 2 the updated logical models have been introduced; the UML diagrams representing the logical models have been refined according to the new requirements, and the changes in their entities/relationships have been also listed.

In the remaining Chapters of this document the ALFA data structures have been presented in a JSON format; the data structures to represent the information regarding the track and all its attributes, the sensor data and the sensor configuration, the heartbeats, have been presented.

Also an overview of the flows regarding the data managed by the ALFA components has been presented in Chapter 6.

# Chapter 8 List of Abbreviations

| Abbreviation | Translation |
|---|---|
| ADSB | Automatic Dependent Surveillance - Broadcast |
| ALFA | Advanced Low Flying Aircraft Detection and Tracking |
| EO | Electro Optic |
| GIS | Geographic Information system |
| HMI | Human-Machine Interface |
| JSON | JavaScript Object Notation |
| RF | Radio Frequency |
| UML | Unified Modelling Language |

# Chapter 9 Bibliography

[ALFA21] ALFA Consortium, D2.1. *ALFA architecture description*.

[ALFA22] ALFA Consortium, D2.2. *ALFA data model description*.