MAKE THE
WORLD SEE

Date: 29 June 2021

Application Note

Metadata Search integration

milestone

Metadata Search integration

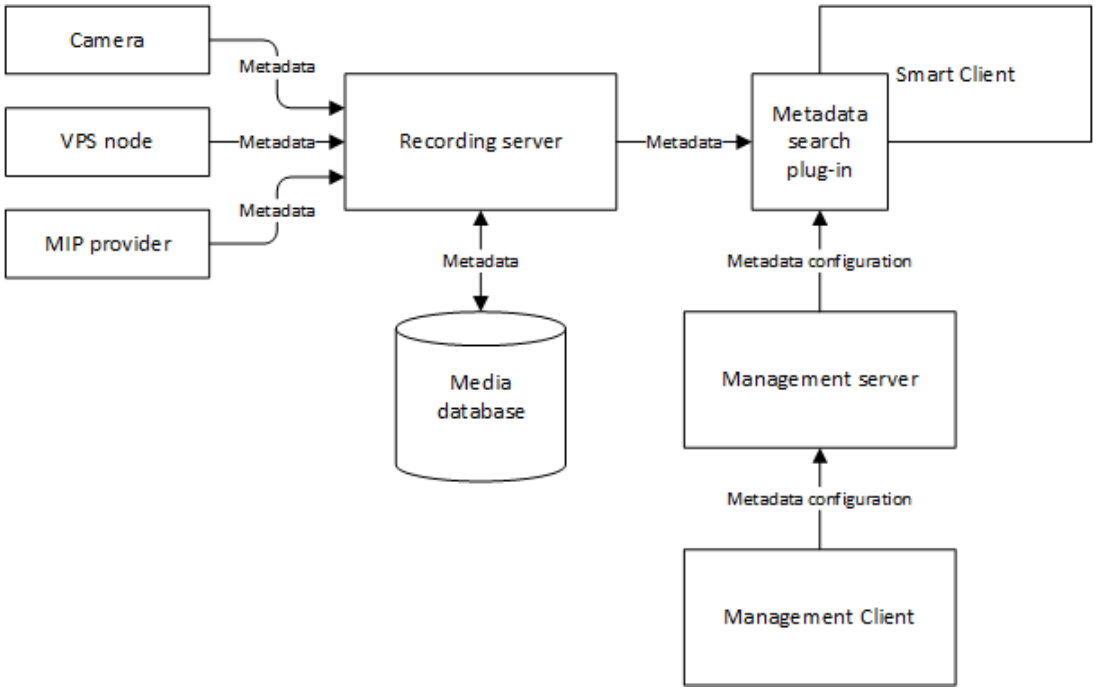
This document is intended to enable Milestone technology partners and Milestone developers to create metadata that will be recognized and available for the Metadata search feature.

Some familiarity with the ONVIF Analytics Service Specification and with XProtect is assumed.

How XProtect VMS works with metadata

Metadata search architecture

This diagram describes the high-level overview of the involved components as well as the flow of metadata.



Metadata can originate from different sources: a camera, a MIP provider, or a VPS node. The recording servers receives the metadata and stores it in media databases. A benefit of storing the metadata alongside the video is that the metadata is exported and deleted along with the video. This simplifies management of the metadata.

In the XProtect Smart Client, a Metadata search plug-in allows the user to define search criteria. The plug-in pulls and filters metadata from the recording servers, and presents the intervals that match the search criteria.

The system needs an understanding of which categories of objects (e.g. people and vehicles) are to available for search. This is configured using the Management Client and stored on the management server.

Metadata search in XProtect Smart Client

Metadata-based search is integrated in the Smart Client along other categories of search filters, such as alarms, events, motion detection, and bookmarks.

You can think of search categories as a grouping of filters for properties related to video recordings, the frames in the recordings, and objects detected in the frames.

A few examples of how metadata is presented in the Smart Client search feature:

- Objects that are recognized as the class `People` can be categorized by the properties `Age`, `Gender`, `Height`, and whether a `Face` was detected. These properties can be used in search filters for the search category `People`.
- An object can have a `Location`. In addition, the recording also can have a `Location` (the position of the camera itself). Thus, `Location` is a search category by itself, with a filter for geo coordinates (latitude, longitude) and search radius.

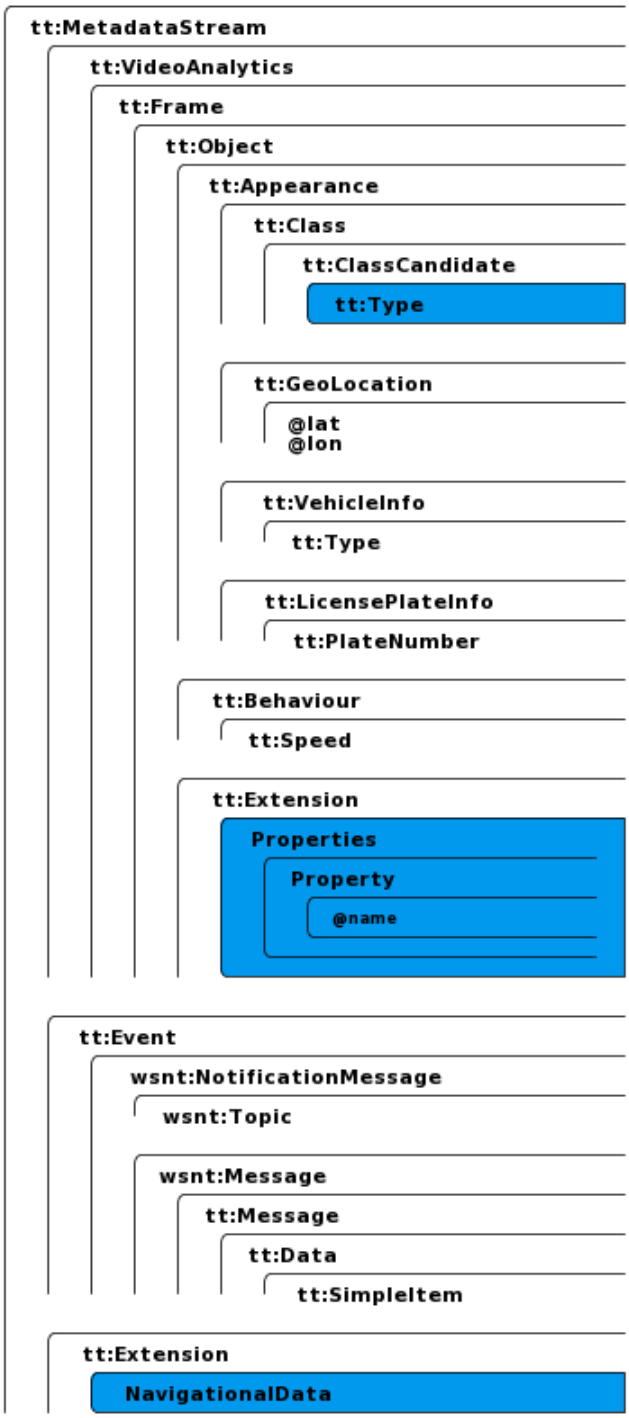
Format

The ONVIF MetadataStream schema

Metadata support in XProtect is based on the ONVIF Analytics Service Specification Version 19.12.

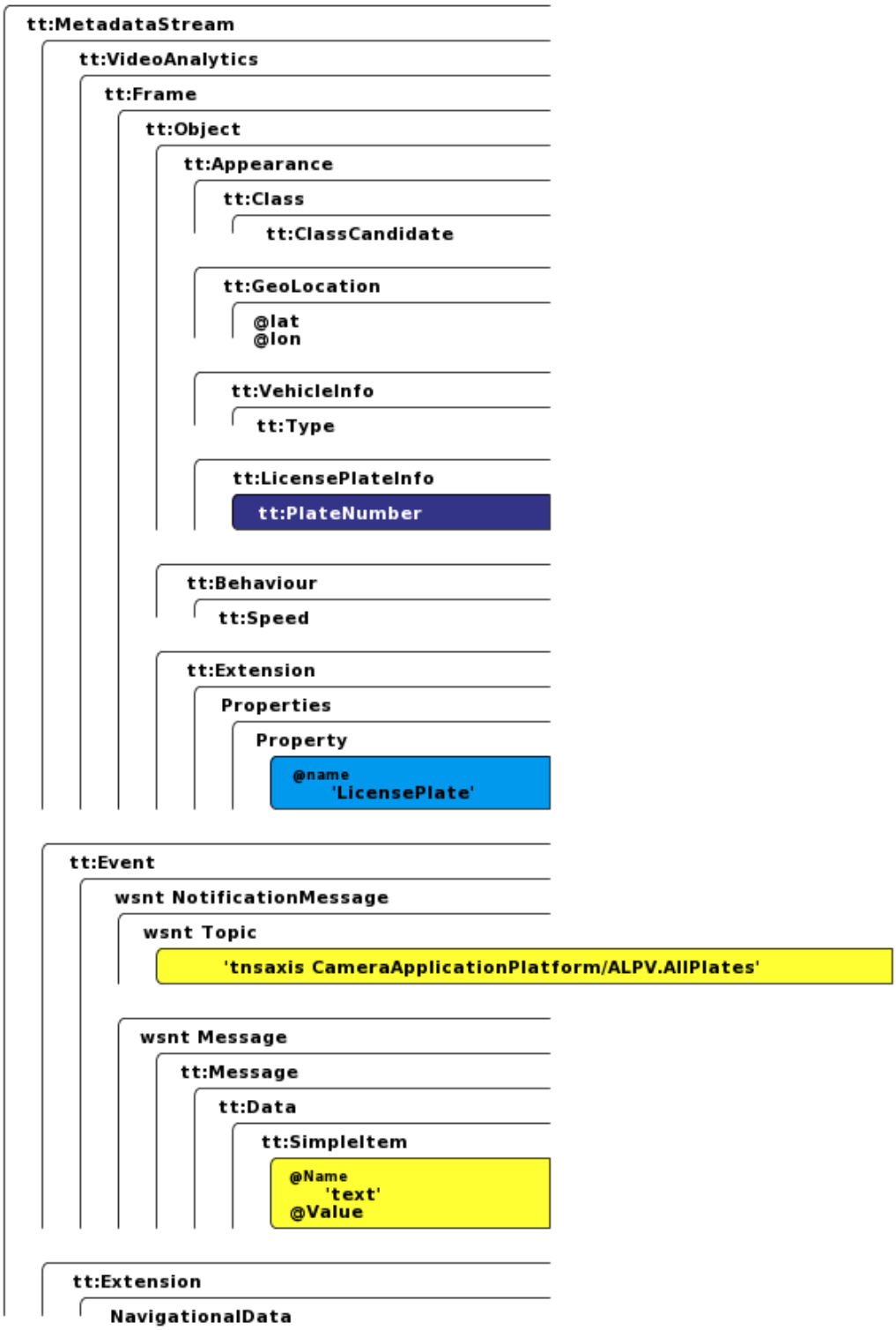
ONVIF schemas are designed to be backwards compatible, forwards compatible, and extensible.

For Metadata search, XProtect extends the ONVIF MetadataStream schema in several places, highlighted in light blue:



At any point in time, there might be several ways to represent the same information, e.g. legacy ONVIF schemas, the current ONVIF schema, and several vendor specific extensions.

For example, for Metadata search, license plate information is found in three places: the default ONVIF representation (dark blue), an XProtect extension (light blue), and another vendor (Axis) extension (yellow).



To accommodate for this variability, XProtect maps ONVIF object classes and object descriptors to XProtect search categories and properties.

The mapping of object classes to search categories is described below in the section Search properties and categories.

ONVIF MetadataStream elements and properties supported by XProtect

Namespace binding

The following prefixes are used:

Prefix	Namespace
`tt`	`http://www.onvif.org/ver10/schema`
`wsnt`	`http://docs.oasis-open.org/wsn/b-2`

MetadataStream elements not used for Metadata search

XProtect supports ONVIF metadata for several purposes, not just for Metadata search.

Some MetadataStream elements are not available for Metadata search, but can be used e.g. for overlay display in the Smart Client:

- `/tt:MetadataStream/tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/`
  - `tt:Shape/tt:BoundingBox`
  - `tt:Shape/tt:Extension/BoundingBoxAppearance`
  - `tt:Extension/Description`

For more information about these, see Introduction to Metadata.

MetadataStream elements used by Metadata search

Object metadata are extracted from several root XPaths in each `tt:MetadataStream` element. Each of these roots contribute independently to Metadata search.

This table describes the element and attribute values extracted from each root XPath.

Root XPath	Element or attribute XPath	Search property
/tt:MetadataStream/tt:VideoAnalytics/tt:Frame/tt:Object/		
	tt:Appearance/tt:Class/tt:ClassCandidate/tt:Type	Class
	tt:Appearance/tt:Class/tt:ClassCandidate/tt:Type[text()='Face']	Face
	tt:Appearance/tt:GeoLocation/	
	@lat	Latitude
	@lon	Longitude
	tt:Appearance/tt:LicensePlateInfo/tt:PlateNumber	LicensePlate
	tt:Appearance/tt:VehicleInfo/tt:Type	VehicleType
	tt:Behaviour/tt:Speed	Speed
	tt:Extension/Properties/	
	Property[@name = "Age"]	Age
	Property[@name = "Gender"]	Gender
	Property[@name = "Height"]	Height
	Property[@name = "Color"]	VehicleColor
	Property[@name = "Class"]	Class
	Property[@name = "LicensePlate"]	LicensePlate
/tt:MetadataStream/tt:Extension/NavigationalData/		
	Latitude	Latitude
	Longitude	Longitude
/tt:MetadataStream/tt:Event/wsnt:NotificationMessage/wsnt:Topic[text() = 'tnsaxis:CameraApplicationPlatform/ALPV.AllPlates']/../wsnt:Message/tt:Message/tt:Data/		
	tt:SimpleItem[@Name='text']/@Value	LicensePlate

Search properties and categories

This table describes the mapping of class values and properties to the Smart Client Metadata search categories.

Properties that are based on extensions or alternatives to ONVIF are described in Extensions and alternatives to the ONVIF Analytics Service Specification.

Search property	Values	Search category	ONVIF specification	
Class			Analytics	5.1.3.5 Object Class descriptor
	`Face`, `Human`, `Person`, `People`	People		
	`Vehical`, `Vehicle`	Vehicles		
Face	boolean	People	Analytics	5.1.3.5 Object Class descriptor
Age	integer [y]	People	Analytics	5.1.3.11 Face descriptor
Gender	`Female`, `Male`	People	Analytics	5.1.3.11 Face descriptor
Height	decimal [m]	People	Analytics	5.1.3.12 Human body descriptor
VehicleColor	RGB, ARGB, HTML (KnownColor) names	Vehicles	Analytics	5.1.3.4 Colour descriptor
VehicleType	`Bicycle`, `Bus`, `Car`, `Motorcycle`, `Truck`	Vehicles	Analytics	5.1.3.7 Vehicle information descriptor
Speed	decimal [m/s]	Vehicles	Analytics	5.1.3.8 Speed descriptor
LicensePlate	string	Vehicles	Analytics	5.1.3.9 License plate information descriptor
Latitude	decimal ±90 [DD]	Location	Core	4.7 Geo Location
Longitude	decimal ±180 [DD]	Location	Core	4.7 Geo Location

To make metadata from an object available in Metadata search categories, both class values and search properties are considered. Currently, the algorithm is as follows:

- If no class value is available for an object, all class specific object properties are used to to infer search categories. For example, if no class value but both `Age` and `Speed` are available for an object, you can get hits for this object in both the `People` and the `Vehicles` search category.
- If any recognized class values are available for an object, you can get hits for this object in the corresponding search categories, and properties not related to those object classes are not considered. For example, if `Class="Vehical"` (and no other class values) and `Gender="Male"` are available for an object, you can only get hits for this object in the `Vehicles` search category.
- If an object has only unrecognized class values, properties are not considered, and you will not get hits for this object in any search category.

The properties `Latitude` and `Longitude` are not related to any specific object class.

> **Note**
>
> As a consequence, if any object class value (even unrecognized) is available, you cannot get hits for this object in the `Location` search category.

But, for the purpose of Metadata search, the `/tt:MetadataStream` element itself is also considered an object, and you can get hits in the `Location` search category if `/tt:MetadataStream/tt:Extension/NavigationalData/Latitude` or `/tt:MetadataStream/tt:Extension/NavigationalData/Longitude` are available.

Extensions and alternatives to the ONVIF Analytics Service Specification

Compared to the ONVIF Analytics Service Specification, XProtect Metadata search extends or supports alternatives for several object descriptors.

ONVIF object descriptor	Search properties
Class	Class
Color	VehicleColor
GeoLocation	Latitude, Longitude
LicensePlate	LicensePlate
HumanFace and HumanBody	Face, Age, Gender, and Height

Each extension is described in the following sections.

Class

XProtect extends the ONVIF Class descriptor as follows:

- Any `tt:Class/tt:ClassCandidate/tt:Type` value is allowed (in addition to the enumeration in the ONVIF Analytics Service Specification)
- `tt:Class/tt:Extension/tt:OtherTypes` is not supported
- `tt:Class/tt:Type` is not supported
- `tt:Class/tt:ClassCandidate/tt:Likelihood` is ignored
- `/tt:MetadataStream/tt:VideoAnalytics/tt:Frame/tt:Object/tt:Extension/Properties/Property[@name = "Class"]` also provides object class values

The following Class Type aliases are recognized for Metadata search:

`tt:Type`	Aliases	Search category
`Face`		`People`
`Human`	`Person`, `People`	`People`
`Vehical`	`Vehicle`	`Vehicles`

`Face` objects are mapped to the search category `People`, and `Face` is also regarded a property of the `People` category.

Color

The ONVIF Analytics Service Specification offers a rich and somewhat complicated representation of object colors (ONVIF Analytics Service Specification, 5.1.3.4 Colour descriptor).

In place of the ONVIF color representation, XProtect uses an extension to support a simpler representation of object color:

```
<?xml version="1.0" encoding="UTF-8"?>
<tt:MetadataStream xmlns:tt="http://www.onvif.org/ver10/schema">
  <tt:VideoAnalytics>
    <tt:Frame UtcTime="2018-01-02T09:28:09.813488Z">
      <tt:Object ObjectId="13">
        ...
        <tt:Extension>
          <Properties>
            <Property name="Color">#564565</Property>
          </Properties>
        </tt:Extension>
      </tt:Object>
    </tt:Frame>
  </tt:VideoAnalytics>
</tt:MetadataStream>
```

The color format and values are those recognized by .NET `System.Drawing.ColorTranslator.FromHtml()`:

- "#rrggbb": 6 hex digits (red, green, blue), for example `#564565`.
- "#aarrggbb": 8 hex digits (alpha, red, green, blue), for example `#ff564565`.
- .NET `System.Drawing.KnownColor` names, based on CSS color keywords. While the CSS color keywords include "grey" aliases for "gray" colors, only the 'gray' names are supported by `System.Drawing.ColorTranslator.FromHtml()`.

`Color` is mapped to the search property `VehicleColor`, related to the search category `Vehicles`.

In the Smart Client, color values are mapped to closest CSS color in this list: `Beige`, `Black`, `Blue`, `Green`, `Grey`, `Red`, `White`, `Yellow`.

GeoLocation

In addition to the object location introduced in the ONVIF 18.06 release,

```
<?xml version="1.0" encoding="UTF-8"?>
<tt:MetadataStream xmlns:tt="http://www.onvif.org/ver10/schema">
  <tt:VideoAnalytics>
    <tt:Frame UtcTime="2018-01-02T09:28:09.813488Z">
      <tt:Object ObjectId="13">
        <tt:Appearance>
          ..
          <tt:GeoLocation lat="16.0735273296356" lon="-27.2426441904356" />
        </tt:Appearance>
        ...
      </tt:Object>
    </tt:Frame>
  </tt:VideoAnalytics>
</tt:MetadataStream>
```

XProtect has supported navigational data related to the metadata stream (i.e., the camera itself) since 2014:

```
<?xml version="1.0" encoding="utf-8"?>
<tt:MetadataStream xmlns:tt="http://www.onvif.org/ver10/schema">
  <tt:VideoAnalytics>
    <tt:Frame UtcTime="2018-01-02T09:28:09.813488Z">
    ...
  </tt:VideoAnalytics>
  <tt:Extension>
    <NavigationalData version="1.0">
      <Latitude>52.069926</Latitude>
      <Longitude>11.796875</Longitude>
      ...
    </NavigationalData>
  </tt:Extension>
</tt:MetadataStream>
```

For Metadata search, the location values are expected to use same notation and geodetic reference as ONVIF: decimal degrees, WGS84.

To support backwards and forwards compatibility, ONVIF applies a schema version and extension policy. For a number of MetadataStream extension points, the ONVIF schema requires vendor element extensions to be declared in a separate vendor specific namespace. `/tt:MetadataStream/tt:Extension` is one of the extension points in which content elements is now required to be in a vendor namespace. But, unless you enforce schema validation, this is very unlikely to cause problems.

LicensePlate

Three different XPaths contribute `LicensePlate` values:

- The ONVIF `tt:LicensePlateInfo` element
- An XProtect extension
- An Axis extension (Axis ALPV)

ONVIF license plate example:

```
<?xml version="1.0" encoding="UTF-8"?>
<tt:MetadataStream xmlns:tt="http://www.onvif.org/ver10/schema">
  <tt:VideoAnalytics>
    <tt:Frame UtcTime="2018-01-02T09:28:09.813488Z">
      <tt:Object ObjectId="13">
        <tt:Appearance>
          ...
          <tt:LicensePlateInfo>
            <tt:PlateNumber>AB12345</tt:PlateNumber>
          </tt:LicensePlateInfo>
        </tt:Appearance>
        ...
      </tt:Object>
    </tt:Frame>
  </tt:VideoAnalytics>
</tt:MetadataStream>
```

XProtect license plate example:

```
<?xml version="1.0" encoding="UTF-8"?>
<tt:MetadataStream xmlns:tt="http://www.onvif.org/ver10/schema">
  <tt:VideoAnalytics>
```

```
    <tt:Frame UtcTime="2018-01-02T09:28:09.813488Z">
      <tt:Object ObjectId="13">
        ...
        <tt:Extension>
          <Properties>
            ...
            <Property name="LicensePlate">AB12345</Property>
          </Properties>
        </tt:Extension>
      </tt:Object>
    </tt:Frame>
  </tt:VideoAnalytics>
</tt:MetadataStream>
```

Axis ALPV license plate example:

```
<?xml version="1.0" encoding="UTF-8"?>
<tt:MetadataStream xmlns:tt="http://www.onvif.org/ver10/schema">
  <tt:Event>
    <wsnt:NotificationMessage
      xmlns:tns1="http://www.onvif.org/ver10/topics"
      xmlns:tnsaxis="http://www.axis.com/2009/event/topics"
      xmlns:wsnt="http://docs.oasis-open.org/wsn/b-2"
      xmlns:wsa5="http://www.w3.org/2005/08/addressing">
      <wsnt:Topic Dialect="http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple
      <wsnt:ProducerReference>
        <wsa5:Address>uri://face2652-97c8-4d13-a4ce-723baa9d6c2c/ProducerReference</
      </wsnt:ProducerReference>
      <wsnt:Message>
        <tt:Message UtcTime="2019-11-29T09:51:08.524672Z">
          <tt:Source/>
          <tt:Data>
            ...
            <tt:SimpleItem Name="text" Value="AB12345"/>
            ...
          </tt:Data>
        </tt:Message>
      </wsnt:Message>
    </wsnt:NotificationMessage>
  </tt:Event>
</tt:MetadataStream>
```

HumanFace and HumanBody

The ONVIF 19.12 release introduced `tt:Object/tt:Appearance/` elements for description of human body and face. While these elements allow description in great detail, they also add considerable complexity, and currently, few devices support these element.

In place of the ONVIF `HumanFace` and `HumanBody` object descriptors, XProtect supports properties for describing a human object through an extension.

Also, in addition to the ONVIF object class `Human`, XProtect recognizes `Person` and `People` as aliases for `Human`.

Human properties example:

```
<?xml version="1.0" encoding="UTF-8"?>
<tt:MetadataStream xmlns:tt="http://www.onvif.org/ver10/schema">
  <tt:VideoAnalytics>
    <tt:Frame UtcTime="2018-01-02T09:28:09.813488Z">
      <tt:Object ObjectId="12">
          ...
          <Properties>
            <Property name="Gender">Male</Property>
            <Property name="Height">1.81</Property>
            <Property name="Age">48</Property>
          </Properties>
        </tt:Extension>
      </tt:Object>
    </tt:Frame>
  </tt:VideoAnalytics>
</tt:MetadataStream>
```

Examples

A few complete examples illustrate the use of ONVIF metadata for Metadata search.

Vehicle example

This example includes the use of these extensions:

- A non-spec class type value accepted in `tt:Class/tt:ClassCandidate/tt:Type`; `tt:Class/tt:Extension/tt:OtherTypes` is not used
- A simple representation of object color

```
<?xml version="1.0"?>
<tt:MetadataStream xmlns:tt="http://www.onvif.org/ver10/schema">
  <tt:VideoAnalytics>
    <tt:Frame UtcTime="2018-01-02T09:28:09.813488Z">
      <tt:Object ObjectId="13">
        <tt:Appearance>
          <tt:Shape><!-- Not used for Metadata search  -->
            <tt:BoundingBox bottom="20.0" top="30.0" right="100.0" left="80.0" />
            <tt:CenterOfGravity x="60.0" y="50.0" />
          </tt:Shape>
          <tt:Class>
            <tt:ClassCandidate>
              <tt:Type>Vehicle</tt:Type><!-- 'Vehicle' accepted as an alias for 'Veh
              <tt:Likelihood>1.0</tt:Likelihood>
            </tt:ClassCandidate>
          </tt:Class>
          <tt:GeoLocation lat="16.0735273296356" lon="-27.2426441904356" />
          <tt:VehicleInfo>
            <tt:Type>Car</tt:Type>
          </tt:VehicleInfo>
          <tt:LicensePlateInfo>
            <tt:PlateNumber>AB12345</tt:PlateNumber>
          </tt:LicensePlateInfo>
        </tt:Appearance>
        <tt:Behaviour>
          <tt:Speed>56</tt:Speed>
```

```
        </tt:Behaviour>
        <tt:Extension>
          <Properties>
            <Property name="Color">#564565</Property><!-- Object color extension -->
          </Properties>
        </tt:Extension>
      </tt:Object>
    </tt:Frame>
  </tt:VideoAnalytics>
</tt:MetadataStream>
```

Human and Face example

This example includes the use of these extensions:

- `tt:Likelihood` is ignored, so the sum does not need to be equal to or less than 1.0
- A simple representation of human properties
- Geo location related to the camera

```
<?xml version="1.0"?>
<tt:MetadataStream xmlns:tt="http://www.onvif.org/ver10/schema">
  <tt:VideoAnalytics>
    <tt:Frame UtcTime="2018-01-02T09:28:09.813488Z">
      <tt:Object ObjectId="12">
        <tt:Appearance>
          <tt:Shape><!-- Not used for Metadata search  -->
            <tt:BoundingBox bottom="20.0" top="30.0" right="100.0" left="80.0"/>
            <tt:CenterOfGravity x="60.0" y="50.0"/>
          </tt:Shape>
          <tt:Class>
            <tt:ClassCandidate>
              <tt:Type>Human</tt:Type>
              <tt:Likelihood>1.0</tt:Likelihood><!-- tt:Likelihood is ignored -->
            </tt:ClassCandidate>
            <tt:ClassCandidate>
              <tt:Type>Face</tt:Type>
              <tt:Likelihood>1.0</tt:Likelihood>
            </tt:ClassCandidate>
          </tt:Class>
        </tt:Appearance>
        <tt:Extension>
          <Properties>
            <Property name="Gender">Male</Property><!-- Human property extension -->
            <Property name="Height">1.81</Property><!-- Human property extension -->
            <Property name="Age">48</Property><!-- Human property extension -->
          </Properties>
        </tt:Extension>
      </tt:Object>
    </tt:Frame>
  </tt:VideoAnalytics>
  <tt:Extension>
    <NavigationalData version="1.0"><!-- Navigational data extension -->
      <Latitude>16.0735273296356</Latitude>
      <Longitude>-27.2426441904356</Longitude>
    </NavigationalData>
</tt:MetadataStream>
```

Guidelines

Alternative representations of object class type are supported for Metadata search in order to support existing devices. For new development, adhere to the ONVIF Analytics Service Specification Version 19.12.

The ONVIF MetadataStream schema allows several `tt:Frame` elements in the same `tt:VideoAnalytics` element, and several `tt:VideoAnalystics` elements within a `tt:MetadataStream` element. We strongly recommend that you have only one `tt:Frame` and one `tt:VideoAnalytics` element per packet sent to the recording server. This makes it easier to find data based on timestamps and to associate metadata with the correct video frames.

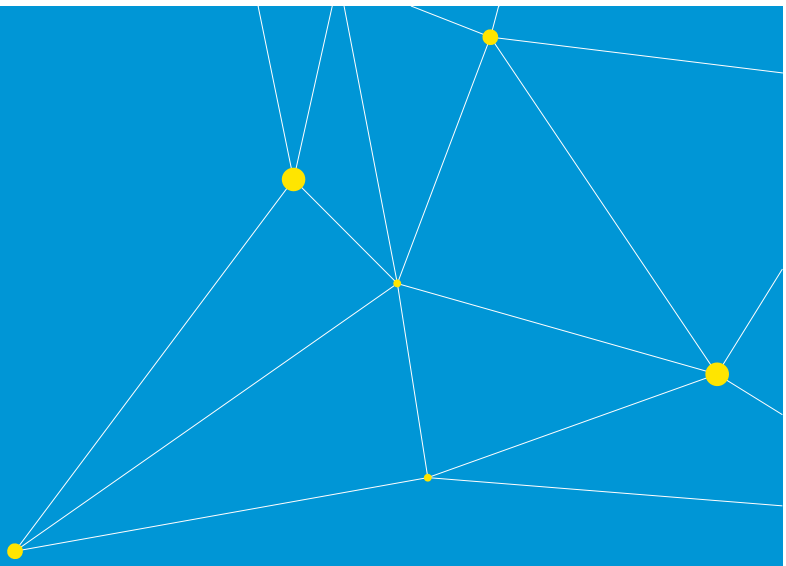
Help and troubleshooting

For further help and troubleshooting, refer to the Milestone Developer Forum.

If you are a Milestone Technology Partner, you can also raise the issue through the Milestone support case system.

References

- ONVIF Network Interface Specifications
- ONVIF Analytics Service Specification Version 19.12
- ONVIF XML Schema Version and Extension Handling White Paper
- W3C CSS Color Module Level 3
- .NET System.Drawing.KnownColor Enum
- Introduction to Metadata
- Milestone Developer Forum



Milestone Systems is a leading provider of open platform video management software; technology that helps the world see how to ensure safety, protect assets and increase business efficiency. Milestone enables an open platform community that drives collaboration and innovation in the development and use of network video technology, with reliable and scalable solutions that are proven in more than 150,000 sites worldwide. Founded in 1998, Milestone is a stand-alone company in the Canon Group.