2020

ASCO – Integrations

VISCO GEO POSITION API

FOURPRO SOLUTIONS AS

Contents

Introduction

This document describes the GEO position integration API between Visco and FourPro Solutions, as part of the ASCO system delivery.

Currently, the URI of the APIs is pointed towards a development instance, which will be used for testing and making adjustments if needed. Once the system goes into production, a production API will be provided.

Authentication

Authentication to the push API is done by using an ApiKey, that is provided by FourPro. ApiKey is sent in the request header of the API request.

GEO Categories API

The GEO categories API returns all GEO object categories which are registered in the FourPro application.

Request headers

```
GET https://api.fourprosolutions.com/geo-locations/categories
Content-Type: application/json
Api-key: xxxxxxxxxxxxxxxxxx
```

Response body

```
{
    “id”: “b1b832d1-08b4-48d4-9f0e-9a6b0a347702”,
    “code”: “VESSEL”,
    “staticPositionDistance”: 20,
    “staticPositionDuration”: 5,
    “displayName”: ”Vessel”,
    “referenceType”: ”MMSI”
}
```

Response Details

Property	Type	Description
Id	Guid	GEO category technical identity
code	String	GEO category code
staticPositionDistance	Double	The maximum distance (in meters) between two positions to determine if a geo object has moved since last recorded position
staticPositionDuration	Double	The minimum time (in minutes) a geo object position has not changed to determine if a geo object has moved since last recorded position
displayName	String	GEO category display name
referenceType	String	Indicates the reference type property of geo objects within the geo category

GEO Position API

The GEO position API is used to send initial positions of objects (e.g. cranes, forklifts and vessels) that are visualized in a 3D model by Visco. It fetches all GEO objects within a radius (indicated by a latitude and longitude as a center, and a radius in meters).

Visco calls this API to get initial positions of objects that should be rendered in the model. Once that is completed, a real-time API will be used that streams “live data” from FourPro to Visco, containing updates to object positions.

Request headers

```
POST https://api.fourprosolutions.com/geo-locations/positions
Content-Type: application/json
Api-key: xxxxxxxxxxxxxxxxxx
```

Request body

```
{
    “latitude”: 58.923279,
    “longitude”: 5.59796,
    “radius”: 1000,
    “updatedAfter”: ”2019-01-01T00:00:00.000Z”,
    “categories”: [
       “vessel”,
       “truck”,
       “forklift”,
       “crane”
     ]
}
```

*Note, that “updatedAfter” and “categories” are optional.

Request Details

Property	Type	Description
latitude	Double	The latitude of the center point
longitude	Double	The longitude of the center point
radius	Integer	The radius from the center point, in meters
updatedAfter	DateTime (UTC)	When specified, only gets geo positions updated after this datetime (UTC time)
categories	List of strings	The categories which should be included in the result. When not specified, all categories will be included

Response body

```
[
   {
     “id”: “5176b290-8f1b-422b-8847-b78fc71e5c9b”,
     “sourceEntityType”: "common.vessel",
     “sourceEntityId”: "181db4ff-aba2-4da3-aaa9-f11c513404b0",
     “dateTimeStamp”: "2020-07-03T12:14:15.799396Z",
     “displayName”: "Normand Drott",
     “objectModel”: null,
     “objectReferenceNo”: null,
     “isMovableObject”: true,
     “isGeofenceSupported”: null,
     “trackingId”: null,
     “direction”: 275,
     “geoObjectCategoryId”: "b1b832d1-08b4-48d4-9f0e-9a6b0a347702",
     “currentLongitude”: 5.593289,
     “currentLatitude”: 58.92501,
     “categoryCode”: "VESSEL",
     “categoryDisplayName”: "Vessel",
     “categoryReferenceType”: "MMSI",
     “geoModelCode”: "50",
     “geoModelName”: "PSV"
   } ,
   …
]
```

Response Details

Property	Type	Description
id	Guid	GEO object technical identity
sourceEntityType	String	Indicates the source entity type
sourceEntityId	Guid	Source entity technical identity
dateTimeStamp	DateTime (UTC)	Datetime stamp of the last recorded position for the entity
displayName	String	Display name of the entity item
objectModel	String	Object model of the entity item
objectReferenceNo	String	Object reference number of the entity item
isMovableObject	Bool	Indicates if the entity is a movable object or not
isGeofenceSupported	Bool	Indicates if geo fence is supported for the geo object
trackingId	String	Tracking identity for the entity item
direction	Double	Direction of GEO object, in degrees
geoObjectCategoryId	Guid	Geo object category identity of the associated geo category of the entity item
currentLongitude	Double	Current longitude of the entity item

currentLatitude	Double	Current latitude of the entity item
categoryCode	String	Code of the associated geo category
categoryDisplayName	String	Display name of the associated geo category
categoryReferenceType	String	Reference type for the associated geo category
geoModelCode	String	Code of the associated GEO model
geoModelName	String	Name of the associated GEO model

GEO Vessels API

The GEO vessels API is used to get additional information about vessels that can be used in the Visco 3D model.

Request headers

```
POST https://api.fourprosolutions.com/geo-locations/vessels
Content-Type: application/json
Api-key: xxxxxxxxxxxxxxxxxx
```

Request body

```
{
   “items”: [
      “c193655f-1e8e-4d62-8634-af0d761c5b63”,
      “7419f104-1dea-491c-8acd-10e473182afd”
   ]
}
```

Request Details

Property	Type	Description
items	List of strings	The vessel identities for the vessels that should be included in the result. When using the “GEO Position API”, then the “sourceEntityId” from the result is used as the input to this list.

Response body

```
{
  “vessels”: [
   {
     “id”: “5176b290-8f1b-422b-8847-b78fc71e5c9b”,
     “imoNumber”: " 9602514",
     “name”: " ISLAND CRUSADER",
     “mmsiNumber”: " 257076000",
     “grossTonnageITC69”: 5400.0,
     “flag”: "Norway",
     “trackingId”: "257076000",
     “metersToBow”: 2,
     “metersToStern”: 45,
     “metersToPortSide”: 8,
     “metersToStarboardSide”: 7
   },
   …
 ]
}
```

Response Details

Property	Type	Description
id	Guid	GEO object technical identity
imoNumber	String	IMO number of the vessel
Name	String	Name of the vessel
mmsiNumber	String	MMSI number of the vessel
grossTonnageITC69	Double	Gross tonnage of the vessel
flag	String	Flag of the vessel
trackingId	String	Vessel tracking identity
metersToBow	Double	Meters from the AIS antenna to the bow
metersToStern	Double	Meters from the AIS antenna to the stern
metersToPortSide	Double	Meters from the AIS antenna to port side of the vessel
metersToStarboardSide	Double	Meters from the AIS antenna to the starboard side of the vessel

Geo Position Streaming API

The geo position streaming API provides updates to geo positions over SignalR.

SignalR is a high-level API that provides real-time data to applications, using multiple transports. The default transport is a web socket, but it will use other technologies if web sockets are not available. The libraries are open source and available for multiple platforms and languages. Documentation and samples are available from Microsoft:

https://docs.microsoft.com/en-us/aspnet/core/signalr/introduction?view=aspnetcore-5.0

FourPro provides access to the streaming API through the Azure SignalR service. Documentation and specifics for Azure SignalR can be found here:

https://docs.microsoft.com/en-us/azure/azure-signalr/

There is also a good tutorial on SignalR here:

https://www.freecodecamp.org/news/getting-started-with-signalr-in-azure-using-javascript/

To connect to the SignalR service you need to provide an endpoint (URL), an access key and a hub name (channel). In addition, you need to know which event or target method to listen for. For geo positions updates the values are:

Parameter	Value
Endpoint	https://fourpro-core-socket-dev.service.signalr.net
Access key	Provided separately by FourPro
Hub	visco
Event	setGeoLocation

The setGeoLocation event has a single argument which is a json object with the following properties:

setGeoLocation Details

Property	Type	Description
Timestamp	DateTime (UTC)	Datetime when the position was updated
Latitude	Double	The current latitude of the object
Longitude	Double	The current longitude of the object
TrackingId	String	Tracking identity (EPC) of the item
Direction	Double	The direction the object is moving in