ADAPTIVE RECOGNITION

HARBARD
API Documentation

This guide contains instructions for integrating your HARBARD API compatible traffic monitoring device into your application.

RELEASE DATE
21.09.2023.

HARBARD API

Compatible device versions include:

Einar-5	v2.3.0
Einar Gen 2	v1.3.0
Carmen Box/Nano	v1.4.0
Enforce Box/Nano	v1.4.0
MicroCAM Gen 2	v1.0.0

Document version: 21.09.2023.

Table of Contents

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Gettings started

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Gettings started

Introduction

This document is the API specification for devices compatible with the HARBARD API.

Multiple types of APIs are available - all accessed through HTTP protocol - but the main focus of this document is the **command** API and any further reference to APIs without specifying the type refers to the command API only.

API requests may accept input parameters in the HTTP REQUEST BODY as a JSON formatted text and the device replies with data in the HTTP RESPONSE BODY as a JSON formatted text. A command can be executed by sending a HTTP POST request to the appropriate URL.

> **Note:** API functions and properties not covered by this document may be changed or removed in the future without notice

Legend

The following is a list of expressions used in this document:

DEVICE_IP	The IP address or network hostname of the device
REQUEST	A HTTP request sent by the user to the device
RESPONSE	A HTTP response sent by the device to a REQUEST
HTTP BODY	Body part of a HTTP message (see http://en.wikipedia.org/wiki/HTTP_body_data)
EXCEPTION	A response given by the device when an error occured

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Gettings started

Authentication

Accessing resources on the device requires an authenticated session.

Login

To acquire a session the client must use the **Login** command available at

```
http://DEVICE_IP/login
```

and supply the **User** and **Password** of the selected user account.

Example login request to the device at 192.168.1.101:

```
POST /login HTTP/1.1
Host: 192.168.1.101
Content-Length: 35
Content-Type: application/json

{"User":"myusername","Password":"myuserpassword"}
```

On successful login the device will respond with a JSON object with a single field called **sid** that contains the unique session identifier of the authenticated session.

Example login reply **body** of a successful login:

```
HTTP/1.1 200 OK
Cache: no-cache
Content-Type: application/json
Content-Length: 61

{
        "Type" : "Response",
        "Data" : {
                "sid" : "60ab2b6b"
        }
}
```

Using the wrong username or password will result in an **InvalidCredentialException** error.

After successfully acquiring a session ID the rest of the device API can be accessed by sending the session id as a GET or COOKIE variable under the name **sid**.

Session lifetime

A session will time out if the user logs out, no new authenticated connections are initiated for a long period of time or the device reboots. Already active and authenticated connections are kept open even when the associated session ends.

Logout

Termination of a session is done by invoking the logout command at

```
http://DEVICE_IP/logout
```

with the session id (sid) sent as a COOKIE or a GET variable. This command will always succeed even if the session identifier is invalid.

Example logout request for session with sid *60ab2b6b*:

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
POST /logout?sid=60ab2b6b HTTP/1.1
Host: 10.10.22.234
Connection: keep-alive
Content-Length: 2
Content-Type: application/json

{}
```

Sessionless access

URLs may be accessed without an active session by providing credentials with each request. The username and password values may be sent with the appropriate **user** and **password** GET parameters.

```
http://DEVICE_IP/SOME/PATH/ON/DEVICE?user=USERNAME&password=PASSWORD
```

Credentials may also be sent using HTTP basic access authentication. Below is an example call using the popular cURL command line tool.

```
curl -v "http://USERNAME:PASSWORD@DEVICE_IP/SOME/PATH/ON/DEVICE"
```

The device does respond with authentication headers by default. Setting the **challenge** GET parameter to 1 on any device URL will force the device to issue a challenge with proper headers when an authenticated resource is requested or the authentication fails.

```
http://DEVICE_IP/SOME/PATH/ON/DEVICE?challenge=1
```

> **Note:** It is strongly recommended to use the session based authentication method. Sessionless access is provided for easy access while experimenting with APIs

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Gettings started

Executing commands

Accessing the API

The core functionality of the device can be accessed through the API URL which is

```
http://DEVICE_IP/api
```

The available methods are grouped into categories. Each category has a set of methods that can perform an action on the device or query the device for information.

To execute a method the client must invoke the full URL representing it which is as follows:

```
http://DEVICE_IP/api/CATEGORY/METHOD_NAME
```

For example the **GetDevice** method of the **System** category is executed by sending a request to the following URL:

```
http://DEVICE_IP/api/System/GetDevice
```

> **Note:** The API requires an authenticated user. The request must include a valid session identifier in the COOKIE or GET variable named **sid**

Input/ouput parameters

Every method's specification may include a **request** and/or a **response** object. These define the input and output parameters of the call. A request object is sent the same way as the login data: as a serialized JSON object in the HTTP POST BODY. The response data is encapsulated in an another layer and contains the response to the method call.

System/RunTest is a dedicated command for testing the API with example requests and responses below.

> **Note:** The response may contain additional undocumented top level keys beside **Type** and **Data** that can be safely ignored

Successful request

We send a RunTest request to the device with the text "First test" and ThrowException set to false.

```
POST /api/System/RunTest?sid=951a6d59 HTTP/1.1
Host: 192.168.1.100
Connection: keep-alive
Content-Type: application/json
Content-Length: 49

{ "Text": "First test", "ThrowException": false }
```

The device will respond with the following HTTP response:

```
HTTP/1.1 200 OK
Cache: no-cache
Content-Type: application/json
Content-Length: 115

{
        "Type" : "Response",
        "Data" : {
                "Text" : "Input received: First test",
                "Size" : 10,
                "User" : "admin"
        }
}
```

The **"Type" : "Response"** indicates that our request was successful and the device executed the method and replied with data.

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

The cURL command-line tool may be used to send the above request using the following call:

```
curl \
    -X POST \
    -H 'Content-Type: application/json' \
    -d '{ "Text": "First test", "ThrowException": false }' \
    "http://192.168.1.100/api/System/RunTest?sid=951a6d59"
```

Failed request with exception

We send a RunTest request to the device with the text "Second test" and ThrowException set to true forcing the device to respond with a TestException.

```
POST /api/System/RunTest?sid=951a6d59 HTTP/1.1
Host: 10.10.22.234
Connection: keep-alive
Content-Length: 44

{
        "Text": "Second test",
        "ThrowException": true
}
```

The device will respond with the following exception:

```
HTTP/1.1 200 OK
Cache: no-cache
Content-Type: application/json
Content-Length: 150

{
        "Type" : "Error",
        "Data" : {
                "ExceptionClass" : "TestException",
                "ErrorMessage" : "This is a test exception for testing error
reporting."
        }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Gettings started

Data types

The JSON format allows transfer of serveral data types but is limited compared to high-level programming languages. The reference of structures used in the device API contains a **Type** field that specifies the real data structure behind the items. The device will try to convert any input to the expected type or ignore the value on conversion failure.

Boolean

The **bool** type represents a boolean with a true or false value. This type can accept JSON booleans, literal "true" or "false" (case-insensitive) strings and numbers aswell.

Integers

The **int8**, **int16**, **int32** and **int64** types represent integers with a fixed bit width. If the input value doesn't fit into the specified bit length then it will be clamped to the nearest valid value.

> **Note:** When sending **int64** types keep in mind that some implementations cannot represent large 64 bit numbers. The device parses any string input as number when a numeric type is expected so it is recommended to send large numbers as strings.

Timestamps

The timestamp information is usually handled as an **int64** number representing a UTC timestamp in milliseconds. The epoch of the timestamp is

```
Monday, January 1, 1601 12:00:00 AM
```

also known as Windows epoch.

```
POSIX_TIME_IN_MS + 11644473600000 = WINDOWS_TIME_IN_MS
WINDOWS_TIME_IN_MS - 11644473600000 = POSIX_TIME_IN_MS
```

Double

The **double** type represents a standard (IEEE 754) 64 bit double-precision number.

GUID

The **guid** type is a string with a fixed format of **{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}** where **X** is a hexadecimal digit (0 to F).

Arrays of integers

Some methods require a long list of numbers (e.g.: coordinates). For this case there is an **Array** type that holds integers. The JSON array type is equivalent with this except **Array** can only contain numbers.

Unnamed keys

There are cases when the sequence of data that must be sent does not have any identifier (key). For this case the API handles numeric keys as unnamed keys. Any entry with a numeric key is considered unnamed and will be parsed accordingly. The actual number used does not make any difference since the numeric keys are not interpreted but the placement order of the elements are preserved.

An example of an object with named (Test1 & Test2) and unnamed (28, 91 & 4) keys:

```
{
    "91" : "Unnamed entry with arbitrary numeric key",
    "Test1" : "Named entry which will be the second in the list",
    "28" : "Another unnamed entry",
    "4" : "Third unnamed entry",
    "Test" : "Another named entry which is the last in the list (5th)"
}
```

Lists

The **List** type contains elements of the same type with unnamed keys.

Map

The **Map** type contains elements of the same type with named keys.

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Gettings started

Command options

Certain structures' parameters are limited to numeric ranges or a list of possible values. These possible values are called **Options**.

Structures with **Options** are commonly used in get/set method pairs (like System/GetNtpSettings and System/SetNtpSettings). When a command pair contains options the setter command will only accept data that fit the restrictions specified by the options in the getter command. Values outside of the specified boundaries will be ignored.

If an **Option** item is present inside the **Data** field of the response then its structure will be the exact copy of the **Data** structure where instead of the normal types and structures, there will be OptionNumericRange and OptionValueList structures describing the allowed values for each entry. The **Option** structure is ready-only and can be omitted when calling the appropriate setter command.

Example:

```
{
        "Type" : "Response",
        "Data" : {
                "TestItem" : 12,
                "TestList" : "Item1",
                "Options" {
                        "TestItem" : {
                                "Default" : 50,
                                "Minimum" : 0,
                                "Maximum" : 100
                        },
                        "TestList" : {
                                "Default" : "Item0",
                                "Values" : {
                                        "0" : "Item0",
                                        "1" : "Item1",
                                        "2" : "Item2",
                                        "3" : "Item3",
                                        "4" : "Item4",
                                        "5" : "Item5",
                                        "6" : "Item6",
                                }
                        }
                }
        }
}
```

The above example structure describes a response where two items are present: **TestItem** and **TestList**. The **Options** entry is present so there are restrictions on what can be set for **TestItem** and **TestList**.

- **TestItem** has a default value of 50 and accepts anything from 0 to 100
- **TestList** has a default value of "Item0" and accepts any of the elements listed under "Values"

> **Note:** The limits imposed by options are different from device to device based on product type and active settings

Gettings started

Features

Devices have different features available to the user based on product type and hardware configuration. These features can be queried using the System/GetDevice command. The response contains a map of modules under the **Modules** name with descriptors for each modules' capabilities. A descriptor may also contain a tree of strings defining available features. Feature lists are fixed and will not change unless the device is restarted.

Common modules

Module	Funtionality	Module descriptor
Analytics	Detectors and events	ModuleAnalytics
IO	External I/O ports	ModuleIO
Media	Audio and video streams	ModuleMedia

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Detectors & Engines

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Detectors & Engines

Types

The analytics module is divided into **engines** and **detectors**.

Engines are core modules running highly specialized algorithms and provide processed data sets for detectors to analyse. Engines do not emit events and don't provide user-queryable output. Depending on the device configuration the following engines may be available:

Engine	Description
ANPR engine	Performs license plate recognition (see Analytics/GetAnprEngine)
iTracking engine	Marks and tracks moving objects (see Analytics/GetTracker)
Motion engine	Performs motion detection on the whole image

Detectors are algorithms that analyze one or more data sets, media streams or peripherals and emit events when algorithm-specific criterias are met. For the events' properties see the Event structure. Depending on the device configuration the following detectors may be available:

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Detector	Reference
AlarmDetectorIO *Input port monitor*	DetectorConfigurationIO EventIO
AlarmDetectorTest *Detector for API testing*	DetectorConfigurationTest EventTest
For ANPR devices only:	
AlarmDetectorANPR *License plate detection*	DetectorConfigurationANPR EventANPR
For Enforcement devices only:	
AlarmDetectorEmergencyLane *Emergency lane violation*	DetectorConfigurationEmergencyLane EventEmergencyLane
AlarmDetectorForbiddenZone *Forbidden zone violation*	DetectorConfigurationForbiddenZone EventForbiddenZone
AlarmDetectorLane *Lane movement*	DetectorConfigurationLane EventLane
AlarmDetectorRedStop *Traffic light violation*	DetectorConfigurationRedStop EventRedStop
AlarmDetectorStoppedObject *Prohibited stop detection*	DetectorConfigurationStoppedObject EventStoppedObject
AlarmDetectorStopViolation *Stop sign violation*	DetectorConfigurationStopViolation EventStopViolation
AlarmDetectorTrafficLine *General line crossing*	DetectorConfigurationTrafficLine EventTrafficLine
AlarmDetectorUTurn *Illegal U-turn detection*	DetectorConfigurationUTurn EventUTurn
AlarmDetectorWhiteLineViolation *White line violation*	DetectorConfigurationWhiteLineViolation EventWhiteLineViolation
AlarmDetectorWrongTurn *Illegal turn violation*	DetectorConfigurationWrongTurn EventWrongTurn
AlarmDetectorWrongWay *Wrong-way driving detection*	DetectorConfigurationWrongWay EventWrongWay

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Detectors & Engines

Geometry

Some detectors and engines require some form of 2D configuration where polygons and lines define how the images are processed.

Coordinate system

The device uses the graphical coordinate system where X values increment to the right and Y values increment downwards. All coordinates are defined in a virtual coordinate system where values are calculated by the following formulas:

```
virtual_x = ( image_x / 16384 + image_width ) / image_width
virtual_y = ( image_y / 16384 + image_width ) / image_width

image_x = ( virtual_x * image_width + 16384 / 2 ) / 16384
image_y = ( virtual_y * image_width + 16384 / 2 ) / 16384
```

Geometry objects

The following is a list of common shapes for configuring detectors:

Name	Data type	Description
Straight line	GeometryLineSegment	Straight line with two points defining the start and end of the line
Segmented line	GeometryLine	Segmented line with at least one segment, each consisting of a start and end point
Ordered segmented lines	GeometryLineGroups	Groups of segmented lines where an order of groups is formed using indicies
Rectangle	GeometryRectangle	Rectangle where each side is parallel to the x or y axis of the image
Polygons	GeometryPolygons	List of polygons. A polygon has at least 3 points and an arbitary shape.

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Events

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Events

Modes

Devices support multiple modes for acquiring emitted events.

Live event query is a polling based event download where the user has to periodically check if new events are available.

Pros	Cons
Moderate latency Device buffers events	Event loss on slow connection No image or video content

Live event stream is a continuous multipart HTTP stream where new events are automatically streamed to the client with accompanying images.

Pros	Cons
Low latency Event image available	Event loss on connection error Event loss on slow connection No video content

Stored event query is a similiar mode to the live event query but uses requires a storage device. Supports filtering by detector and metadata.

Pros	Cons
Event image and video available Advanced filtering	Requires storage device Significant latency Client implementation may be complex

Stored event upload supports GDS and HTTP/HTTPS uploading of stored events to a remote server. The HTTP variant uses multipart POST requests to stream events with accompanying media data.

Pros	Cons
Event image and video available Event region of interest image available Compatible with most HTTP server implementation	Requires storage device Significant latency

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

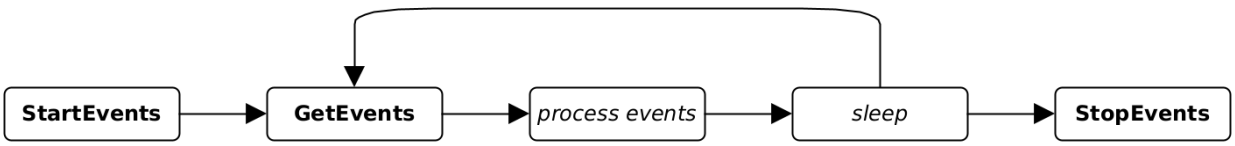
Events

Live event query

The easiest method of querying events is to poll the events using the Analytics/GetEvents call. To start polling initiate a buffer on the device using the Analytics/StartEvents call. This tells the device to allocate a buffer for the session and start queueing emitted events.

After the buffer is initiated the Analytics/GetEvents call can be used to periodically download collected events and flush the buffer. It is recommended to wait at least a second between two calls to prevent resource exhaustion or activation of the device's DoS protection.

When events are no longer needed the polling can be aborted using the Analytics/StopEvents call.



Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Events

Live event stream

Live events can be continuously downloaded by sending an authenticated GET request to the device on

```
http://DEVICE_IP/live/events
```

The device will respond with a **multipart/mixed** type connection and start sending events and associated images as they are emitted.

Events are sent with the multipart Content-Type of **application/json**. Additional headers include:

X-Event-Index	Index incrementing by one for each event. A gap in the indicies means the device was unable to send a packet probably due to slow connection and buffer limitations and dropped the event
X-Timestamp	Posix UTC timestamp of the event in milliseconds

Images are sent with the multipart Content-Type of **image/jpeg**. Additional headers include:

X-Image-Index	Index incrementing by one for each image. A gap in the indicies means the device was unable to send a packet probably due to slow connection and buffer limitations and dropped the image
X-Frame-Id	ID of sensor frame from which this JPEG was encoded
X-Frame-Timestamp	Monotonic timestamp of the image in milliseconds that is independent of the wall clock and is not affected by clock changes
X-Frame-Width	Image width
X-Frame-Height	Image height
X-Timestamp	Posix UTC timestamp of the image in milliseconds
X-Keep-Alive	Keepalive duration in seconds (see Keepalive below)

The **X-Event-Index** and **X-Image-Index** counters increment by one for each event or image queued respectively. An increment larger than one indicates that the device buffer filled up and data was dropped.

Stream format

The stream is in chronological order (except when device time changes) so events with the same timestamp will always be sent together. Images belonging to the events are always sent before the related event and have matching timestamps. If more than one event exists with the same timestamp the image will only be sent once.

The following example demonstrates the order of data when multiple events exist with the same timestamp:

Part #	Type	Source	Timestamp
1	image/jpeg		2021-01-11 19:32:03.978
2	application/json	Detector1	2021-01-11 19:32:03.978
3	application/json	Detector2	2021-01-11 19:32:03.978
4	image/jpeg		2021-01-11 19:39:56.004
5	application/json	Detector2	2021-01-11 19:39:56.004

Image attachment

Images can be disabled by setting the GET parameter **image** to zero:

```
http://DEVICE_IP/live/events?image=0
```

Resume stream

Network issues may close the connection prematurely and events may be lost while the client is reconnecting. To recover from such scenario the **timestamp** GET parameter can be used to provide the device with a starting point. The device will look up events in its internal buffer and send out any that matches or newer than the timestamp. The unit of timestamp is Windows milliseconds (same as the EventTime property of events).
Using the URL below the client will receive available events starting from 2021-05-14 12:09:41 UTC.

```
http://DEVICE_IP/live/events?timestamp=13265460581098
```

Filtering

The stream contains all events from all detectors by default. The events can be filtered by providing a comma separated list of detector ids with the **filter** GET parameters.
Using the URL below the client will only receive events from two detectors with ids **{6309907F-5708-47D1-B410-50F02C8882FB}** and **{B4C797C3-3AF3-4277-194D-9EF952A202A2}**.

```
http://DEVICE_IP/live/events?filter={6309907F-5708-47D1-B410-50F02C8882FB},
{B4C797C3-3AF3-4277-194D-9EF952A202A2}
```

Keepalive

During quiet periods the device may not transmit any data for a significant amount of time. Many network equipment may detect such connection as stale and close it prematurely.
Set the **keepalive** GET parameter to a duration in seconds to activate the keepalive messages. The device will automatically send an update message with Content-Type of **application/x-keepalive** when no data transfer was detected for the specified duration.

> Note: The device may override the keepalive parameter if set too low. The actual keepalive duration is always sent back in the X-Keep-Alive HTTP header. A zero value means keepalive is turned off.

Using the URL below the client will receive a keepalive message after a minute without any data transfer:

```
http://DEVICE_IP/live/events?keepalive=60
```

Below is an example update message:

```
--IPCamEventStreamBoundary
Content-Type: application/x-keepalive
Content-Length: 0


--IPCamEventStreamBoundary
```

Example stream

Example event stream request to the device at 192.168.1.101:

```
GET /live/events HTTP/1.1
Host: 192.168.1.101
Connection: keep-alive
Cookie: sid=60ab2b6b
```

Beginning of the response to the above request that contains one signal event and an image:

```
HTTP/1.1 200 OK
Pragma: no-cache
Expires: Thu, 01 Dec 2003 16:00:00 GMT
Connection: close
Content-Type: multipart/mixed; boundary=IPCamEventStreamBoundary
Cache: no-cache
Accept-Ranges: none
X-KeepAlive: 0
X-Timestamp: 1620986981098
X-Windows-Timestamp: 13265460581098

--IPCamEventStreamBoundary
Content-Type: image/jpeg
Content-Length: 498749
X-Timestamp: 1620986982002
X-Image-Index: 1
X-Frame-Id: 521699
X-Frame-Timestamp: 757030579
X-Frame-Width: 2560
X-Frame-Height: 1920

binary data
--IPCamEventStreamBoundary
Content-Type: application/json
Content-Length: 308
X-Event-Index: 1
X-Timestamp: 1620986982042

{
        "DetectorVersion" : 131072,
        "DetectorID" : "{6309907F-5708-47D1-B410-50F02C8882FB}",
        "DetectorClassID" : -835316578,
        "EventTime" : "13265460582042",
        "State" : "dsSignal",
        "EventCode" : 100,
        "EventInfo" : {},
        "EventID" : "{4F34A399-9E02-1846-ADA7-98A2798B46B9}",
        "DetectorEventType" : "detSignal"
}
```

Events

Stored event query

Devices with storage enabled can be queried for stored events using the Storage/GetEvents function.

It is recommended to first check the available time range on the storage device using the Storage/GetStatistics call then download in moderate segments. Specifying too large durations will result in slow or partial responses (see **Status** in StorageEvents).

Image

Images related to stored events can be downloaded with the following url:

```
http://DEVICE_IP/playback/image?
detector=DETECTOR_ID&event=EVENT_ID&timestamp=EVENT_TIMESTAMP
```

The GET parameters of **DETECTOR_ID**, **EVENT_ID** and **EVENT_TIMESTAMP** correspond to the values of **DetectorID**, **EventID** and **EventTime** from StorageEvents respectively.

> **Note:** A HTTP status code may be returned when the image is not available.

Video

Videos for the stored events may be requested with the following url:

```
http://DEVICE_IP/playback/video?start=START_TIMESTAMP&end=END_TIMESTAMP
```

The GET parameters specify the time range of the video using the same format as **EventTime**.

> **Note:** A HTTP status code may be returned when no video content is available in the specified time range.

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Events

Stored event upload

Devices with storage enabled can automatically upload events to an Adaptive Recognition Globessey Data Server (GDS) or a compatible HTTP/HTTPS server. This chapter describes the HTTP/HTTPS mode only.

Process

Upon activation the event uploader begins searching for events on the storage device in chronological order. Once an event is found a single standard POST request of **multipart/form-data** type is initiated to the configured URL and all data are transmitted.

Error handling

The server must respond with a HTTP status code of 200 for a successful transfer. Other responses are handled as follows:

- When a connection error occurs the uploader will retry indefinitely until the event is no longer available.
- Server may respond with a HTTP status code of 503 or 504 to signal that it is unable to accept requests. The uploader will retry indefinitely until the event is no longer available.
- When any other errors are encountered the uploader will retry a limited number of times then discard the event.

Request format

Event data and related media is uploaded in multipart fields identified by their **name**. The name and order of the fields are as follows:

Field name	MIME type	Count	Description
event_timestamp	text/plain	1	Field contains the posix UTC timestamp of the event in milliseconds
event_video_*NUM*	video/mp4	0≤	Related video content
event_image_*NUM*	image/jpeg	0≤	Related image content
event_cropped_image_*NUM*	image/jpeg	0≤	Region of interest cropped out from the original image
event_descriptor	application/ json	1	Event descriptor in JSON format (see Event)

The value of *NUM* is a zero-based index (e.g.: event_image_0, event_image_1, ...).

By default data is sent as standard form-data fields with only a **name** property but - using the web interface - a **filename** property can be added to media fields (image and video).

> **Note:** When using PHP POST fields are accessed through the $_POST variable but fields with filenames are available in the $_FILES variable

Field header when only names are sent:

```
Content-Disposition: form-data; name="FIELD_NAME"
Content-Type: MIME_TYPE
```

Field header of media data when filenames are configured aswell:

```
Content-Disposition: form-data; name="FIELD_NAME"; filename="FIELD_NAME.EXTENSION"
Content-Type: MIME_TYPE
```

Below is an example event upload transfer between a device and the server at 192.168.1.102 where the server's responses are marked red:

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
POST /http_upload_server_php/ar_http_upload.php HTTP/1.1
Host: 192.168.1.102
User-Agent: IntellioHttpPostUploader/1.0
Accept: */*
Cache-Control: no-cache
Content-Type: multipart/form-data; boundary=IntellioHttpPostUploaderBoundary
Content-Length: 4330662
Expect: 100-continue

HTTP/1.1 100 Continue

--IntellioHttpPostUploaderBoundary
Content-Disposition: form-data; name="event_timestamp"
Content-Type: text/plain

1631732906436
--IntellioHttpPostUploaderBoundary
Content-Disposition: form-data; name="event_video_0"
Content-Type: video/mp4

binary data
--IntellioHttpPostUploaderBoundary
Content-Disposition: form-data; name="event_image_0"
Content-Type: image/jpeg

binary data
--IntellioHttpPostUploaderBoundary
Content-Disposition: form-data; name="event_cropped_image_0"
Content-Type: image/jpeg

binary data
--IntellioHttpPostUploaderBoundary
Content-Disposition: form-data; name="event_descriptor"
Content-Type: application/json

{
        "DetectorVersion" : 131072,
        "DetectorID" : "{7D0829EA-E8FD-7546-92C7-3528E6216CBB}",
        "DetectorClassID" : 1968398405,
        "DetectorClass" : "AlarmDetectorANPR",
        "EventTime" : "13276206506436",
        "State" : "dsNormal",
        "EventCode" : 114,
        "EventInfo" : {
                "Text" : "ABC123",
                "Confidence" : 0.81999999284744262695,
                "Country" : "BIH",
                "CountryCode" : 113004,
                "Coords" : [
                        7808,
                        5606,
                        8992,
                        5632,
                        8992,
                        5843,
                        7808,
                        5818
                ],
                "BackgroundColor" : "",
                "DedicatedAreaColor" : "",
                "TextColor" : ""
        },
        "EventID" : "{93B5A26B-3069-E346-8E89-383ABA7A275C}",
        "DetectorEventType" : "detSimpleEvent"
}
```

```
--IntellioHttpPostUploaderBoundary--
HTTP/1.1 200 OK
Content-Length: 0
Content-Type: text/html; charset=UTF-8
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Miscellaneous

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Miscellaneous

GPIO state stream

Live I/O state can be continuously downloaded by sending an authenticated GET request to the device on

```
http://DEVICE_IP/live/io
```

The device will respond with a **multipart/mixed** type connection and start sending updates about I/O port state changes.

I/O state changes are sent with the multipart Content-Type of **application/json**. Additional headers include:

X-Timestamp	Posix UTC timestamp of the state change in milliseconds
X-Keep-Alive	Keepalive duration in seconds (see Keepalive below)

Stream format

The stream always starts with the last known states of the available ports. State changes are sent as GpioPortStateChange data structures. The stream is in chronological order (except when device time changes).

Filtering

The stream contains all state changes from all ports by default. The state changes can be filtered by providing a comma separated list of port names with the **filter** GET parameters.
Using the URL below the client will only receive state changes of two ports named **IN_0** and **IN_1**.

```
http://DEVICE_IP/live/io?filter=IN_0,IN_1
```

Keepalive

During quiet periods the device may not transmit any data for a significant amount of time. Many network equipment may detect such connection as stale and close it prematurely.
Set the **keepalive** GET parameter to a duration in seconds to activate the keepalive messages. The device will automatically send an update message with Content-Type of **application/x-keepalive** when no data transfer was detected for the specified duration.

> **Note:** The device may override the keepalive parameter if set too low. The actual keepalive duration is always sent back in the X-Keep-Alive HTTP header. A zero value means keepalive is turned off.

Using the URL below the client will receive a keepalive message after a minute without any data transfer:

```
http://DEVICE_IP/live/io?keepalive=60
```

Below is an example update message:

```
--IPCamIOStreamBoundary
Content-Type: application/x-keepalive
Content-Length: 0


--IPCamIOStreamBoundary
```

Example stream

Example I/O stream request to the device at 192.168.1.101:

```
GET /live/io HTTP/1.1
Host: 192.168.1.101
Connection: keep-alive
Cookie: sid=60ab2b6b
```

Beginning of the response to the above request that contains states for port IN_0 and OUT_0:

```
HTTP/1.1 200 OK
Pragma: no-cache
Expires: Thu, 01 Dec 2003 16:00:00 GMT
Connection: close
Content-Type: multipart/x-mixed-replace; boundary=IPCamIOStreamBoundary
Cache: no-cache
Accept-Ranges: none
X-KeepAlive: 0

--IPCamIOStreamBoundary
Content-Type: application/json
Content-Length: 104
X-Timestamp: 1620986982042

{
    "Active" : false,
    "Port" : "IN_0",
    "Timestamp" : "13265460582042"
    "Type" : "Input",
}

--IPCamIOStreamBoundary
Content-Type: application/json
Content-Length: 106
X-Timestamp: 1620986982042

{
    "Active" : false,
    "Port" : "OUT_0",
    "Timestamp" : "13265460582042"
    "Type" : "Output",
}

--IPCamIOStreamBoundary
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Functions

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

category
Analytics

The **Analytics** category is a collection of methods for managing analytics engines, detectors and querying events.

Methods

Method	Description
Analytics/GetEvents	Get the buffered events
Analytics/StartEvents	Start the event buffering for the calling session
Analytics/StopEvents	Stop the event buffering for the calling session
Analytics/TriggerEngine	Manually trigger an analytics engine
ANPR	
Analytics/GetAnprEngine	Get the current configuration of the ANPR engine
Analytics/GetAnprEngineDefaults	Get the default configuration of the ANPR engine
Analytics/GetAnprEngineState	Get the current state of the ANPR engine
Analytics/SetAnprEngine	Change the configuration of the ANPR engine
Detectors	
Analytics/CreateDetector	Create a new detector instance
Analytics/DeleteAllDetectors	Delete all detector instances
Analytics/DeleteDetector	Delete the detector instance
Analytics/DisableDetector	Disable the detector
Analytics/EnableDetector	Enable the detector
Analytics/GetDetector	Get the configuration of the detector
Analytics/GetDetectorDefaults	Get the default configuration of a detector type
Analytics/GetDetectorState	Get the state of the detector
Analytics/GetDetectors	Get the active detector instances on this device
Analytics/GetSupportedDetectors	Get the supported detector types on this device
Analytics/SetDetector	Set the configration of the detector
Tracker	
Analytics/GetTracker	Get the current configuration of the tracker
Analytics/GetTrackerDefaults	Get the default configuration of the tracker
Analytics/SetTracker	Change the configuration of the tracker

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/CreateDetector

Create a new detector instace with the specified type and unique id.

Specification

User level	ADMINISTRATOR
Request data	DetectorCreateConfiguration
Response data	*none*
Exceptions	**DetectorIdMissingExecption**: The ID of the new detector instance must be specified. **DetectorIdExistsException**: The ID of the new detector instance is already in use. **DetectorLimitReachedException**: Cannot create more detectors of this type. See [strong]InstanceLimit[/strong] in Analytics/GetSupportedDetectors. **InvalidDetectorTypeException**: The specified detector type is unknown. See [strong]DetectorClass[/strong] in Analytics/GetSupportedDetectors.

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/DeleteAllDetectors

Deletes all detector instances except built-in detectors

Specification

User level	ADMINISTRATOR
Request data	*none*
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/DeleteDetector

Deletes a detector instance. Built-in detectors cannot be deleted.

See also: Analytics/DisableDetector, Analytics/EnableDetector, Analytics/GetDetector, Analytics/GetDetectorState

Specification

User level	ADMINISTRATOR
Request data	DetectorRequest
Response data	*none*
Exceptions	**DetectorNotFoundException**: The specified detector does not exist. **AccessDeniedException**: The detector specified cannot be removed because it is a built-in detector.

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/DisableDetector

Disable the selected detector. A disabled detector will not process signals and analytics. A disabled detector will not emit events except ones that indicate change in configuration and initialization state.

See also: Analytics/DeleteDetector, Analytics/EnableDetector, Analytics/GetDetector, Analytics/GetDetectorState

Specification

User level	ADMINISTRATOR
Request data	DetectorRequest
Response data	*none*
Exceptions	**DetectorNotFoundException**: The specified detector does not exist.

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/EnableDetector

Enable the selected detector so it may resume processing signals and analytics. Enabling an already enabled detector has no effect.

See also: Analytics/DeleteDetector, Analytics/DisableDetector, Analytics/GetDetector, Analytics/GetDetectorState

Specification

User level	ADMINISTRATOR
Request data	DetectorRequest
Response data	*none*
Exceptions	**DetectorNotFoundException**: The specified detector does not exist

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetAnprEngine

Get the current configuration of the ANPR engine

See also: Analytics/GetAnprEngineDefaults, Analytics/SetAnprEngine

Specification

User level	ADMINISTRATOR
Request data	*none*
Response data	AnprEngineConfiguration
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetAnprEngineDefaults

Get the default configuration of the ANPR engine

See also: Analytics/GetAnprEngine, Analytics/SetAnprEngine

Specification

User level	ADMINISTRATOR
Request data	*none*
Response data	AnprEngineConfiguration
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetAnprEngineState

Get the current state of the ANPR engine

Specification

User level	ADMINISTRATOR
Request data	*none*
Response data	AnprEngineState
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetDetector

Get the current configuration of the selected detector. The content of the response varies depending on the detector type.

See also: Analytics/DeleteDetector, Analytics/DisableDetector, Analytics/EnableDetector, Analytics/GetDetectorDefaults, Analytics/GetDetectorState, Analytics/SetDetector

Specification

User level	ADMINISTRATOR
Request data	DetectorRequest
Response data	Detector
Exceptions	**DetectorNotFoundException**: The specified detector does not exist

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetDetectorDefaults

Get the default configuration of the specified detector type. The default parameters will be used when creating a detector without specifying any detector specific configuration.

See also: Analytics/GetDetector, Analytics/SetDetector

Specification

User level	ADMINISTRATOR
Request data	DetectorClassRequest
Response data	Detector
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetDetectorState

Get the current state of the detector.[br] The detector state indicates if the detector is properly initialized and ready to process data.

See also: Analytics/DeleteDetector, Analytics/DisableDetector, Analytics/EnableDetector, Analytics/GetDetector

Specification

User level	USER
Request data	DetectorRequest
Response data	DetectorState
Exceptions	**DetectorNotFoundException**: The specified detector does not exist.

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetDetectors

Get the active detector instances on this device

Specification

User level	USER
Request data	*none*
Response data	DetectorList
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetEvents

Get all events collected since the last call or since the buffering was started. Events may be dropped when the internal buffer allocated for this session is full.

Specification

User level	USER
Request data	*none*
Response data	BufferedEvents
Exceptions	**StreamNotStartedException**: Event buffering was not started on this session

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetSupportedDetectors

Lists all of the supported detector types on this device along other statistics of each type

Specification

User level	USER
Request data	*none*
Response data	SupportedDetectors
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetTracker

Get the current configuration of the tracker

See also: Analytics/GetTrackerDefaults, Analytics/SetTracker

Specification

User level	ADMINISTRATOR
Request data	*none*
Response data	TrackerConfiguration
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/GetTrackerDefaults

Get the default parameters used by the tracker when parameters are missing during a Analytics/SetTracker configuration.

> **See also:** Analytics/GetTracker, Analytics/SetTracker

Specification

User level	ADMINISTRATOR
Request data	*none*
Response data	TrackerConfiguration
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/SetAnprEngine

Change the configuration of the ANPR engine

See also: Analytics/GetAnprEngine, Analytics/GetAnprEngineDefaults

Specification

User level	ADMINISTRATOR
Request data	AnprEngineConfiguration
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/SetDetector

Update the configuration of the selected detector. The required configuration parameters depend on the detector type.

See also: Analytics/GetDetector, Analytics/GetDetectorDefaults

Specification

User level	ADMINISTRATOR
Request data	Detector
Response data	*none*
Exceptions	**DetectorNotFoundException**: The specified detector does not exist

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/SetTracker

Change the configuration of the tracker

See also: Analytics/GetTracker, Analytics/GetTrackerDefaults

Specification

User level	ADMINISTRATOR
Request data	TrackerConfiguration
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/StartEvents

Start the event buffering on this session. If the event buffering was already started this method does nothing. Buffered events can be queried using the Analytics/GetEvents method and stopped with Analytics/StopEvents.

The events can be filtered by detectors by specifying their IDs. For more details see the input parameters of this method.

Specification

User level	USER
Request data	BufferedEventsRequest
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/StopEvents

Stop the event buffering for the calling session

Specification

User level	USER
Request data	*none*
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Analytics/TriggerEngine

Manually trigger an analytics engine

Specification

User level	USER
Request data	AnalyticsEngineTrigger
Response data	AnalyticsEngineTriggerResponse
Exceptions	**InvalidTriggerException**: The specified engine does not exist or doesn't support triggers.

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

category
Storage

The **Storage** category is a collection of methods for managing the on-board storage and querying stored data.

Methods

Method	Description
Storage/GetEvents	Perform a query on the stored events
Storage/GetStatistics	Get general statistics from the storage subsystem

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Storage/GetEvents

Get the list of events from the storage device that match the specified parameters.

Specification

User level	USER
Request data	StorageEventsRequest
Response data	StorageEvents
Exceptions	**EventsNotFoundException**: Events could not be retrieved due to read error

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

Storage/GetStatistics

Get general statistics from the storage subsystem

Specification

User level	USER
Request data	*none*
Response data	StorageStatistics
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

category
System

The **System** category is a collection of methods that allow configuring general aspects of the device like name, time or user accounts. When connecting to a device for the first time it is recommneded to use the System/GetDevice method to get general information about it.

Methods

Method	Description
System/ClearSecurityHistory	Release the block on all clients that are currently banned
System/FactoryReset	Factory reset the settings and reboot
System/GetDevice	Get general information about the device
System/GetSecurityHistory	List the active session and blocked clients
System/GetSecuritySettings	Get the security settings
System/GetVersion	Get the version of the JSON API
System/Reboot	Start the reboot of the device
System/RunTest	Testing method for checking JSON API
System/SetDevice	Change the name and description of the device
System/SetSecuritySettings	Change the security settings
Date & time	
System/GetNtpSettings	Get the NTP settings
System/GetTime	Get the current timestamp
System/SetNtpSettings	Change the NTP settings
System/SetTime	Change the current timestamp
I/O	
System/GetGpioSettings	Get the available digital inputs and outputs on this device
System/GetGpioStates	Get the last known state of available digital inputs and outputs on this device
System/SetGpioInputSettings	Change the configuration of a digital input port
System/SetGpioOutput	Change the state of a digital output port
System/SetGpioOutputSettings	Change the configuration of a digital output port
System/TriggerGpioOutput	Send an impulse to a digital output port
Users	
System/AddUser	Add a new user account
System/DeleteUser	Remove a user account
System/GetCurrentUser	Get the user of the current session
System/GetUsers	List all users accounts on the device. The password field is present but will not contain any information.
System/ModifyUser	Modify the properties of a user account

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/AddUser

Add a new user account

See also: System/DeleteUser, System/GetCurrentUser, System/ModifyUser

Specification

User level	ADMINISTRATOR
Request data	User
Response data	*none*
Exceptions	**UserValueException**: An invalid parameter was sent **UserExistsException**: A user with the same name already exists

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/ClearSecurityHistory

Release the block on all clients that are currently banned

Specification

User level	ADMINISTRATOR
Request data	*none*
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/DeleteUser

Remove a user account

See also: System/AddUser, System/GetCurrentUser, System/ModifyUser

Specification

User level	ADMINISTRATOR
Request data	UserId
Response data	*none*
Exceptions	**DeleteSelfException**: A user cannot remove its own account **UserNotExistsException**: Tried to remove a non-existing user account

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/FactoryReset

Request a soft factory reset of the device. The device will restore all except the network settings to factory defaults and request a reboot. For a full factory reset the physical reset button on the device must be pressed if available.

Specification

User level	ADMINISTRATOR
Request data	*none*
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/GetCurrentUser

Get the user of the current session

See also: System/AddUser, System/DeleteUser, System/ModifyUser

Specification

User level	USER
Request data	*none*
Response data	UserInfo
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/GetDevice

This method is used for discovering the capabilities of a device after a successful authentication. The response contains the availability of various modules, firmware and product information and lists of supported features.

See also: System/SetDevice

Specification

User level	USER
Request data	*none*
Response data	SystemSettingsResponse
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/GetGpioSettings

Get the available digital inputs and outputs on this device

Specification

User level	USER
Request data	*none*
Response data	GpioSettings
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/GetGpioStates

Get the last known state of available digital inputs and outputs on this device

Specification

User level	USER
Request data	*none*
Response data	GpioStates
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/GetNtpSettings

Get the NTP settings

See also: System/SetNtpSettings

Specification

User level	USER
Request data	*none*
Response data	NtpSettings
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/GetSecurityHistory

List the active session and blocked clients

Specification

User level	USER
Request data	*none*
Response data	SecurityHistory
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/GetSecuritySettings

Get the security settings of the device tha controls allowed authentication attemps and blocking duration. If the number of authentication fails by a client exceeds the limit the client will be blocked for the specified duration and all authentication attemps - regardless of the used credentials - will be ignored until the block expires.

See also: System/SetSecuritySettings

Specification

User level	USER
Request data	*none*
Response data	SecuritySettings
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/GetTime

Get the current timestamp

See also: System/SetTime

Specification

User level	USER
Request data	*none*
Response data	TimeSettings
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/GetUsers

List all users accounts on the device. The password field is present but will not contain any information.

Specification

User level	ADMINISTRATOR
Request data	*none*
Response data	Users
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/GetVersion

Get the version of the JSON API. The individual commands' structure and the commands itself may change without the API version changing. Only major structural or workflow changes are reflected here.

Specification

User level	USER
Request data	*none*
Response data	ApiVersion
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/ModifyUser

Modify the properties of a user account

See also: System/AddUser, System/DeleteUser, System/GetCurrentUser

Specification

User level	ADMINISTRATOR
Request data	User
Response data	*none*
Exceptions	**UserValueException**: An invalid parameter was sent **ModifySelfException**: A user cannot modify its own role **UserNotExistsException**: Tried to modify a non-existing user account

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/Reboot

Request the device the reboot. The device will reboot shortly after the request.

Specification

User level	ADMINISTRATOR
Request data	RebootSettings
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/RunTest

This method is used for testing the functionality of the JSON API and making implementation easier. This method does not execute actual logic on the device but just returns canned responses.

Specification

User level	USER
Request data	TestInput
Response data	TestOutput
Exceptions	**TestException**: This is an exception thrown when the [strong]ThrowException[/strong] of the input is set to true

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/SetDevice

Change the name, description and location of the device usually visible on user interfaces.

See also: System/GetDevice

Specification

User level	ADMINISTRATOR
Request data	*none*
Response data	SystemSettings
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/SetGpioInputSettings

Change the configuration of a digital input port

See also: System/SetGpioOutput, System/SetGpioOutputSettings, System/TriggerGpioOutput

Specification

User level	ADMINISTRATOR
Request data	GpioInputPort
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/SetGpioOutput

Change the state of a digital output port

See also: System/SetGpioInputSettings, System/SetGpioOutputSettings, System/TriggerGpioOutput

Specification

User level	OPERATOR
Request data	GpioOutputPortState
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/SetGpioOutputSettings

Change the configuration of a digital output port

See also: System/SetGpioInputSettings, System/SetGpioOutput, System/TriggerGpioOutput

Specification

User level	ADMINISTRATOR
Request data	GpioOutputPort
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/SetNtpSettings

Change the NTP settings

See also: System/GetNtpSettings

Specification

User level	ADMINISTRATOR
Request data	NtpSettings
Response data	*none*
Exceptions	*none*

function

System/SetSecuritySettings

Change the security settings

See also: System/GetSecuritySettings

Specification

User level	ADMINISTRATOR
Request data	SecuritySettings
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/SetTime

Change the current timestamp

See also: System/GetTime

Specification

User level	ADMINISTRATOR
Request data	TimeSettings
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

function

System/TriggerGpioOutput

Send an impulse to a digital output port

See also: System/SetGpioInputSettings, System/SetGpioOutput, System/SetGpioOutputSettings

Specification

User level	OPERATOR
Request data	GpioPortId
Response data	*none*
Exceptions	*none*

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Data structures

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

ActiveSession

Active session information

Structure

Parameter	Type	Description
LastSeen	int64	Elapsed time in milliseconds since the last activity on this session
Source	string	Source of the session, usually an IP address
User	string	The authenticated user name on the session

Pseudo code

```
{
    "LastSeen": ...,
    "Source": "...",
    "User": "..."
}
```

structure

AnalyticsEngineTrigger

Properties of a manual engine trigger.

The **Count** property defines the number of detection attempts before the trigger is considered done. By setting this property to zero you can cancel still active manual triggers.

See also: Analytics/TriggerEngine

Structure

Parameter	Type	Description
Count	int32	Number of triggers to issue
Target	string	Name of engine to trigger (only "Anpr" is supported)
TriggerSource		Advanced settings for Software trigger mode

Pseudo code

```
{
    "Count": ...,
    "Target": "...",
    "TriggerSource":
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

AnalyticsEngineTriggerResponse

Properties of a manual engine trigger.

See also: Analytics/TriggerEngine

Structure

Parameter	Type	Description
Name	string	Name of the trigger
Source	string	Name of the triggered engine
Timestamp	int64	Timestamp when the trigger was received by the device

Pseudo code

```
{
    "Name": "...",
    "Source": "...",
    "Timestamp": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

AnprEngineConfiguration

Configuration of the ANPR engine.
The engine only operates inside the specified mask and emits an event for each recognized license plate that meet the configured criteria.

By default the engine is automatically triggered by the on-board plate finder and accepts external triggers aswell. This can be changed using the **TriggerModes** option. When using external triggers the engine reads license plates until the specified count is reached. Setting the **InterruptOnRecognition** to true aborts the read after the first successful license plate read. The on-board plate finder - if enabled - is paused while there is an active external trigger.

Available trigger modes are:
- **PlateFinder:** Engine is triggered automatically by the on-board license plate finder
- **Software:** Engine can be triggered using the Analytics/TriggerEngine call
- **Hardware:** Engine is triggered by a configured GPIO input port

The **HardwareTriggerSettings/TriggerMode** option controls how the activation of the input port triggers the engine when hardware trigger is used.
- **Impulse:** Activation of the input port triggers the engine to make **ReadCount** number of successful reads
- **State:** The engine continouously tries to read license plates while the input port is active

> See also: Analytics/GetAnprEngine, Analytics/GetAnprEngineDefaults, Analytics/SetAnprEngine

Structure

Parameter	Type	Description
Config		

Pseudo code

```
{
    "Config":
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

AnprEngineState

Current state of the ANPR engine

See also: Analytics/GetAnprEngineState

Structure

Parameter	Type	Description
Config		

Pseudo code

```
{
    "Config":
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

ApiVersion

JSON API information

See also: System/GetVersion

Structure

Parameter	Type	Description
Version	int32	Current version of the JSON API

Pseudo code

```
{
    "Version": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

BufferedEvents

Query collected events in a sessions buffer.

When Analytics/GetEvents is called all events from the internal buffer are returned then deleted and subsequent calls will only return events emitted after this call. If too many events are emitted or the duration between two Analytics/GetEvents calls are too long the internal buffer may fill up and events may be discarded until the buffer is emptied. The number of discarded events can be monitored using the **DiscardedEvents** property.

> See also: Analytics/GetEvents

Structure

Parameter	Type	Description
DiscardedEvents	int32	Number of events discarded since the start of buffering
EventList	Event	List of events
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "DiscardedEvents": ...,
    "EventList":
    {
        "Config":
        {
            "BuiltIn": ...,
            "Class": "...",
            "Description": "...",
            "DetectorClassID": ...,
            "DetectorID": "{...}",
            "DisplayName": "...",
            "Enabled": ...,
            "FpsLimit": ...,
            "RestoreDelayMs": ...,
            "Version": ...,
            "ViolationTimeMs": ...
        },
        "DetectorClassID": ...,
        "DetectorEventType": "...",
        "DetectorID": "{...}",
        "DetectorVersion": ...,
        "EventCode": ...,
        "EventID": "{...}",
        "EventTime": ...,
        "EventTriggerTime": ...,
        "State": "..."
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

BufferedEventsRequest

Parameters for starting event buffering on the current session.

When the **Filter** parameter is filled with detector IDs only events from those detectors will be buffered and other events will be discarded. If not specified or left empty all events will be available for query.

See also: Analytics/StartEvents

Structure

Parameter	Type	Description
Filter	List/guid	List of detector IDs

Pseudo code

```
{
    "Filter":
    {
        "0": "{...}",
        "1": "{...}"
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

Detector

Configuration of the detector. The contents of this data collection depends on the selected detector type.

See also: Analytics/GetDetector, Analytics/GetDetectorDefaults, Analytics/SetDetector

Structure

Parameter	Type	Description
Config	DetectorConfiguration	Contains further configuration options specific to the detector type
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorID	guid	(optional) Unique ID of the detector instance.[br]This option should only be specified when requesting data from the device.

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorID": "{...}"
}
```


Adaptive Recognition America
Adaptive Recognition Hungary
Adaptive Recognition Nordic
Adaptive Recognition Singapore
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorClassRequest

Property the uniquely identifies a detector type.

See also: Analytics/GetDetectorDefaults

Structure

Parameter	Type	Description
DetectorClass	string	String id of the detector type

Pseudo code

```
{
    "DetectorClass": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfiguration

Inherited by: DetectorConfigurationANPR, DetectorConfigurationIO, DetectorConfigurationTest, TrackingDetectorConfiguration

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

```
{
    "BuiltIn": ...,
    "Class": "...",
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationANPR

Configuration of the ANRP detector.

By default the detector signals for all license plates. When whitelist is enabled events will only be emitted for license plates found in the filter.

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
Filter	string	New-line separated list of license plates to signal for
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
Whitelist	bool	Enable filter usage

Pseudo code

```
{
    "BuiltIn": ...,
    "Class": "...",
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "Filter": "...",
    "FpsLimit": ...,
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...,
    "Whitelist": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationEmergencyLane

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
Masks	List/Array/ int16	Mask defining the working area of the detector (see GeometryPolygons)
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "Masks":
    {
        "0": [ ..., ..., ... ],
        "1": [ ..., ..., ... ]
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationForbiddenZone

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
Masks	List/Array/ int16	Mask defining the working area of the detector (see GeometryPolygons)
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "Masks":
    {
        "0": [ ..., ..., ... ],
        "1": [ ..., ..., ... ]
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationIO

Configuration of the IO detector.

The detector will signal when the configured input port leaves the normal state and ends when the port normalizes.

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
InputPort	string	Name of the input port to monitor
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

```
{
    "BuiltIn": ...,
    "Class": "...",
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "InputPort": "...",
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationLane

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
Masks	List/Array/ int16	Mask defining the working area of the detector (see GeometryPolygons)
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "Masks":
    {
        "0": [ ..., ..., ... ],
        "1": [ ..., ..., ... ]
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationRedStop

Detector monitors for objects that cross [strong]Lines[/strong] and leave the area through [strong]ExitLines[/strong] after the light turns red and [strong]GracePeriod[/strong] had elapsed. The [strong]TrafficLight[/strong] type can be configured to be [strong]RogColumn[/strong] (vertical road traffic light), [strong]RrwRailRoad[/strong] (triangular railroad light) or [strong]RrwRailRoad2[/strong] (horizontal railroad light).

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
Direction	string	[em]unused[/em]
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
ExitLines	IndexedTrackingDetectorLines	List of segments defining the exit line of the detector
Id	int8	Index of the line
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
GracePeriod	int64	The grace period in milliseconds after the a light turns red where crossing is still allowed
Lines	GeometryLineSegment	List of segments defining the entry line for the detector (see GeometryLine)
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
TrafficLight		
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "Direction": "...",
    "DisplayName": "...",
    "Enabled": ...,
    "ExitLines":
    {
        "Id": ...,
        "X0": ...,
        "X1": ...,
        "Y0": ...,
        "Y1": ...
    },
    "FpsLimit": ...,
    "GracePeriod": ...,
    "Lines":
    {
        "X0": ...,
        "X1": ...,
        "Y0": ...,
        "Y1": ...
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "TrafficLight": ,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationStopViolation

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
Direction	string	Direction of crossing that is monitored
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
Lines	GeometryLineSegment	List of segments (see GeometryLine). Objects must stop before this line before crossing it.
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "Direction": "...",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "Lines":
    {
        "X0": ...,
        "X1": ...,
        "Y0": ...,
        "Y1": ...
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationStoppedObject

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
Masks	List/Array/ int16	Mask defining the working area of the detector (see GeometryPolygons)
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "Masks":
    {
        "0": [ ..., ..., ... ],
        "1": [ ..., ..., ... ]
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationTest

Configure the test detector.

Based on the configuration the detector will emit signal/restore pairs or plain events periodically.
When **Timeout** is larger than zero the detector repeats the cycle of emiting a signal after **Interval** and restoring it after **Timeout**.
When **Timeout** is set to zero the detector will simply emit an event every **Interval** milliseconds.

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
Interval	int64	Duration of normal state in milliseconds
RestoreDelayMs	int64	[em]unused[/em]
Timeout	int64	Duration of signalling state in milliseconds
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

```
{
    "BuiltIn": ...,
    "Class": "...",
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "Interval": ...,
    "RestoreDelayMs": ...,
    "Timeout": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationTrafficLine

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
Direction	string	Direction of crossing that is monitored
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
Lines	GeometryLineSegment	List of segments defining the line that is monitored for crossing objects (see GeometryLine)
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "Direction": "...",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "Lines":
    {
        "X0": ...,
        "X1": ...,
        "Y0": ...,
        "Y1": ...
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationUTurn

Detector monitors for objects that perform a complete U-turn while crossing the line in the specified direction.

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
Direction	string	Direction of crossing that is monitored
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
Lines	GeometryPolygons	List of segments defining the line that is parallel and inbetween the two straights of the U path (see GeometryLine)
Masks	List/Array/int16	List of masks. Each mask is a list of coordinates where odd and even indicies are x and y coordinates of a corner in the polygon (x0, y0, x1, y1, ...).
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "Direction": "...",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "Lines":
    {
        "Masks":
        {
            "0": [ ..., ..., ... ],
            "1": [ ..., ..., ... ]
        }
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationWhiteLineViolation

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
Direction	string	Direction of crossing that is monitored
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
Lines	GeometryLineSegment	List of segments defining the white line on the road surface (see GeometryLine)
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "Direction": "...",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "Lines":
    {
        "X0": ...,
        "X1": ...,
        "Y0": ...,
        "Y1": ...
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationWrongTurn

Detector monitors for objects that cross the lines in the order of their sequence number.

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
LineGroups	GeometryLineGroup	Mask defining the working area of the detector (see GeometryLineGroups)
Lines	GeometryLineSegment	List of line segments
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point
SequenceNumber	int32	Numeric id of this group for ordering
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "LineGroups":
    {
        "Lines":
        {
            "X0": ...,
            "X1": ...,
            "Y0": ...,
            "Y1": ...
        },
        "SequenceNumber": ...
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorConfigurationWrongWay

Detector monitors for objects that move in the specified direction inside the mask. The monitored direction can be extended using [strong]AngleRange[/strong]. For example the value of [em]Angle=90[/em] and [em]AngleRange=10[/em] sets the monitored direction range to 80° - 100°.

Structure

Parameter	Type	Description
Angle	double	Angle of forbidden direction in degrees. Value of 0° points right and 90° points up.
AngleRange	double	Extends monitored angle in both direction with this degree value
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
LocationX	int32	X coordinate of the visual aid used for configuration. Does not affect the operation of the detector.
LocationY	int32	Y coordinate of the visual aid used for configuration. Does not affect the operation of the detector.
Masks	List/Array/ int16	Mask defining the working area of the detector (see GeometryPolygons)
ObjectTypes	List/string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "Angle": ...,
    "AngleRange": ...,
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "LocationX": ...,
    "LocationY": ...,
    "Masks":
    {
        "0": [ ..., ..., ... ],
        "1": [ ..., ..., ... ]
    },
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorCreateConfiguration

Initial settings for a new detector instance.

See also: Analytics/CreateDetector

Structure

Parameter	Type	Description
DetectorClass	string	Detector type
DetectorID	guid	Unique ID of the detector instance

Pseudo code

```
{
    "DetectorClass": "...",
    "DetectorID": "{...}"
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorInfo

Collection of properties defining an instance of a detector type.[br] A built-in detector is a special instance that is created by the device the first time it is booted and it cannot be delete by the user.

Structure

Parameter	Type	Description
BuiltIn	bool	Indicates if this is a built-in detector or added by a user
Description	string	Description of the detector instance
DetectorClass	string	Detector type
DetectorClassID	int32	Detector type ID
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of the detector instance
State	string	Current state of the detector
Version	int32	Version of this detector

Pseudo code

```
{
    "BuiltIn": ...,
    "Description": "...",
    "DetectorClass": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "State": "...",
    "Version": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorList

See also: Analytics/GetDetectors

Structure

Parameter	Type	Description
Detectors	DetectorInfo	List of the currently available detector instances
BuiltIn	bool	Indicates if this is a built-in detector or added by a user
Description	string	Description of the detector instance
DetectorClass	string	Detector type
DetectorClassID	int32	Detector type ID
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of the detector instance
State	string	Current state of the detector
Version	int32	Version of this detector

Pseudo code

```
{
    "Detectors":
    {
        "BuiltIn": ...,
        "Description": "...",
        "DetectorClass": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "State": "...",
        "Version": ...
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorRequest

Collection of properties that uniquely identifies a detector instance.

See also: Analytics/DeleteDetector, Analytics/DisableDetector, Analytics/EnableDetector, Analytics/GetDetector, Analytics/GetDetectorState

Structure

Parameter	Type	Description
DetectorID	guid	Unique ID of the detector instance

Pseudo code

```
{
    "DetectorID": "{...}"
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorState

The detector state value

Numeric value	String value	Description
0	dsNotConfigured	Detector is not configured or the current configuration is invalid
1	dsInit	Detector is currently initializing the state machine and loading configuration
2	dsError	Detector is in an erroneous state and cannot operate
3	dsUnableToOperate	The current device environment does not allow normal operation of detector. This state does not require user interaction and the detector will resume operation once impeding factors are resolved.
4	dsNormal	Detector operation is normal
5	dsSignal	Detector raised one or more signals that are still active. Detector operation is normal.
6	dsDisabled	Detector is disabled and does not process data

See also: Analytics/GetDetectorState

Structure

Parameter	Type	Description
State	int32	Numeric id of the current detector state

Pseudo code

```
{
    "State": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

DetectorTypeInfo

Collection of properties defining a detector type. The device won't allow creation of the more that [strong]InstanceLimit[/strong] of one type including the build-in detectors.

Structure

Parameter	Type	Description
DetectorClass	string	Detector type
InstanceCount	int32	Currently available detectory of this type
InstanceLimit	int32	Maximum number of this type allowed on the device
Version	int32	Available version of this detector type

Pseudo code

```
{
    "DetectorClass": "...",
    "InstanceCount": ...,
    "InstanceLimit": ...,
    "Version": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

Event

Descriptor of an event emitted by a detector.

DetectorEventType uses the following values:
- **detSimpleEvent:** Basic event type where the event has no duration.
- **detSignal:** Signals the start of a longer event. The associated detector will also enter signal state until all signalled events are ended.
- **detRestore:** Ends a previously signalled long event. The **EventID** of the start and end events are the same. The associated detector will return to normal state if **all** signals are ended

Restore event types usually don't contain additional information about the previously started event and only serve to mark the end of a detected occurence.

EventCode is a detector specific numeric code to identify what change caused the event. The following are common event codes used by all detectors:
- **2:** Detector finished initialization
- **3:** Detector failed to initialized and stopped working
- **4:** Detector is unable to operate under the current conditions
- **5:** Detector started initializing
- **6:** Detector was created (by user)
- **7:** Detector was destroyed (by user)
- **100:** Generic event code to mark signal/restore event pairs

Event codes above 100 are detector type specific and may overlap.

Events have two timestamps that may have different values based on detector operation. **EventTime** is the timestamp when the detector event was created because all required conditions were met. **EventTriggerTime** may be an earlier timestamp that points to the exact moment the interest of the event happened. For example with tracking detectors where a line is monitored an object crosses a line but it is not yet validated or categorized. The moment of crossing is saved as the original trigger time and when later the object is validated an event is emitted with the current timestamp but with an earlier trigger timestamp.

> Inherited by: EventANPR, EventEmergencyLane, EventForbiddenZone, EventIO, EventLane, EventRedStop, EventStopViolation, EventStoppedObject, EventTest, EventTrafficLine, EventUTurn, EventWhiteLineViolation, EventWrongTurn, EventWrongWay

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventANPR

License plate detection event. The [strong]EventCode[/strong] associated with this event is [strong]114[/strong].

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	EventANPRLicensePlate	May contain detector specific additional information
BackgroundColor	string	Background color of the license plate in #RRGGBB format
CharacterSize	int32	Average character size of the license plate
Confidence	double	Confidence of the detection
Coords	Array/int16	Coordinates of the found license plate's boundaries
Country	string	License plate county code
CountryCode	int32	Numeric license plate country code
DedicatedAreaColor	string	Dedicated area color of the license plate in #RRGGBB format
Direction	string	Estimated direction of the vehicle. Possible values are [strong]Approaching[/strong], [strong]Moving away[/strong] or [strong]Unknown[/strong].
MMR		Make and model recognition results
Text	string	License plate text
TextColor	string	Text color of the license plate in #RRGGBB format
TriggerSource		Properties of the trigger that started the license plate recognition

Parameter	Type	Description
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "BackgroundColor": "...",
        "CharacterSize": ...,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Country": "...",
        "CountryCode": ...,
        "DedicatedAreaColor": "...",
        "Direction": "...",
        "MMR": ,
        "Text": "...",
        "TextColor": "...",
        "TriggerSource":
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition America
Adaptive Recognition Hungary
Adaptive Recognition Nordic
Adaptive Recognition Singapore
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventANPRLicensePlate

License plate properties

Structure

Parameter	Type	Description
BackgroundColor	string	Background color of the license plate in #RRGGBB format
CharacterSize	int32	Average character size of the license plate
Confidence	double	Confidence of the detection
Coords	Array/ int16	Coordinates of the found license plate's boundaries
Country	string	License plate county code
CountryCode	int32	Numeric license plate country code
DedicatedAreaColor	string	Dedicated area color of the license plate in #RRGGBB format
Direction	string	Estimated direction of the vehicle. Possible values are [strong]Approaching[/strong], [strong]Moving away[/strong] or [strong]Unknown[/strong].
MMR		Make and model recognition results
Text	string	License plate text
TextColor	string	Text color of the license plate in #RRGGBB format
TriggerSource		Properties of the trigger that started the license plate recognition

Pseudo code

```
{
    "BackgroundColor": "...",
    "CharacterSize": ...,
    "Confidence": ...,
    "Coords": [ ..., ..., ... ],
    "Country": "...",
    "CountryCode": ...,
    "DedicatedAreaColor": "...",
    "Direction": "...",
    "MMR": ,
    "Text": "...",
    "TextColor": "...",
    "TriggerSource":
}
```

structure

EventEmergencyLane

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	TrackedObjectInfo	Details of the object that entered the emergency lane
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventForbiddenZone

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	TrackedObjectInfo	Details of the object that entered the zone
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventIO

Input port activation event

Structure

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventLane

Structure

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	TrackedObjectInfo	Details of the object that entered the lane
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventRedStop

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	RedStopViolationInfo	Details of the object that ran the red light
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
OrangeTimestamp	int64	Wall clock timestamp in milliseconds when the light entered orange state
RedTimestamp	int64	Wall clock timestamp in milliseconds when the light entered red state
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met

Parameter	Type	Description
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "OrangeTimestamp": ...,
        "RedTimestamp": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventStopViolation

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	TrackedObjectInfo	Details of the object that did not stop for the stop sign
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventStoppedObject

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	TrackedObjectInfo	Details of the object that stopped in the zone
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventTest

Basic test event

Structure

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
Index	int64	A numeric counter that increments when the detector emitted an event of any type
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventTime": ...,
    "EventTriggerTime": ...,
    "Index": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventTrafficLine

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	TrackedObjectInfo	Details of the object that crossed the line
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventUTurn

Structure

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	TrackedObjectInfo	Details of the object that performed an illegal U-turn
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventWhiteLineViolation

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	TrackedObjectInfo	Details of the object that crossed the white line
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventWrongTurn

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	TrackedObjectInfo	Details of the object that turned in the wrong direction
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

EventWrongWay

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventInfo	TrackedObjectInfo	Details of the object that is moving in the wrong direction
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)

Pseudo code

```
{
    "Config":
    {
        "BuiltIn": ...,
        "Class": "...",
        "Description": "...",
        "DetectorClassID": ...,
        "DetectorID": "{...}",
        "DisplayName": "...",
        "Enabled": ...,
        "FpsLimit": ...,
        "RestoreDelayMs": ...,
        "Version": ...,
        "ViolationTimeMs": ...
    },
    "DetectorClassID": ...,
    "DetectorEventType": "...",
    "DetectorID": "{...}",
    "DetectorVersion": ...,
    "EventCode": ...,
    "EventID": "{...}",
    "EventInfo":
    {
        "Center": ,
        "Confidence": ...,
        "Coords": [ ..., ..., ... ],
        "Id": ...,
        "StartTime": ...,
        "State": "...",
        "Type": "..."
    },
    "EventTime": ...,
    "EventTriggerTime": ...,
    "State": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GPSSettings

Structure

Parameter	Type	Description
Latitude	double	Latitude coordinate in decimal degrees
Longitude	double	Longitude coordinate in decimal degrees

Pseudo code

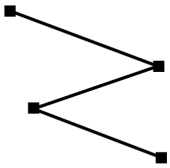
```
{
    "Latitude": ...,
    "Longitude": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GeometryLine

Segmented line with at least one segment, each consisting of a start and end point



Structure

Parameter	Type	Description
Lines	GeometryLineSegment	List of line segments
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point

Pseudo code

```
{
    "Lines":
    {
        "X0": ...,
        "X1": ...,
        "Y0": ...,
        "Y1": ...
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GeometryLineGroup

Segmented line with at least one segment, each consisting of a start and end point and index for sorting.

Structure

Parameter	Type	Description
Lines	GeometryLineSegment	List of line segments
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point
SequenceNumber	int32	Numeric id of this group for ordering

Pseudo code

```
{
    "Lines":
    {
        "X0": ...,
        "X1": ...,
        "Y0": ...,
        "Y1": ...
    },
    "SequenceNumber": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GeometryLineGroups

Groups of segmented lines where an order of groups is formed using indicies

Structure

Parameter	Type	Description
LineGroups	GeometryLineGroup	List of line groups
Lines	GeometryLineSegment	List of line segments
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point
SequenceNumber	int32	Numeric id of this group for ordering

Pseudo code

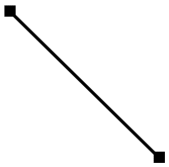
```
{
    "LineGroups":
    {
        "Lines":
        {
            "X0": ...,
            "X1": ...,
            "Y0": ...,
            "Y1": ...
        },
        "SequenceNumber": ...
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GeometryLineSegment

Straight line with two points defining the start and end of the line



Inherited by: IndexedTrackingDetectorLines

Structure

Parameter	Type	Description
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point

Pseudo code

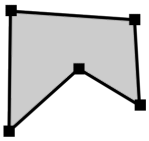
```
{
    "X0": ...,
    "X1": ...,
    "Y0": ...,
    "Y1": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GeometryPolygons

List of polygons. A polygon has at least 3 points with and an arbitrary shape.



Structure

Parameter	Type	Description
Masks	List/Array/ int16	List of masks. Each mask is a list of coordinates where odd and even indicies are x and y coordinates of a corner in the polygon (x0, y0, x1, y1, ...).

Pseudo code

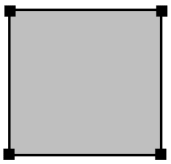
```
{
    "Masks":
    {
        "0": [ ..., ..., ... ],
        "1": [ ..., ..., ... ]
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GeometryRectangle

Rectangle where each side is parallel to the x or y axis of the image



Structure

Parameter	Type	Description
X0	int32	X coordinate of the top left corner
X1	int32	X coordinate of the bottom right corner
Y0	int32	Y coordinate of the top left corner
Y1	int32	Y coordinate of the bottom right

Pseudo code

```
{
    "X0": ...,
    "X1": ...,
    "Y0": ...,
    "Y1": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GpioInputPort

Settings of a digital input port

See also: System/SetGpioInputSettings

Structure

Parameter	Type	Description
ActiveState	bool	State of the port that is considered active/triggered (HIGH/CLOSED = true, LOW/OPEN = false)
Port	string	Unique identifier of a digital input/output port

Pseudo code

```
{
    "ActiveState": ...,
    "Port": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GpioOutputPort

Settings of a digital output port

See also: System/SetGpioOutputSettings

Structure

Parameter	Type	Description
ActiveState	bool	State of the port that is considered active/triggered (HIGH/CLOSED = true, LOW/OPEN = false)
ActiveTime	int32	Duration of the active state after the output is triggered
DetectorList	List/guid	List of detector IDs that can automatically trigger this output with an event
OutputMode	string	Output signal form. Only the "Impulse" mode is supported.
Port	string	Unique identifier of a digital input/output port

Pseudo code

```
{
    "ActiveState": ...,
    "ActiveTime": ...,
    "DetectorList":
    {
        "0": "{...}",
        "1": "{...}"
    },
    "OutputMode": "...",
    "Port": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GpioOutputPortState

Settings for changing the state of a digital output port

> See also: System/SetGpioOutput

Structure

Parameter	Type	Description
Active	bool	New state of the digital output port
Port	string	Unique identifier of a digital input/output port

Pseudo code

```
{
    "Active": ...,
    "Port": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GpioPort

Settings of a digital input/output port

Inherited by: GpioInputPort, GpioOutputPort

See also: System/SetGpioInputSettings, System/SetGpioOutputSettings

Structure

Parameter	Type	Description
ActiveState	bool	State of the port that is considered active/triggered (HIGH/CLOSED = true, LOW/OPEN = false)
Port	string	Unique identifier of a digital input/output port

Pseudo code

```
{
    "ActiveState": ...,
    "Port": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GpioPortId

Inherited by: GpioOutputPortState, GpioPort, GpioPortState

See also: System/SetGpioInputSettings, System/SetGpioOutput, System/SetGpioOutputSettings, System/TriggerGpioOutput

Structure

Parameter	Type	Description
Port	string	Unique identifier of a digital input/output port

Pseudo code

```
{
    "Port": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GpioPortState

State of a digital port

Inherited by: GpioPortStateChange

Structure

Parameter	Type	Description
Active	bool	Current state of the digital port
Port	string	Unique identifier of a digital input/output port
Timestamp	int64	Wall clock timestamp in milliseconds when the digital port changed to this state

Pseudo code

```
{
    "Active": ...,
    "Port": "...",
    "Timestamp": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GpioPortStateChange

Structure

Parameter	Type	Description
Active	bool	Current state of the digital port
Port	string	Unique identifier of a digital input/output port
Timestamp	int64	Wall clock timestamp in milliseconds when the digital port changed to this state
Type	string	Value of "Input" or "Output" indicating the port type

Pseudo code

```
{
    "Active": ...,
    "Port": "...",
    "Timestamp": ...,
    "Type": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GpioSettings

Settings of all digital input/output ports

See also: System/GetGpioSettings

Structure

Parameter	Type	Description
Inputs	GpioInputPort	Settings of available digital input ports. Port name is used as map key.
ActiveState	bool	State of the port that is considered active/triggered (HIGH/CLOSED = true, LOW/OPEN = false)
Port	string	Unique identifier of a digital input/output port
Outputs	GpioOutputPort	Settings of available digital output ports. Port name is used as map key.
ActiveState	bool	State of the port that is considered active/triggered (HIGH/CLOSED = true, LOW/OPEN = false)
ActiveTime	int32	Duration of the active state after the output is triggered
DetectorList	List/guid	List of detector IDs that can automatically trigger this output with an event
OutputMode	string	Output signal form. Only the "Impulse" mode is supported.
Port	string	Unique identifier of a digital input/output port

Pseudo code

```
{
    "Inputs":
    {
        "ActiveState": ...,
        "Port": "..."
    },
    "Outputs":
    {
        "ActiveState": ...,
        "ActiveTime": ...,
        "DetectorList":
        {
            "0": "{...}",
            "1": "{...}"
        },
        "OutputMode": "...",
        "Port": "..."
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

GpioStates

Last known state of all digital input/output ports

See also: System/GetGpioStates

Structure

Parameter	Type	Description
Inputs	GpioPortState	States of available digital input ports. Port name is used as map key.
Active	bool	Current state of the digital port
Port	string	Unique identifier of a digital input/output port
Timestamp	int64	Wall clock timestamp in milliseconds when the digital port changed to this state
Outputs	GpioPortState	States of available digital output ports. Port name is used as map key.
Active	bool	Current state of the digital port
Port	string	Unique identifier of a digital input/output port
Timestamp	int64	Wall clock timestamp in milliseconds when the digital port changed to this state

Pseudo code

```
{
    "Inputs":
    {
        "Active": ...,
        "Port": "...",
        "Timestamp": ...
    },
    "Outputs":
    {
        "Active": ...,
        "Port": "...",
        "Timestamp": ...
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

IndexedTrackingDetectorLines

Structure

Parameter	Type	Description
Id	int8	Index of the line
X0	int32	X coordinate of the start point
X1	int32	X coordinate of the end point
Y0	int32	Y coordinate of the start point
Y1	int32	Y coordinate of the end point

Pseudo code

```
{
    "Id": ...,
    "X0": ...,
    "X1": ...,
    "Y0": ...,
    "Y1": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

LocationSettings

Structure

Parameter	Type	Description
GPS	GPSSettings	Location as GPS coordinates
Latitude	double	Latitude coordinate in decimal degrees
Longitude	double	Longitude coordinate in decimal degrees

Pseudo code

```
{
    "GPS":
    {
        "Latitude": ...,
        "Longitude": ...
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

ModuleAnalytics

Capabilities of the Analytics module. The feature list may contain but not limited to the following values:

Tracker	Supports the iTracking tracker engine (see Analytics/GetTracker)
TrafficDetectors	Supports traffic focused detectors
CarmenEngine	Supports CARMEN license plate recognition (see Analytics/GetAnprEngine)

Structure

Parameter	Type	Description
Features	List/string	List of features available in this module
RequiredCarmenVersion	string	Minimum CARMEN version that can be uploaded to the device

Pseudo code

```
{
    "Features":
    {
        "0": "...",
        "1": "..."
    },
    "RequiredCarmenVersion": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

ModuleIO

Capabilities of the IO module

Structure

Parameter	Type	Description
Inputs	List/string	Names of available input ports
Outputs	List/string	Names of available output ports

Pseudo code

```
{
    "Inputs":
    {
        "0": "...",
        "1": "..."
    },
    "Outputs":
    {
        "0": "...",
        "1": "..."
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

ModuleMedia

Capabilities of the Media module. The feature map contains a list of features for each available sensor. Each feature list may contain but not limited to the following values:

InfraLed	Infrared LED illumination is available
MotorizedFocus	Focus can be adjusted using the motods on the lens
MotorizedZoom	Zoom can be adjusted using the motors on the lens
WDR	Supports wide dynamic range

Structure

Parameter	Type	Description
Features	Map/List/string	List of features available in this module
Sensors	int32	Number of sensors available
Streams	int32	Number of video stream configurations available

Pseudo code

```
{
    "Features":
    {
        "named_key0":
        {
            "0": "...",
            "1": "..."
        },
        "named_key1":
        {
            "0": "...",
            "1": "..."
        }
    },
    "Sensors": ...,
    "Streams": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

NtpSettings

NTP client settings

See also: System/GetNtpSettings, System/SetNtpSettings

Structure

Parameter	Type	Description
Enabled	bool	Enabled state of the device's NTP client
Servers	List/string	List of NTP server addresses or hostnames used when NTP is enabled

Pseudo code

```
{
    "Enabled": ...,
    "Servers":
    {
        "0": "...",
        "1": "..."
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

OptionNumericRange

The numeric range option defines an item's allowed value range from a minimum to a maximum (inclusive). Values outside of the specified range will be ignored as if not sent.

Structure

Parameter	Type	Description
Default	numeric	Default value of the item if not set or the value set is out of range
Maximum	numeric	The maximum value the item accepts
Minimum	numeric	The minimum value the item accepts

Pseudo code

```
{
    "Default": ...,
    "Maximum": ...,
    "Minimum": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

OptionValueList

The value list option defines a limited set of allowed values for an item. A value not present in the list will be ignored as if not sent.

Structure

Parameter	Type	Description
Default	string	Default value of the item if not set
Values	List/string	List of values the item can accept

Pseudo code

```
{
    "Default": "...",
    "Values":
    {
        "0": "...",
        "1": "..."
    }
}
```

structure

RebootSettings

Reboot parameters

> See also: System/Reboot

Structure

Parameter	Type	Description
Message	string	Optional message as the cause of the reboot used for diagnostic purposes

Pseudo code

```
{
    "Message": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

RedStopViolationInfo

Structure

Parameter	Type	Description
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
OrangeTimestamp	int64	Wall clock timestamp in milliseconds when the light entered orange state
RedTimestamp	int64	Wall clock timestamp in milliseconds when the light entered red state
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object

Pseudo code

```
{
    "Center": ,
    "Confidence": ...,
    "Coords": [ ..., ..., ... ],
    "Id": ...,
    "OrangeTimestamp": ...,
    "RedTimestamp": ...,
    "StartTime": ...,
    "State": "...",
    "Type": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

SecurityHistory

List of security related information like blocked sources and active sessions

> See also: System/GetSecurityHistory

Structure

Parameter	Type	Description
BlockedSources	Map/int64	A key/value mapping of blocked sources where the key is the source identifier (usually an IP address) and the value is the duration in milliseconds until the source is unblocked
Sessions	ActiveSession	List of currently active sessions
LastSeen	int64	Elapsed time in milliseconds since the last activity on this session
Source	string	Source of the session, usually an IP address
User	string	The authenticated user name on the session

Pseudo code

```
{
    "BlockedSources":
    {
        "named_key0": ...,
        "named_key1": ...
    },
    "Sessions":
    {
        "LastSeen": ...,
        "Source": "...",
        "User": "..."
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

SecuritySettings

Information required to identify a user account

See also: System/GetSecuritySettings, System/SetSecuritySettings

Structure

Parameter	Type	Description
AuthenticationAttemptLimit	int32	Allowed number of failed authentication attemps before a source is blocked
SourceBlockDuration	int64	Block length in milliseconds

Pseudo code

```
{
    "AuthenticationAttemptLimit": ...,
    "SourceBlockDuration": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

StorageEvents

Result of a stored event query. The parameters of the original query are returned with [strong]StartTime[/strong] and [EndTime]EndTime[/strong] modified to reflect the actual timerange of the result. [br][br] The [strong]Status[/strong] field will contain one of the following values:

- [strong]OK:[/strong] The query returned successfully with at least one event
- [strong]NO_CONTENT:[/strong] The query returned successfully but no events were found that match the criteria
- [strong]PARTIAL_CONTENT[/strong] The query ended successfully but not all events could be returned due to resource constraints

When [strong]PARTIAL_CONTENT[/strong] is returned the device responds with a modified [strong]EndTime[/strong] parameter that is the timestamp of the last event that could successfully be returned in this response. To query the rest of the events perform the same query with [strong]StartTime[/strong] set the the previously returned [strong]EndTime[/strong].

See also: Storage/GetEvents

Structure

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

Parameter	Type	Description
EndTime	int64	Wall clock timestamp in milliseconds of the end of the search range
EventList	Event	List of events that match the search criteria
Config	DetectorConfiguration	Configuration of the detector when this event was created
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Class	string	Detector type name
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.
DetectorClassID	int32	Type ID of the detector
DetectorEventType	string	Type of this event
DetectorID	guid	Unique ID of the detector
DetectorVersion	int32	Version of the detector
EventCode	int32	Detector specific event code
EventID	guid	Unique ID of the event
EventTime	int64	Wall clock timestamp in milliseconds of the event creation
EventTriggerTime	int64	Wall clock timestamp in milliseconds when the conditions for the event were met
State	string	State of the detector after the event was emitted (see DetectorState)
Filter	StorageEventsRequestFilter	(optional) Additional filter parameters
FuzzySearch	bool	Set to true to allow fuzzy search that includes not only exact matches but similiar matches too where one character may be different
Params	string	(optional) Comma separated list of key:value pairs
Pattern	string	String pattern to match for. May use placeholders to match any characters. A question mark (?) indicates one character, an asterisk (*) indicates zero or more.
ID	guid	(optional) Unique ID of the detector to search for

Parameter	Type	Description
StartTime	int64	Wall clock timestamp in milliseconds of the beggining of the search range
Status	string	Final status of the query

Pseudo code

```
{
    "EndTime": ...,
    "EventList":
    {
        "Config":
        {
            "BuiltIn": ...,
            "Class": "...",
            "Description": "...",
            "DetectorClassID": ...,
            "DetectorID": "{...}",
            "DisplayName": "...",
            "Enabled": ...,
            "FpsLimit": ...,
            "RestoreDelayMs": ...,
            "Version": ...,
            "ViolationTimeMs": ...
        },
        "DetectorClassID": ...,
        "DetectorEventType": "...",
        "DetectorID": "{...}",
        "DetectorVersion": ...,
        "EventCode": ...,
        "EventID": "{...}",
        "EventTime": ...,
        "EventTriggerTime": ...,
        "State": "..."
    },
    "Filter":
    {
        "FuzzySearch": ...,
        "Params": "...",
        "Pattern": "..."
    },
    "ID": "{...}",
    "StartTime": ...,
    "Status": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

StorageEventsRequest

Search parameters for a stored event query

Inherited by: StorageEvents

See also: Storage/GetEvents

Structure

Parameter	Type	Description
EndTime	int64	Wall clock timestamp in milliseconds of the end of the search range
Filter	StorageEventsRequestFilter	(optional) Additional filter parameters
FuzzySearch	bool	Set to true to allow fuzzy search that includes not only exact matches but similiar matches too where one character may be different
Params	string	(optional) Comma separated list of key:value pairs
Pattern	string	String pattern to match for. May use placeholders to match any characters. A question mark (?) indicates one character, an asterisk (*) indicates zero or more.
ID	guid	(optional) Unique ID of the detector to search for
StartTime	int64	Wall clock timestamp in milliseconds of the beggining of the search range

Pseudo code

```
{
    "EndTime": ...,
    "Filter":
    {
        "FuzzySearch": ...,
        "Params": "...",
        "Pattern": "..."
    },
    "ID": "{...}",
    "StartTime": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

StorageEventsRequestFilter

Additional search parameters for a stored event query. [br] [br] [strong]Pattern[/strong] is used to filter out events whose metadata does not match the pattern.[br] [strong]Params[/strong] can be used to specify modifiers for the search. As of now only "country" is supported (e.g.: "country:NOR" to search for license plates from Norway).[br] [br] Currently only ANPR events have metadata in the form of license plate strings and country codes.

Structure

Parameter	Type	Description
FuzzySearch	bool	Set to true to allow fuzzy search that includes not only exact matches but similiar matches too where one character may be different
Params	string	(optional) Comma separated list of key:value pairs
Pattern	string	String pattern to match for. May use placeholders to match any characters. A question mark (?) indicates one character, an asterisk (*) indicates zero or more.

Pseudo code

```
{
    "FuzzySearch": ...,
    "Params": "...",
    "Pattern": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

StorageStatistics

General statistics from the storage subsystem

See also: Storage/GetStatistics

Structure

Parameter	Type	Description
EndTime	int64	Wall clock timestamp in milliseconds of the newest available data on the storage device
InUse	int64	Number of bytes in used on the used storage device
StartTime	int64	Wall clock timestamp in milliseconds of the oldest available data on the storage device
Total	int64	Total number of bytes available on the used storage device

Pseudo code

```
{
    "EndTime": ...,
    "InUse": ...,
    "StartTime": ...,
    "Total": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

SupportedDetectors

See also: Analytics/GetSupportedDetectors

Structure

Parameter	Type	Description
DetectorTypes	DetectorTypeInfo	List of supported detector types
DetectorClass	string	Detector type
InstanceCount	int32	Currently available detectory of this type
InstanceLimit	int32	Maximum number of this type allowed on the device
Version	int32	Available version of this detector type

Pseudo code

```
{
    "DetectorTypes":
    {
        "DetectorClass": "...",
        "InstanceCount": ...,
        "InstanceLimit": ...,
        "Version": ...
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

SystemSettings

Inherited by: SystemSettingsResponse

See also: System/GetDevice, System/SetDevice

Structure

Parameter	Type	Description
Description	string	User-specified description
Location	LocationSettings	User-specified location
GPS	GPSSettings	Location as GPS coordinates
Latitude	double	Latitude coordinate in decimal degrees
Longitude	double	Longitude coordinate in decimal degrees
Name	string	User-specified name

Pseudo code

```
{
    "Description": "...",
    "Location":
    {
        "GPS":
        {
            "Latitude": ...,
            "Longitude": ...
        }
    },
    "Name": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

SystemSettingsDevice

Structure

Parameter	Type	Description
Description	string	Additional information about the product
FirmwareVersion	string	Firmware version in x.x.x.x format
ProductClass	string	Class name of the product lineup with similiar features
ProductDisplayName	string	Human-readable name of the product design. May be the same as ProductName.
ProductName	string	Name of the product design
ProductSubclass	string	Subclass of the lineup identifying a specific use-case
RequiredFirmwareVersion	string	Minimum firmware version in x.x.x.x format that this device accepts when a new firmware is uploaded
Serial	string	Unique device serial number

Pseudo code

```
{
    "Description": "...",
    "FirmwareVersion": "...",
    "ProductClass": "...",
    "ProductDisplayName": "...",
    "ProductName": "...",
    "ProductSubclass": "...",
    "RequiredFirmwareVersion": "...",
    "Serial": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

SystemSettingsModule

Inherited by: ModuleAnalytics, ModuleIO, ModuleMedia

Structure

Parameter	Type	Description

Pseudo code

```
{

}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

SystemSettingsResponse

See also: System/GetDevice

Structure

Parameter	Type	Description
Description	string	User-specified description
Device	SystemSettingsDevice	General system properties
Description	string	Additional information about the product
FirmwareVersion	string	Firmware version in x.x.x.x format
ProductClass	string	Class name of the product lineup with similiar features
ProductDisplayName	string	Human-readable name of the product design. May be the same as ProductName.
ProductName	string	Name of the product design
ProductSubclass	string	Subclass of the lineup identifying a specific use-case
RequiredFirmwareVersion	string	Minimum firmware version in x.x.x.x format that this device accepts when a new firmware is uploaded
Serial	string	Unique device serial number
InstanceId	int64	Unique ID that changes every time the system restarts
Location	LocationSettings	User-specified location
GPS	GPSSettings	Location as GPS coordinates
Latitude	double	Latitude coordinate in decimal degrees
Longitude	double	Longitude coordinate in decimal degrees
Modules	SystemSettingsModule	List of module specific entries that describe each module's capabilities
Name	string	User-specified name
Uptime	int64	Elapsed milliseconds since the system started

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "Description": "...",
    "Device":
    {
        "Description": "...",
        "FirmwareVersion": "...",
        "ProductClass": "...",
        "ProductDisplayName": "...",
        "ProductName": "...",
        "ProductSubclass": "...",
        "RequiredFirmwareVersion": "...",
        "Serial": "..."
    },
    "InstanceId": ...,
    "Location":
    {
        "GPS":
        {
            "Latitude": ...,
            "Longitude": ...
        }
    },
    "Modules":
    {

    },
    "Name": "...",
    "Uptime": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

TestInput

Configure the response given to the System/RunTest method. The **Text** may be set to anything or left empty. Using the **ThrowException** field, one can control the type of response the **RunTest** command may return.

- If this is false the response will be success (given no other higher level errors occur) and a TestOutput object will be returned.
- If this is true the response will be an error of a **TextException** type.

> See also: System/RunTest

Structure

Parameter	Type	Description
Text	string	Arbitrary test input that the System/RunTest will return if no exceptions are thrown
ThrowException	bool	If this field is set to true the response to System/RunTest will be an exception

Pseudo code

```
{
    "Text": "...",
    "ThrowException": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

TestOutput

Response to a successful System/RunTest method call.

> See also: System/RunTest

Structure

Parameter	Type	Description
Size	int32	Length of the original input text in bytes
Text	string	The original input text preceeded with the "Input recieved: " string
User	string	Name of the user executing the command

Pseudo code

```
{
    "Size": ...,
    "Text": "...",
    "User": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

TimeSettings

Device time settings

> See also: System/GetTime, System/SetTime

Structure

Parameter	Type	Description
Timestamp	int64	Current wall clock timestamp on the device (UTC)

Pseudo code

```
{
    "Timestamp": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

TrackedObjectInfo

Inherited by: RedStopViolationInfo

Structure

Parameter	Type	Description
Center		
Confidence	double	Confidence of object tracking and categorization on a scale of 0 to 1
Coords	Array/int16	Coordinate pairs of the object's bounding box (x0,y0,x1,y1,...)
Id	int64	Unique id of the tracked object
StartTime	int64	Wall clock timestamp in milliseconds of the moment the object first appeared
State	string	State of object when the event was created
Type	string	Type of object

Pseudo code

```
{
    "Center": ,
    "Confidence": ...,
    "Coords": [ ..., ..., ... ],
    "Id": ...,
    "StartTime": ...,
    "State": "...",
    "Type": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

TrackerConfiguration

Configuration of the iTracking engine.
The engine operates inside the configured mask or the whole image if none specified. Moving objects are tracked and categorized and sent to track based detectors for further analysis.

See also: Analytics/GetTracker, Analytics/GetTrackerDefaults, Analytics/SetTracker

Structure

Parameter	Type	Description
Config		

Pseudo code

```
{
    "Config":
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

TrackingDetectorConfiguration

Inherited by: DetectorConfigurationEmergencyLane, DetectorConfigurationForbiddenZone, DetectorConfigurationLane, DetectorConfigurationRedStop, DetectorConfigurationStopViolation, DetectorConfigurationStoppedObject, DetectorConfigurationTrafficLine, DetectorConfigurationUTurn, DetectorConfigurationWhiteLineViolation, DetectorConfigurationWrongTurn, DetectorConfigurationWrongWay

Structure

Parameter	Type	Description
BuiltIn	bool	Automatically created detectors are marked built-in and cannot be deleted
Center	bool	Set to true to operate using an object's center point instead of all corners
Class	string	Detector type name
Confidence	int8	Minimum allowed object confidence when [strong]ConfidenceEnabled[/strong] is set to true
ConfidenceEnabled	bool	Set to true to use a confidence treshold for object monitoring
Description	string	Description of this detector instance for easier identification
DetectorClassID	int32	Detector type code
DetectorID	guid	Unique ID of the detector instance
DisplayName	string	Name of this detector instance displayed on user-facing interfaces
Enabled	bool	Controls the enabled state of the detector
FpsLimit	double	Limits the run speed of the detector to a specific FPS. Set to zero for no limit.
ObjectTypes	List/ string	List of object types that are monitored or empty list for all types
RestoreDelayMs	int64	[em]unused[/em]
Version	int32	Detector type version
ViolationTimeMs	int64	Violations have to be present for this duration before an event is emitted. Not all detectors may use this field.

Pseudo code

Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

```
{
    "BuiltIn": ...,
    "Center": ...,
    "Class": "...",
    "Confidence": ...,
    "ConfidenceEnabled": ...,
    "Description": "...",
    "DetectorClassID": ...,
    "DetectorID": "{...}",
    "DisplayName": "...",
    "Enabled": ...,
    "FpsLimit": ...,
    "ObjectTypes":
    {
        "0": "...",
        "1": "..."
    },
    "RestoreDelayMs": ...,
    "Version": ...,
    "ViolationTimeMs": ...
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

User

All user account information

See also: System/AddUser, System/ModifyUser

Structure

Parameter	Type	Description
Name	string	User name
Password	string	User password (write only)
Role	string	User role

Pseudo code

```
{
    "Name": "...",
    "Password": "...",
    "Role": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

UserId

Information required to identify a user account

Inherited by: UserInfo

See also: System/AddUser, System/DeleteUser, System/GetCurrentUser, System/ModifyUser

Structure

Parameter	Type	Description
Name	string	User name

Pseudo code

```
{
    "Name": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

UserInfo

User account information

Inherited by: User

See also: System/AddUser, System/GetCurrentUser, System/ModifyUser

Structure

Parameter	Type	Description
Name	string	User name
Role	string	User role

Pseudo code

```
{
    "Name": "...",
    "Role": "..."
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM

structure

Users

Contains information about all user accounts available on the device

See also: System/GetUsers

Structure

Parameter	Type	Description
Users	User	List of user accounts
Name	string	User name
Password	string	User password (write only)
Role	string	User role

Pseudo code

```
{
    "Users":
    {
        "Name": "...",
        "Password": "...",
        "Role": "..."
    }
}
```


Adaptive Recognition **America**
Adaptive Recognition **Hungary**
Adaptive Recognition **Nordic**
Adaptive Recognition **Singapore**
REQUESTINFO@ADAPTIVERECOGNITION.COM
WWW.ADAPTIVERECOGNITION.COM