

Physical Security Interoperability Alliance
PSIA Specification: Video Analytics Specification
Version 1.0

September 24, 2010

Copyright Physical Security Interoperability Alliance, All Rights Reserved



PSIA Video Analytics Specification

V1.0

Video Analytics Working Group

Document History

Version	Date	Author	Description
0.0	11/13/08	G. Myers	"OV Ready" specification contributed by ObjectVideo; used as baseline document
0.1	1/25/09	G. Myers	Initial VAS draft for Analytics Working Group review
0.2	1/26/09	B. Cutting	Initial draft with accepted additions/deletions to clear out remnants from source document
0.71	11/2/09	B. Cutting	Version edited for metadata content
0.73	12/15/09	J. Cook	Updated to include new ObjectModel fields
0.74	12/28/09	B. Cutting	Accepted multiple edits based on recent metadata, events, and Object model discussions in order to create a clean baseline as we get into final review. Accepted insertion of new Analytics Capabilities content for the group to begin review and edit.
0.75	2/2/10	B. Cutting	Includes final decisions on Device level analytic capabilities. Accepted comments and revisions in Section 3.4.
0.76	2/5/10	G. Myers	Updated sections 3.4.1.1, 3.4.1.2, 3.5.1 and 3.5.2 based on recent feedback in preparation for final review.
0.77	2/11/10	G. Myers	Updated section 3.4 based on the review on 2/10.
0.78	2/26/10	G. Myers	Updated section 3.4 and 3.5 based on the comments.
0.79	3/10/10	G. Myers	Changed timestamps in sections 3.4 and 3.5 to be UTC based; changed ordering of element in AnalyticsAlert and AnalyticsObject to group related items together.
0.81	04/13/10	J. Cook	Added initial VideoAnalytics root service and capabilities specification
0.82	5/11/10	B. Cutting	Reformatted Event and Metadata Output sections to follow standard format
0.83	5/26/10	J.Cook	Added CMEM stream setup, incorporated comments
0.84	6/2/10	J.Cook	Committed all changes to date, cleaned up formatting. Made changes suggested at 5/26/10 meeting
0.86	6/15/10	J.Cook	Including initial review feedback
0.87	6/22/10	J.Cook	Validated all schemas and examples

Version	Date	Author	Description
0.88	7/2/10	B. Cutting	Clean copy for final PSIA internal review
0.89	7/12/10	B. Cutting	Final draft submission from the Analytics Working Group to the PSIA for approval prior to making it available for public review.
0.89a	7/14/10	R. Richter	Minor modifications to correlate CMEM v1.1 to final Video Analytics spec.
0.90	7/21/10	B. Cutting	Final draft for public review period
1.0 Draft 1	9/6/10	B. Cutting	Incorporates agreed to changes based on public review feedback
1.0 Draft 2	9/16/10	B. Cutting/J. Cook	XSD Updated and some other minor edits
1.0	9/24/10	B. Cutting	Final 1.0 Version

Disclaimer

THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. Without limitation, PSIA disclaims all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification and to the implementation of this specification, and PSIA disclaims all liability for cost of procurement of substitute goods or services, lost profits, loss of use, loss of data or any incidental, consequential, direct, indirect, or special damages, whether under contract, tort, warranty or otherwise, arising in any way out of use or reliance upon this specification or any information herein.

No license, express or implied, by estoppel or otherwise, to any PSIA or PSIA member intellectual property rights is granted herein.

Except that a license is hereby granted by PSIA to copy and reproduce this specification for internal use only.

Contact the Physical Security Interoperability Alliance at info@psialliance.org for information on specification licensing through membership agreements.

Any marks and brands contained herein are the property of their respective owners.

Table of Contents

1	INTRODUCTION	6
1.1.	SCOPE	6
1.1.1	Conformance	6
1.1.2	References	6
1.2	REST OVERVIEW	7
1.2.1	HTTP Methods and CRUD	7
1.2.2	Rules of Thumb	7
1.2.3	Glossary of Terms	8
1.2.4	Name Case Recommendation	8
1.2.5	XML Reserved Characters	9
1.3	VIDEO ANALYTICS DEVICE DISCOVERY	10
1.4	EVENT STREAMING	10
1.5	VIDEO ANALYTICS RESOURCE STRUCTURE	11
2	VIDEO ANALYTICS SERVICE REST APIS	13
2.1	PSIA/VIDEOANALYTICS/INDEX	13
2.2	PSIA/VIDEOANALYTICS/DESCRIPTION	14
2.3	PSIA/VIDEOANALYTICS/CAPABILITIES	14
2.3.1	PSIA/VideoAnalytics/capabilities Schema Definition (XSD)	15
2.3.2	PSIA/VideoAnalytics/capabilities Elements	16
2.4	PSIA/VIDEOANALYTICS/INFO	16
	PSIA/VideoAnalytics/info Schema Definition (XSD)	17
2.4.1		17
2.4.2	PSIA/VideoAnalytics/info Elements	17
2.5	PSIA/VIDEOANALYTICS/STATUS	17
2.5.1	PSIA/VideoAnalytics/status Schema Definition (XSD)	18
2.5.2	PSIA/VideoAnalytics/status Elements	18
2.6	PSIA/VIDEOANALYTICS/CHANNELS	19
2.6.1	PSIA/VideoAnalytics/channels Schema Definition (XSD)	19
2.6.2	PSIA/VideoAnalytics/channels Elements	20
2.7	PSIA/VIDEOANALYTICS/CHANNELS/[ID]	21
2.7.1	PSIA/VideoAnalytics/channels/[ID] Schema Definition (XSD)	21
2.7.2	PSIA/VideoAnalytics/channels/[ID] schema elements	22
3	EVENT OUTPUT	24
3.1	EVENT STREAMING	24
3.1.1	Connecting to analytics streams	24
3.1.2	Streaming via RTSP stream	27
3.1.3	<AnalyticsAlert> Event Document format	29
3.1.3.1	AnalyticsAlert Elements	31
3.1.4	<AnalyticsCount> Event Document Definition	35
3.1.4.1	AnalyticsCOUNT ELEMENTS	37
3.1.4.2	<AnalyticsStatus> Event Document Definition	40
3.1.4.2	AnalyticsSTATUS Attributes	41
3.1.4.3	event streaming uri summary	42
4	METADATA STREAMING	43
4.1	METADATA PROPERTIES	45
4.2	METADATA STREAMING OPERATIONS	48
5	APPENDIX A: ANALYTICS EVENT SCHEMA DEFINITION (XSD)	49
6	APPENDIX B: REFERENCES	58
6.1	REST	58
6.2	HTTP	58

6.3	MIME	58
6.4	XML	58
6.5	DATA ENCODINGS.....	58
6.6	TIME FORMAT	58
6.7	SHA-1 HASHING ALGORITHM	58
6.8	ZLIB COMPRESSION	58
6.9	DNS-SD.....	58

1 Introduction

This document specifies an interface that enables physical security and video management systems to communicate with various IP media devices in a standardized way. This eliminates the need for device driver customization in order to achieve interoperability among products from different manufacturers. The intent of this specification is to improve the interoperability of IP-based physical security products from different vendors.

1.1. Scope

The scope for the initial release of the PSIA Video Analytics Specification (VAS) will be focused entirely on analytics capabilities discovery and analytic data output. From an output perspective, this will include multiple types of analytic events, including alerts and counts, and video analytics metadata.

The scope for the video analytic capabilities will include standard configuration data exchange to enable any analytic device to communicate to another device or application its basic analytic capabilities at the device level and the video channel level (for multi-channel devices). This includes information such as PSIA VAS version number supported analytic vendor information (name, software version number, etc.), event types and mechanisms supported, and other supported configurations.

The initial scope within this specification will NOT address analytic rule configurations. The creation of standard protocols for individual rule types will be included in the next release of the VAS.

From a technical perspective, the VAS defines a common protocol using XML over HTTP/HTTPS. This protocol is similar in nature to Web services but is geared towards lightweight computing requirements on the device. As such, this release will not use Simple Object Access Protocol (SOAP) as defined by the W3C-defined Web services but uses a simplified XML schema. In addition, the VAS protocol treats all configuration and management aspects as resources utilizing the REpresentational State Transfer (REST) architecture. The REST architecture is consistent with the core PSIA Service Model.

1.1.1 Conformance

The Video Analytics Device will host PSIA compliant services and adhere to the PSIA Service Model and PSIA Common Meta Data and Event Definition as defined in referenced documents below.

1.1.2 References

- | | |
|--------------------|--|
| [RFC 1945] | "Hypertext Transfer Protocol -- HTTP/1.0", T. Berners-Lee et al, May 1996
URL:http://www.ietf.org/rfc/rfc1945.txt |
| [RFC 2326] | "Real Time Streaming Protocol (RTSP)", H. Schulzrinne et al, April 1998
URL:http://www.ietf.org/rfc/rfc2326 |
| PSIA Service Model | www.psialliance.org/documents/PSIA-Service- |

specification [Model version 1.0.pdf](#), March 17, 2009
PSIA IP Media Device specification [www.psialliance.org/documents/PSIA-IPMD-v1r7.pdf](#),
March 17, 2009
PSIA Common Metadata and Event Model Specification [www.psialliance.org/documents/PSIA-Common-MetaEvent-Model-v1-1-r03.pdf](#)

1.2 REST Overview

REST is an approach to creating services that expose all information as resources in a uniform way. This approach is quite different from the traditional Remote Procedure Call (RPC) mechanism which identifies the functions that an application can call. Put simply, a REST Web application is noun-driven while an RPC Web application is verb-driven. For example, if a Web application were to define an RPC API for user management, it might be written as follows:

```
GET http://webserver/getUserList
GET http://webserver/getUser?userid=100
POST http://webserver/addUser
POST http://webserver/updateUser
GET http://webserver/deleteUser?userid=100
```

On the other hand, a REST API for the same operations would appear as follows:

```
GET http://webserver/users
GET http://webserver/users/user100
POST http://webserver/users
PUT http://webserver/users/user100
DELETE http://webserver/users/user100
```

Part of the simplicity of REST is its uniform interface for operations. Since everything is represented as a resource, create, retrieve, update, and delete (CRUD) operations use the same URI.

1.2.1 HTTP Methods and CRUD

The CRUD operations are defined by the HTTP method as shown in the table below.

HTTP Method	Operation
POST	Create the resource
GET	Retrieve the resource
PUT	Update the resource
DELETE	Delete the resource

1.2.2 Rules of Thumb

-
1. GET calls should never change the system state. They are meant to only return data to the requestor and not to have any side effects
 2. POST calls should only be used to ADD something that did not already exist.
 3. PUT calls are expected to update an existing resource but if the resource specified does not already exist, it can be created as well. This will be the assumed default behavior of PUT calls. If any resource wishes to deviate from this behavior, it should be considered an exception and this should be noted in the implementation notes of the resource.

1.2.3 Glossary of Terms

- **Video Analytics Device (“Device”)** - IP device or platform that hosts a single channel or multiple channels of video analytic processing. It is able to take in video for analytic processing from external video sources or in itself be the video source (i.e., an IP camera)
- **Software Application (“Application”, e.g. VMS)** - Software Application that is capable of communicating with and controlling PSIA Services and IP Media Devices. This Software Application communicates with Video Analytic Devices to receive analytic events and metadata and provides user interfaces to query the data.
- **Events** – A specific activity detected within the video analytic device as defined by specific rules or conditions, which can be sent to a monitoring Application.
- **Metadata** – Frame-by-frame data generated by the video analytics device representing the ongoing activity and object motion for a video channel, which can be used to annotate a video stream or event with additional information.

1.2.4 Name Case Recommendation

There are no requirements for use of case (capitalization) of URI (Service names) or XML/XSD (Tag-names).

As a historical note, the PSIA Core members (during review of Service Model specification) had suggested the following recommendations for use of upper & lower casing of names:

- **For REST URIs: Service names have the first letter capitalized (“upper camel case”), while Resources are not capitalized (“lower camel case”). Example: /PSIA/System/video/inputs.**
- **For XML/XSD: Tag-names which refer to PSIA-defined schema types (simple or complex) have the first letter capitalized (“upper camel case”). Tag-names which refer to predefined, W3C schema types are not capitalized (“lower camel case”). Obviously, no restriction can be placed on values within the tags. Example:**

```
<xs:complexType name="Entry">
  <xs:sequence>
    <xs:element name="id" type="xs:integer" />
    <xs:element name="OtherType" type="OtherType" />
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="OtherType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="a" />
    <xs:enumeration value="b" />
    <xs:enumeration value="c" />
  </xs:restriction>
</xs:simpleType>
```

(Also, see 'Service.xsd' from Core Model.)

Given the exceptions and difficulty with enforcing this recommendation, PSIA REST URIs and XSDs are not required to abide strictly by these recommendations. Basically, all REST URIs that are subordinate to a parent service (i.e. child nodes) are lower camel-case in notation since the “index” resource of the parent service identifies them explicitly as services or resources.

1.2.5 XML Reserved Characters

Within an XML document, some characters are reserved for language use. If these characters appear in data values, they should be replaced with their Entity Reference equivalents (akin to ANSI Escapes) to avoid parsing errors.

Character	Description	Entity Reference	Comments
<	Less than	<	May never appear in data.
&	Ampersand	&	May never appear in data.
>	Greater than	>	Replace as best practice.
“	Double quote	"	Replace as best practice.
‘	Single quote	'	Replace as best practice.
%	Percent	%	Replace as best practice.

Note that � (null) is not permitted.

For example, the URL

“rtsp://144.70.13.92:554/PSIA/Streaming/tracks/27?offset=a07724&endtime=2009-05-18T10:31.25” would appear as follows in XML:

```
<playbackURI>rtsp://144.70.13.92:554/PSIA/Streaming/tracks/27?offset=a07724&amp;endtime=2009-05-18T10:31.25</playbackURI>
```

1.3 Video Analytics Device Discovery

Video Analytics devices that are standalone—in that they do not support other PSIA services—must support discovery using the standard mechanisms defined by the PSIA. Zeroconf (Zero Configuration Networking) technology specifies the mDNS (Multicast DNS) and DNS-SD (DNS Service Discovery) protocols, as described in <http://files.dns-sd.org/draft-cheshire-dnsextdns-sd.txt>), as the mechanisms to discover/locate service and devices on an IP network. All PSIA protocols require DNS-SD and mDNS for device discovery. To support this discovery model, the PSIA is registering a DNS SRV (RFC 2782) service type to be used to discover all PSIA nodes via DNS-SD and/or mDNS.

All PSIA devices are required, under the PSIA Service Model, to support ZeroConf network discovery (see PSIA Service Model specification), but not ZeroConf's use of Local IP address allocation; this is optional. The discovery requirement consists of support for both DNS-SD **and** mDNS such that large and small networks can be supported. As a result, this requires a Video Analytics device to attempt to register, via DNS-SD, its own service record (i.e. SRV), during system startup, with each of its designated DNS servers (as defined in its IP configuration). Irrespective of whether the SRV registrations pass, or fail, the device is still required to honor mDNS requests for its service type definition. The format of a PSIA compliant Video Analytics device's SRV should always start with the DNS SRV 'services' and 'protocol' prefixes of : “**_psia.tcp.**”, followed by the other pertinent information as outlined in RFC 2782, and in the following paragraph.

DNS-SD discoveries, initiated by entities seeking PSIA devices and services (i.e. clients, management servers, etc.), should use the PSIA's public DNS service type to discover the device according to DNS Service Discovery (<http://www.dns-sd.org/ServiceTypes.html>). Once a device is established as a PSIA-compliant device, its services and resources can be discovered using standard HTTP GETs using the standard, mandatory REST resources.

The following information should be advertised in the SRV record:

- A (REST) path of “/index” – can be obtained from the “path” key in the TXT record
- The {host} – via an IP address or domain name, can be obtained from the service's SRV record
- The {port} – can be obtained from the service's SRV record
- The version of the DNS SRV record in “txtvers”
- The PSIA protocol version in “protovers”

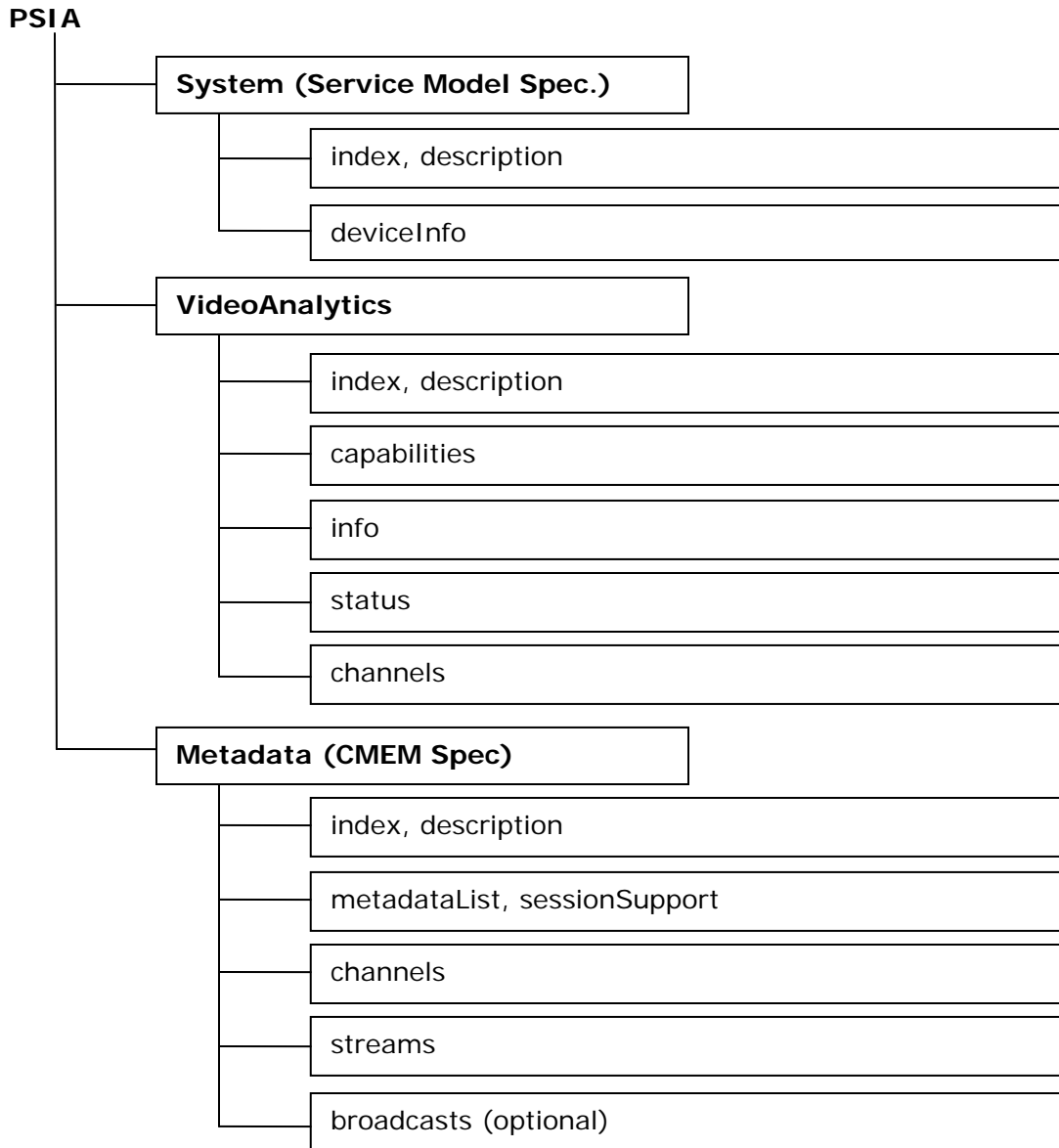
Once a PSIA-compliant device has been discovered, an HTTP GET of its mandatory '/index' resource returns a list of the services that the device supports. At this point, the standard methods are used to “walk” the namespace tree and discover the supported services and resources. It should be noted that the “index” resource returns only the first level resources of a node while the “indexr” resource will return a recursive tree structured list with the current resource as root.

As previously noted, all PSIA standalone Video Analytics Devices must support the required '/System' and '/Metadata' based resources, in addition to the previously specified '/VideoAnalytics' resources. Other REST resources, as defined by system functionality, may be provided by manufacturers.

1.4 Event Streaming

All Video Analytic Devices that support events, alarming, notification, etc., must conform to the PSIA Common Metadata and Event Model as defined in the “**PSIA-Common-MetaEvent-Model-v1-1-r03.pdf**” specification. This specification is available to PSIA members on the PSIA members on the PSIA web site.

1.5 Video Analytics Resource Structure



The above diagram depicts the basic REST resources supported by a spec-compliant PSIA Video Analytics Device as outlined in this document. The colored boxes in the diagram indicate resources that are ‘Services’. Services are resources that carry attributes defined by “description” and “capabilities” schemas (see below and PSIA Service Model specification). Additionally, the resource hierarchy determines the REST URI structures used to interact with each resource.

Resource Name	Description	Mandatory/Optional
description	will respond to an HTTP GET with a <ResourceDescription> datablock	Mandatory
capabilities	will respond to an HTTP GET with a resource-specific datablock	Generally Optional; Required for some Video Analytic Services
index	will respond to an HTTP GET with a <ResourceList> datablock	Mandatory
indexr	will respond to an HTTP GET with a <ResourceList> datablock	Optional

2 Video Analytics Service REST APIs

The 'VideoAnalytics' service is the 'root' service for all VAS functions. Since Video Analytic devices are required to be compliant with the PSIA Service Model, these devices must also have the required '/System' service and its relevant resources. For more details, see the PSIA Service Model specification.

The Video Analytics root contains a number of 'local' resources:

- 'index' – Required by the PSIA Service Model
- 'description' – Required by the PSIA Service Model
- 'capabilities' – Required due to the complexity of the services provided, and defines the capabilities of the individual device
- 'info' – Required to provide information regarding the VideoAnalytics vendor and version
- 'status' – Required to provide the status of the analytic features of the device
- 'channels' – Required to provide summary and detailed information for each configured channel within the video analytics device

2.1 PSIA/VideoAnalytics/index

PSIA mandated resource that lists all of the 1st level resources contained by '/VideoAnalytics'. Please note that the recursive version, 'indexr' is not required for VideoAnalytics Devices. (See the 'capabilities' resource description for more details)

URI	/PSIA/VideoAnalytics/index			Type	Resource
Requirement Level	- All -				
Function	PSIA Mandatory REST resource/object that enumerates the 1 st level child resources for '/VideoAnalytics'				
Methods	Query String(s)	Inbound Data	Return Result		
GET	None		<ResourceList>		
PUT	N/A	N/A	<ResponseStatus w/error code>		
POST	N/A	N/A	<ResponseStatus w/error code>		
DELETE	N/A	N/A	<ResponseStatus w/error code>		
Notes	The 'GET' request issued to retrieve an instance of the 'ResourceList' XML schema. See the Service Model specification, Section 7, 8, and 10 for schema details. The resources listed are exposed by the VideoAnalytics service and are available to be queried by the client.				
Example(s)	<pre><?xml version="1.0" encoding="UTF-8"?> <ResourceList version="1.0" xmlns="urn:psialliance-org"> <Resource xlink:href="/PSIA/VideoAnalytics/index"> <name>index</name> <type>resource</type> </Resource> <Resource xlink:href="/PSIA/VideoAnalytics/description"> <name>description</name> <type>resource</type> </Resource> <Resource xlink:href="/PSIA/VideoAnalytics/capabilities"> <name>capabilities</name> <type>resource</type> </Resource></pre>				

	<pre> <Resource xlink:href="/PSIA/VideoAnalytics/info"> <name>info</name> <type>service</type> </Resource> <Resource xlink:href="/PSIA/VideoAnalytics/status"> <name>status</name> <type>service</type> </Resource> <Resource xlink:href="/PSIA/VideoAnalytics/channels"> <name>channels</name> <type>service</type> </Resource> </ResourceList> </pre>
--	---

2.2 /PSIA/VideoAnalytics/description

URI	/PSIA/VideoAnalytics/description	Type	Resource
Requirement Level	- All -		
Function	PSIA REST resource/object that describes the functional behavior of the base resource (see PSIA Service Model Sections 7, 8, and 10 for more details). The description returned provides details on which resource methods are exposed and what query parameters are supported.		
Methods	Query String(s)	Inbound Data	Return Result
GET	None	None	<ResourceDescription>
PUT	N/A	N/A	<ResponseStatus w/error code>
POST	N/A	N/A	<ResponseStatus w/error code>
DELETE	N/A	N/A	<ResponseStatus w/error code>
Notes	The 'GET' request issued to retrieve an instance of the 'ResourceDescription' XML schema. See the Service Model specification for schema details.		
Example(s)	<pre> <?xml version="1.0" encoding="UTF-8"?> <ResourceDescription version="1.0" xmlns="urn:psialliance-org"> <name>VideoAnalytics</name> <version>1</version> <type>service</type> <get> <queryStringParameterList>none</queryStringParameterList> <inboundXML>none</inboundXML> <function>Video Analytics root service</function> <returnResult>none</returnResult> <notes/> </get> <put/> <post/> <delete/> </ResourceDescription> </pre>		

2.3 /PSIA/VideoAnalytics/capabilities

URI	/PSIA/VideoAnalytics/capabilities		Type	Resource
Requirement Level	- All -			
Function	PSIA REST resource/object that describes the Video Analytics capabilities of the device. The document returned conforms to the AnalyticsCapabilities schema which contains a list of attributes regarding the functionality supported by this instance of the VideoAnalytics Service.			
Methods	Query String(s)	Inbound Data	Return Result	
GET	None	None	<AnalyticsCapabilities>	
PUT	N/A	N/A	<ResponseStatus w/error code>	
POST	N/A	N/A	<ResponseStatus w/error code>	
DELETE	N/A	N/A	<ResponseStatus w/error code>	
Notes	The 'GET' request issued to retrieve an instance of the 'AnalyticsCapabilities' XML schema. See the Service Model specification for schema details.			
Example(s)	<pre><?xml version="1.0" encoding="UTF-8"?> <AnalyticsCapabilities xmlns="urn:psialliance-org/analytics"> <SupportsAnalyticsUpgrade>true</SupportsAnalyticsUpgrade> <SupportsAnalyticsLicenseUpgrade>true</SupportsAnalyticsLicenseUpgrade> <SupportsChannelReset>true</SupportsChannelReset> <SupportsAlertStreaming>true</SupportsAlertStreaming> <SupportsCountStreaming>true</SupportsCountStreaming> <SupportedAnalyticsFrameSizes> <AnalyticFrameSize> <Width>640</Width> <Height>480</Height> </AnalyticFrameSize> <AnalyticFrameSize> <Width>320</Width> <Height>240</Height> </AnalyticFrameSize> </SupportedAnalyticsFrameSizes> <SupportsSnapshot>true</SupportsSnapshot> </AnalyticsCapabilities></pre>			

2.3.1 /PSIA/VideoAnalytics/capabilities Schema Definition (XSD)

The schema for the VideoAnalytics base service provides a list of information regarding the VideoAnalytics capabilities of the device.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:psialliance-org/analytics" elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:cmem="urn:psialliance-org"
  xmlns="urn:psialliance-org/analytics">
  <xs:element name="AnalyticsCapabilities" type="AnalyticsCapabilities"/>
  <xs:complexType name="AnalyticsCapabilities">
    <xs:sequence>
      <xs:element name="SupportsAnalyticsUpgrade" type="xs:boolean" />
      <xs:element name="SupportsAnalyticsLicenseUpgrade" type="xs:boolean" />
      <xs:element name="SupportsChannelReset" type="xs:boolean" />
      <xs:element name="SupportsAlertStreaming" type="xs:boolean" />
      <xs:element name="SupportsCountStreaming" type="xs:boolean" />
      <xs:element name="SupportedAnalyticsFrameSizes"
type="SupportedAnalyticsFrameSizes" />
      <xs:element name="SupportsSnapshot" type="xs:boolean" />
    </xs:sequence>
  </xs:complexType>
```

```

<xs:complexType name="SupportedAnalyticsFrameSizes">
  <xs:annotation>
    <xs:documentation xml:lang="en">Defines the list of supported frame
sizes</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="AnalyticFrameSize" minOccurs="1" maxOccurs="unbounded"
type="AnalyticFrameSize"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="AnalyticFrameSize">
  <xs:sequence>
    <xs:element name="Width" minOccurs="1" maxOccurs="1" type="xs:int"/>
    <xs:element name="Height" minOccurs="1" maxOccurs="1" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>

```

2.3.2 /PSIA/VideoAnalytics/capabilities Elements

Element Name	Requirement	Description
SupportsAnalyticsUpgrade	Mandatory	Provides a boolean value which determines whether the VideoAnalytics device is software upgradable
SupportsAnalyticsLicenseUpgrade	Optional	Indicates whether the channel on this device can be relicensed through some mechanism to change the operation of the analytics
SupportsChannelReset	Mandatory	Provides a boolean value which determines whether the VideoAnalytics device supports resets to be sent to an individually configured channel. If true channel reset is supported, if false channels can only be reset via a device reset.
SupportsAlertStreaming	Mandatory	Boolean value which determines whether the device supports alert streaming to the client via a reliable connection. If true connection can be made to the device to receive alerts, if false then alert push is used to pass alerts.
SupportsCountStreaming	Optional	Boolean value which determines whether count information can be streamed to clients via reliable connection
SupportedAnalyticsFrameSizes	Optional	Provides a list of the supported frame sizes used for Analytic processing
SupportsSnapshot	Mandatory	Analytics engines supporting snapshots should set this value to true. Otherwise, the engine does not support Snapshots

2.4 /PSIA/VideoAnalytics/info

URI	/PSIA/VideoAnalytics/info		Type	Resource
Requirement Level	- All -			
Function	PSIA REST resource/object that provides basic vendor and version information regarding the VideoAnalytics service			
Methods	Query String(s)	Inbound Data	Return Result	
GET	None	None	<AnalyticsInfo>	
PUT	N/A	N/A	<ResponseStatus w/error code>	

POST	N/A	N/A	<ResponseStatus w/error code>
DELETE	N/A	N/A	<ResponseStatus w/error code>
Notes	The 'GET' request issued to retrieve an instance of the 'info' XML schema, defined below		
Example(s)	<pre><?xml version="1.0" encoding="UTF-8"?> <AnalyticsInfo xmlns="urn:psialliance-org/analytics"> <AnalyticsVendor>ABC</AnalyticsVendor> <AnalyticsVersion>1</AnalyticsVersion> </AnalyticsInfo></pre>		

2.4.1 /PSIA/VideoAnalytics/info Schema Definition (XSD)

The info schema for the VideoAnalytics base service provides a list of information regarding the VideoAnalytics vendor and software version. In addition, vendor specific information can be added to this as necessary.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:psialliance-org/analytics" elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:cmem="urn:psialliance-org"
  xmlns="urn:psialliance-org/analytics">
  <xs:element name="AnalyticsInfo" type="AnalyticsInfo"/>
  <xs:complexType name="AnalyticsInfo">
    <xs:sequence>
      <xs:element name="AnalyticsVendor" type="xs:token" />
      <xs:element name="AnalyticsVersion" type="xs:token" />
      <xs:any namespace="##any" processContents="lax" minOccurs="0"
maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

2.4.2 /PSIA/VideoAnalytics/info Elements

Element Name	Requirement	Description
AnalyticsVendor	Mandatory	Provides a vendor-specific string identifying the organization
AnalyticsVersion	Mandatory	Provides a vendor-specific string identifying the software/firmware version

2.5 /PSIA/VideoAnalytics/status

URI	/PSIA/VideoAnalytics/status	Type	Resource
Requirement Level	- All -		
Function	PSIA REST resource/object that provides basic information regarding the status of the VideoAnalytics service		
Methods	Query String(s)	Inbound Data	Return Result
GET	None	None	<AnalyticsStatus>
PUT	N/A	N/A	<ResponseStatus w/error code>

POST	N/A	N/A	<ResponseStatus w/error code>
DELETE	N/A	N/A	<ResponseStatus w/error code>
Notes	The 'GET' request issued to retrieve an instance of the 'info' XML schema, defined below		
Example(s)	<pre><?xml version="1.0" encoding="UTF-8"?> <AnalyticsStatus xmlns="urn:psialliance-org/analytics"> <Status>OK</Status> <StatusMessage>OK</StatusMessage> <CurrentTimestamp>2010-06-21T14:26:00</CurrentTimestamp> <LastBootTimestamp>2010-06-21T14:26:00</LastBootTimestamp> </AnalyticsStatus></pre>		

2.5.1 /PSIA/VideoAnalytics/status Schema Definition (XSD)

The schema for the VideoAnalytics base service provides the current status of the Analytics service.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:psialliance-org/analytics" elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:cmem="urn:psialliance-org"
  xmlns="urn:psialliance-org/analytics">
  <xs:element name="AnalyticsStatus" type="AnalyticsStatus"/>
  <xs:complexType name="AnalyticsStatus">
    <xs:sequence>
      <xs:element name="Status" type="ValidStatus" />
      <xs:element minOccurs="0" maxOccurs="1" name="StatusMessage" type="xs:token" />
      <xs:element name="CurrentTimestamp" maxOccurs="1" type="xs:dateTime" />
      <xs:element name="LastBootTimestamp" minOccurs="0" maxOccurs="1" type="xs:dateTime" />
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="ValidStatus">
    <xs:restriction base="xs:NMTOKEN">
      <xs:enumeration value="OK" />
      <xs:enumeration value="WARNING" />
      <xs:enumeration value="ERROR" />
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

2.5.2 /PSIA/VideoAnalytics/status Elements

Element Name	Requirement	Description
Status	Mandatory	Provides the current status of the analytics engine. Valid values are "OK", "WARNING" and "ERROR" Usage Definitions: OK – Video Analytics operating normally WARNING – Video Analytics operating, but some condition exists that may affect performance ERROR – Video Analytics not operating
StatusMessage	Optional	Provides a vendor-specific description of the status of the analytics engine. While optional, this is a recommended attribute to include
CurrentTimestamp	Mandatory	Provides a timestamp for the time that the status value was provided

Element Name	Requirement	Description
LastBootTimestamp	Optional	Provides a timestamp when the analytics service was last booted

2.6 /PSIA/VideoAnalytics/channels

URI	/PSIA/VideoAnalytics/channels	Type	Resource
Requirement Level	- All -		
Function	PSIA REST resource/object that provides a list of the channels defined within the system		
Methods	Query String(s)	Inbound Data	Return Result
GET	None	None	<Channels>
PUT	N/A	N/A	<ResponseStatus w/error code>
POST	N/A	N/A	<ResponseStatus w/error code>
DELETE	N/A	N/A	<ResponseStatus w/error code>
Notes	The 'GET' request issued to retrieve an instance of the 'info' XML schema, defined below		
Example(s)	<pre><?xml version="1.0" encoding="UTF-8"?> <Channels xmlns="urn:psialliance-org/analytics"> <ChannelSummary> <ID>1</ID> <Name>Channel 1</Name> <VideoSource>3E4A2E98-78A7-11DF-84E5-95D4DED72085</VideoSource> <AnalyticsType>TripWire</AnalyticsType> <IsAnalyticsEnabled>true</IsAnalyticsEnabled> </ChannelSummary> </Channels></pre>		

2.6.1 /PSIA/VideoAnalytics/channels Schema Definition (XSD)

The schema for the VideoAnalytics base service provides a list of the defined channels within the service with a summary of the current configuration.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:psialliance-org/analytics" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:cmem="urn:psialliance-org"
xmlns="urn:psialliance-org/analytics">
  <xs:element name="Channels" type="Channels" />
  <xs:complexType name="Channels">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="ChannelSummary"
type="ChannelSummary" />
    </xs:sequence>
  </xs:complexType>

  <xs:element name="ChannelSummary" nillable="true" type="ChannelSummary" />
  <xs:complexType name="ChannelSummary">
    <xs:sequence>
      <xs:element name="ID" type="xs:unsignedInt" />
      <xs:element minOccurs="1" maxOccurs="1" name="Name" type="xs:token" />
      <xs:element minOccurs="0" maxOccurs="1" name="VideoSource" type="GUID"/>
      <xs:element minOccurs="0" maxOccurs="1" name="AnalyticsType" type="xs:token"
/>
      <xs:element name="IsAnalyticsEnabled" type="xs:boolean" />
    </xs:sequence>
```

```

</xs:complexType>

<xs:simpleType name="GUID">
  <xs:annotation>
    <xs:documentation xml:lang="en"> The representation of a GUID, generally the
id of an
        element. </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:pattern
      value="[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-
F0-9]{12}"
    />
  </xs:restriction>
</xs:simpleType>
</xs:schema>

```

2.6.2 /PSIA/VideoAnalytics/channels Elements

Element Name	Requirement	Description
ID	Mandatory	Provides a specific unsigned integer value as an ID to identify the analytics channel. This ID is unique within the device and can be used to reference detailed channel information.
Name	Mandatory	A user defined string associated with the analytics channel.
VideoSource	Optional	Provides the GUID of the associated Video Source. This ID can be used to reference the corresponding source within the IPMD root service
AnalyticsType	Optional	Provides a way to identify the type of analytics running on the channel, such as a package name, license type/ID, etc.
IsAnalyticsEnabled	Mandatory	Boolean value which indicates the current state of analytics on this channel. (Default: false)

2.7 /PSIA/VideoAnalytics/channels/[ID]

URI	/PSIA/VideoAnalytics/channels/[id]	Type	Resource
Requirement Level	- All -		
Function	PSIA REST resource/object that provides detailed information of the channel configured within the Analytics service.		
Methods	Query String(s)	Inbound Data	Return Result
GET	None	None	<Channel>
PUT	N/A	N/A	<ResponseStatus w/error code>
POST	N/A	N/A	<ResponseStatus w/error code>
DELETE	N/A	N/A	<ResponseStatus w/error code>
Notes	The 'GET' request issued to retrieve an instance of the 'info' XML schema, defined below		
Example(s)	<pre><?xml version="1.0" encoding="UTF-8"?> <Channel xmlns="urn:psialliance-org/analytics"> <ID>1</ID> <Name>Channel 1</Name> <VideoSource>3E4A2E98-78A7-11DF-84E5-95D4DED72085</VideoSource> <AnalyticsType>TripWire</AnalyticsType> <IsAnalyticsEnabled>true</IsAnalyticsEnabled> <IsAnalyticsCalibrated>false</IsAnalyticsCalibrated> <AnalyticsFrameSize> <Width>320</Width> <Height>240</Height> </AnalyticsFrameSize> <AnalyticsCapabilities> <MaxRules>1</MaxRules> <SupportsAlertOutput>true</SupportsAlertOutput> <SupportsFullAlerts>true</SupportsFullAlerts> <SupportsCountOutput>false</SupportsCountOutput> <SupportsMetadataOutput>true</SupportsMetadataOutput> </AnalyticsCapabilities> </Channel></pre>		

2.7.1 /PSIA/VideoAnalytics/channels/[ID] Schema Definition (XSD)

The channels resource, with [ID], provides the detailed information for the specified channel. The document returned should conform to the following schema.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:psialliance-org/analytics" elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:cmem="urn:psialliance-org"
  xmlns="urn:psialliance-org/analytics">
  <xs:element name="Channel" type="Channel" />
  <xs:complexType name="Channel">
    <xs:sequence>
      <xs:element name="ID" type="xs:unsignedInt" />
      <xs:element minOccurs="1" maxOccurs="1" name="Name" type="xs:token" />
      <xs:element minOccurs="0" maxOccurs="1" name="VideoSource" type="GUID" />
      <xs:element minOccurs="0" maxOccurs="1" name="AnalyticsType" type="xs:token" />
      <xs:element name="IsAnalyticsEnabled" minOccurs="0" maxOccurs="1"
        type="xs:boolean" />
      <xs:element name="IsAnalyticsCalibrated" minOccurs="0" maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

```

type="xs:boolean" />
  <xs:element name="AnalyticsFrameSize" minOccurs="0" maxOccurs="1"
type="AnalyticFrameSize" />
  <xs:element name="AnalyticsCapabilities" type="AnalyticsCapabilities" />
</xs:sequence>
</xs:complexType>

<xs:element name="AnalyticsCapabilities" nillable="true" type="AnalyticsCapabilities"
/>
<xs:complexType name="AnalyticsCapabilities">
  <xs:sequence>
    <xs:element name="MaxRules" type="xs:int" />
    <xs:element name="SupportsAlertOutput" type="xs:boolean" />
    <xs:element name="SupportsFullAlerts" type="xs:boolean" />
    <xs:element name="SupportsCountOutput" type="xs:boolean" />
    <xs:element name="SupportsMetadataOutput" type="xs:boolean" />
  </xs:sequence>
</xs:complexType>

<xs:complexType name="AnalyticFrameSize">
  <xs:sequence>
    <xs:element name="Width" minOccurs="1" maxOccurs="1" type="xs:int"/>
    <xs:element name="Height" minOccurs="1" maxOccurs="1" type="xs:int"/>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="GUID">
  <xs:annotation>
    <xs:documentation xml:lang="en"> The representation of a GUID, generally the
id of an
        element.
    </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:pattern
      value="[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-
F0-9]{12}"
    />
  </xs:restriction>
</xs:simpleType>
</xs:schema>

```

2.7.2 /PSIA/VideoAnalytics/channels/[ID] schema elements

Element Name	Requirement	Description
ID	Mandatory	Provides a specific unsigned integer value as an ID to identify the analytics channel. This ID is unique within the device and can be used to reference detailed channel information.
AnalyticsType	Optional	Provides a way to identify the type of analytics running on the channel, such as a package name, license type/ID, etc.
Name	Mandatory	A user defined string associated with the analytics channel.
VideoSource	Optional	Provides the GUID of the associated Video Source. This ID can be used to reference the corresponding source within the IPMD root service
IsAnalyticsEnabled	Optional	Provides a flag indicating the current state of analytics on this channel. (Default: false)
IsAnalyticsCalibrated	Optional	Provides a boolean value indicating whether the current analytics Field of view is calibrated. As such information provided within events will contain calibrated values.
AnalyticsFrameSize	Optional	Identifies the frame size actually being processed by the video analytic engine

Element Name	Requirement	Description
AnalyticsCapabilities	Mandatory	Provides values determining the analytics capabilities of the device. These settings are also returned by querying the VideoAnalytics/capabilities URL.
MaxRules	Mandatory	Defines the maximum number of rules that can be defined on this channel.
SupportsAlertOutput	Mandatory	Specifies whether the channel supports alert output
SupportsFullAlerts	Mandatory	Specifies whether the channel supports all full or basic alert information
SupportsCountOutput	Mandatory	Specifies whether the channel supports counting output
SupportsMetadataOutput	Mandatory	Specifies whether the channel supports meta-data streaming.

3 Event Output

The VAS assumes the analytics Device outputs events in three forms: Alerts, Counts, and Status Events. The primary difference between these three event types is that alert output (for security applications) contains the full information about the alert including snapshots (if supported) and target (object) information, whereas the count output (for business intelligence applications) contains only the numerical value so as to reduce the bandwidth and storage requirements for count data. Depending on the rule and output mechanism, the numerical value for counts could be cumulative (e.g., the number of times a line was crossed) or instantaneous (e.g., the number of people in the area at this exact moment).

The two mechanisms to receive events are:

- Through a synchronous mechanism where an application initiates the communication (streaming) at both the channel and device level and maintains a persistent connection.
- Through an asynchronous mechanism where the device connects and sends events to an external application (pushing) on the device level and closes the connection after each push of event information

Streaming of alert and count data passes the information to an application as the events happen. This mechanism is best suited for when the application needs every event as there is less likelihood of missed events as long as there is a listening application.

Pushing is similar to streaming except that the device is responsible for connecting to an external application (called an event receiver) and pushing the event. In streaming, if there is no listening application, the event is not passed along. The VAS allows specification of up to two event receivers working in either a failover or redundancy mode. Pushing is best suited for networks where persistent connections are not feasible or reliable since the device can connect on demand to push the events. The push mechanism is only configured at the device level, although the individual channels can be configured to participate or not.

For event output, the VAS references the event stream functionality (PSIA/Metadata/stream) defined in the PSIA Common Metadata/Event Model (CMEM) specification. The VAS will define the video analytics event data types but leverage the output APIs in the CMEM specification.

3.1 Event Streaming

Alerts, counts and status events can be streamed to a waiting application on the same connection. A client application would make a request for asynchronous events on a socket that is dedicated to receiving events. Once the device receives this request, it will maintain the connection and send out events as they happen to the waiting application(s). Events can be streamed for the entire device or from a specified channel.

The PSIA video analytics specification conforms to the PSIA CMEM for sending events. This specification defines the mechanism by which client applications can connect to analytics meta-data and event streams. This document covers the resources, with examples, that are required for conforming analytics systems to support. For additional information on the complete CMEM model, refer to the PSIA CMEM specification, Sections 9 and 10 for details.

3.1.1 Connecting to analytics streams

According to the CMEM specification, for an application to retrieve events and meta-data from an analytic device, a request is made to the following REST resource:

URI	/PSIA/Metadata/stream		Type	Resource
Requirement Level	- All -			
Function	CMEM defined REST resource/object that describes how to stream ALL video analytics event types from a device. The documents returned conforms to the AnalyticsAlert, AnalyticsCount, and AnalyticsStatus schemas defined in subsequent sections, which contains a list of attributes that define the relevant information for an analytics alert.			
Methods	Query String(s)	Inbound Data	Return Result	
GET	class=[AnalyticsType] &streamID=[ID]	None	<ResponseStatus> w/error code OR <MetaSessionParms>	
PUT	N/A	N/A	<ResponseStatus w/error code>	
POST	class=[AnalyticsType]	<MetaSessionParms>	<ResponseStatus w/error code>	
DELETE	N/A	N/A	<ResponseStatus w/error code>	
Notes	<p>The Analytics specification allows additional query parameters to be sent to extend the functionality of the CMEM specification. These parameters are used to setup streams of different types and also to determine if the output format for the events is "full", i.e. all optional fields are required to be sent if available. Examples are as follows:</p> <p>To retrieve a stream of events with "full" information, use the following query parameters:</p> <pre style="text-align: center;">/PSIA/Metadata/stream?output=full</pre> <p>To retrieve a stream containing only certain types of events (Note: this is a shortcut instead of utilizing the <metaDataNameList> element within the <MetaSessionParms> document):</p> <pre style="text-align: center;">/PSIA/Metadata/stream?class=[AnalyticsType]</pre> <p>Where [AnalyticsType] can be one of the following:</p> <ul style="list-style-type: none"> • "/psialliance.org/VideoAnalytics/Alert" or ///Alert (*when a device does not have other classes of metadata/event information) • "/psialliance.org/VideoAnalytics/Count" or ///Count (*Ibid) • "/psialliance.org/VideoAnalytics/Status" or ///Status (*Ibid) <p>In addition, if the application requires that all video analytics events are to be streamed then the following value can be used for the [AnalyticsType].</p> <ul style="list-style-type: none"> • "/psialliance.org/VideoAnalytics/ 			
Example 1	The following example shows how to create a non-transactional Asynchronous Reliable push stream with host 192.168.1.100, port 1234,			

	without authentication, and requesting only analytics alerts.
	<pre> <MetaSessionParms version="1.1" xmlns="urn:psialliance-org"> <MetaXportParms> <metaSessionID>0</metaSessionID> <metaFormat>xml-psia</metaFormat> <metaSessionProtocolType> RESTAsyncSessionBackSourceSend </metaSessionProtocolType> <metaSessionFlowType>datastream</metaSessionFlowType> <metadataNameList> <metadataIDString> /psialliance.org/VideoAnalytics/Alert </metadataIDString> </metadataNameList> <netaddress>192.168.1.100:1234</netaddress> </MetaXportParms> </MetaSessionParms> </pre>
Example 2	The following example shows how to create a transactional Asynchronous Reliable push stream, with authentication, requesting only analytics alerts and status events.
	<pre> <MetaSessionParms version="1.1" xmlns="urn:psialliance-org"> <MetaXportParms> <metaSessionID>0</metaSessionID> <metaFormat>xml-psia</metaFormat> <metaSessionProtocolType> RESTAsyncSessionBackSourceSend </metaSessionProtocolType> <metaSessionLogin> <authMode>digest</authMode> <userLogin> <userName>aUser</userName> <password>aPassword</password> </userLogin> </metaSessionLogin> <metaSessionFlowType>transaction</metaSessionFlowType> <metadataNameList> <metadataIDString> /psialliance.org/VideoAnalytics/Alert </metadataIDString> <metadataIDString> /psialliance.org/VideoAnalytics/Status </metadataIDString> </metadataNameList> <netaddress>192.168.1.100:1234</netaddress> </MetaXportParms> </MetaSessionParms> </pre>
Example 3	The following example shows how to create a UDP stream, via an HTTP request to the stream resource for Alerts only.

```

<MetaSessionParms version="1.1" xmlns="urn:psialliance-org">
  <MetaXportParms>
    <metaSessionID>0</metaSessionID>
    <metaFormat>xml-psia</metaFormat>
    <metaSessionProtocolType>
      RESTRTPStreamSrcOutUDP
    </metaSessionProtocolType>
    <netAddress>239.240.108.32:5004</netAddress>
    <netcastMode>multicast</netcastMode>
    <metadataNameList>
      <metadataIDString>
        /psialliance.org/VideoAnalytics/Alert
      </metadataIDString>
    </metadataNameList>
  </MetaXportParms>
</MetaSessionParms>

```

3.1.2 Streaming via RTSP stream

In addition to being able to create a session via HTTP, the video analytics specification also supports passing event and metadata information as part of an RTSP stream. In order to do this additional information needs to be added to the SDP, returned as a result of an RTSP DESCRIBE, that defines where the stream is located. Below is an example of the information to be added to the SDP for streaming RTP/UDP meta-data on port 5003:

```

m=metadata 5003 RTP/AVP 99
i=PSIA Metadata Stream using XML format
a=control:rtsp://PSIA/Metadata/stream?format=xml-psia
a=rtpmap:99 XML-PSIA/0

```

The full SDP example showing the addition of video and audio tracks is as follows:

```

v=0
i=RTSP Session description
a=type:broadcast
c=IN IP4 232.0.1.112/255
m=video 5000 RTP/AVP 96
i=H.264 AVC VGA video stream
a=control:rtsp://PSIA/Streaming/channels/2
a=rtpmap:96 H264/90000
a=fmtp:96 packetization-mode=1;profile-level-
id=4D400C;sprop-parameter-sets
=J01ADKkYUI/LgDUGAQa2wrXvfaQ=,KN4JF6A=a=
b=AS:1200
a=framerate:30
a=framesize:96 640-480
m=audio 5001 RTP/AVP 97
i=Audio stream for video
a=sendonly
a=control:rtsp://PSIA/Streaming/channels/2/audio
a=rtpmap:97 G726-32/8000

```

```
m=metadata 5003 RTP/AVP 99
i=PSIA Metadata Stream using XML format
a=control:rtsp://PSIA/Metadata/stream?format=xml-psia
a=rtpmap:99 XML-PSIA/0
```

3.1.3 <AnalyticsAlert> Event Document format

The format of the XML for alerts differs from the XML format of the counts and status messages in that it provides more information. Additionally, a device may send a “basic” set of alert data or a “full” set of alert data depending on the client request. The basic set of data is required output for all analytics devices that support Alert output. The full set of information includes additional data about the object(s) involved in the event and any video images that might be included with the alert as snapshots. See the descriptions for each item in the full set to determine whether it is required or optional.

To specifically request the analytic alert data type, the URI is:

```
/PSIA/Metadata/stream?class=///Alert
```

By default, only the basic data is sent with alert streaming. To request the full data, the client appends the ‘output=full’ query parameter on the URI as shown in the examples below:

```
/PSIA/Metadata/stream?class=///Alert?output=full  
/PSIA/Metadata/stream?class=///Alert?channel=0&channel=2&output=full
```



Note that the order of the query parameters is not mandated so that the output=full could come before other parameters.

For the complete schema definition refer to Appendix A: Analytics EVENT Schema Definition (XSD).

Example	
	<pre><?xml version="1.0" encoding="utf-8"?> <AnalyticsAlert xmlns="urn:psialliance-org/analytics" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:psialliance-org/analytics psia-analytics- events-0.90.xsd" xmlns:xlink="http://www.w3.org/1999/xlink"> <MetadataHeader> <MetaVersion xmlns="">1.0</MetaVersion> <MetaID xmlns="">/psialliance.org/VideoAnalytics/Alert// 852a45d8-ecla-4fc1-9bc6-d41e305af681</MetaID> <MetaSourceID xmlns="">{bd33f5e9-41ba-44ac-95ff- 57a2452ad8bd}</MetaSourceID> <MetaSourceLocalID xmlns="">1</MetaSourceLocalID> <MetaTime xmlns="">2009-07-08T21:41:08.2428064Z</MetaTime> <MetaPriority xmlns="">5</MetaPriority> <MetaLink xmlns="">{a7d66a49-e419-41df-8696- 4882e260cec7}</MetaLink> <MetaHdrExtension xmlns=""> <ExtensionName>/psialliance.org/VideoAnalytics</ExtensionName> <AnalyticsViewNumber>1</AnalyticsViewNumber> <AnalyticsVideoSource> 32ad512e-2f51-4b4c-bfc3- 48249283bcbc</AnalyticsVideoSource> </MetaHdrExtension> </MetadataHeader> <RuleList> <Rule> <RuleID>c8759b2e-d4c1-46ff-87b4-e964d0d25427</RuleID> <RuleName>Perimeter Intrusion</RuleName> <RuleElementList> <RuleElement> <RuleType>line</RuleType> <Coordinates> <Point></pre>

```

        <X>0.5</X>
        <Y>0.5</Y>
    </Point>
    <Point>
        <X>0.4</X>
        <Y>0.4</Y>
    </Point>
</Coordinates>
</RuleElement>
</RuleElementList>
<Action>
    <Intrusion>Enter Area</Intrusion>
</Action>
<Duration>0</Duration>
</Rule>
</RuleList>
<Message>Person enters area.</Message>
<ObjectList>
    <Object>
        <ID>b4e2da9c-479b-43fd-88c7-6f028040a79b</ID>
        <BoundingBox>
            <X>0.5</X>
            <Y>0.5</Y>
            <Width>0.35</Width>
            <Height>0.35</Height>
        </BoundingBox>
        <CentroidVelocity>
            <X>0.7348</X>
            <Y>-0.3523</Y>
        </CentroidVelocity>
        <Classification>Human</Classification>
        <TimeOffset>0</TimeOffset>
    </Object>
    <Object>
        <ID>b4e2da9c-479b-43fd-88c7-6f028040a79b</ID>
        <BoundingBox>
            <X>0.55</X>
            <Y>0.55</Y>
            <Width>0.36</Width>
            <Height>0.36</Height>
        </BoundingBox>
        <CentroidVelocity>
            <X>0.7348</X>
            <Y>-0.3523</Y>
        </CentroidVelocity>
        <Classification>Human</Classification>
        <TimeOffset>100</TimeOffset>
    </Object>
</ObjectList>
<ObjectMotionList>
    <ObjectMotion>
        <ObjectID>b4e2da9c-479b-43fd-88c7-6f028040a79b</ObjectID>
        <CalibratedMotion>
            <Location>
                <Longitude>50</Longitude>
                <Latitude>-50</Latitude>
            </Location>
            <CalibratedSpeed>
                <Units>FeetPerSecond</Units>
                <Speed>10</Speed>
            </CalibratedSpeed>
            <CalibratedPath>
                <CalibratedPoint>
                    <Longitude>90</Longitude>
                    <Latitude>90</Latitude>
                </CalibratedPoint>
                <CalibratedPoint>
                    <Longitude>90.01</Longitude>
                    <Latitude>90</Latitude>
                </CalibratedPoint>
            </CalibratedPath>
        </CalibratedMotion>
    </ObjectMotion>
</ObjectMotionList>

```

```

</CalibratedMotion>
<PixelMotion>
  <Location>
    <X>0.75</X>
    <Y>0.75</Y>
  </Location>
  <Speed>20</Speed>
  <Path>
    <Point>
      <X>0.3</X>
      <Y>0.3</Y>
    </Point>
    <Point>
      <X>0.4</X>
      <Y>0.4</Y>
    </Point>
    <Point>
      <X>0.5</X>
      <Y>0.5</Y>
    </Point>
    <Point>
      <X>0.6</X>
      <Y>0.6</Y>
    </Point>
  </Path>
</PixelMotion>
</ObjectMotion>
</ObjectMotionList>

<Confidence>0</Confidence>
<Snapshots>
  <Snapshot xlink:type="simple" xlink:href="/images/alert001-
1.jpg">
    <TimeOffset>0</TimeOffset>
    <Width>320</Width>
    <Height>240</Height>
    <SizeInBytes>33123</SizeInBytes>
  </Snapshot>
  <Snapshot xlink:type="simple" xlink:href="/images/alert001-
2.jpg">
    <TimeOffset>100</TimeOffset>
    <Width>320</Width>
    <Height>240</Height>
    <SizeInBytes>32531</SizeInBytes>
  </Snapshot>
</Snapshots>
</AnalyticsAlert>

```

3.1.3.1 ANALYTICALERT ELEMENTS


Element Name	Requirement	Description
MetadataHeader	Required	The common PSIA Metadata header. The usage of fields as they relate to video analytics is listed below. Refer to the Common Event Proposal document for more information on all the fields. Note that the Analytics Specification adds additional fields into this header. These start with <i>Analytics</i> . This is required for both Basic and Full.
MetaID	Required	This URI defines the unique identifier of the Alert.
MetaSourceID	Required	The GUID of the analytics device that generated this alert.

Element Name	Requirement	Description
MetaSourceLocalID	Required	The channel number on the device that generated this alert.
MetaTime	Required	The time that the alert occurred. The time is formatted in the RFC 3339 format.
MetaPriority	Optional	The priority of the message as defined within the CMEM specification. Typical values will be 5, which is Notification, and 6 which is Information.
MetaLink	Optional	Provides a reference back to another event.
AnalyticsViewNumber	Optional	Integer value used with analytic software that can support multiple views for a single channel (i.e., for PTZ presets). If the analytics software only supports a single view, then either a default value of 0 is used or this element is not included. For an analytics channel that supports <i>n</i> views, the AnalyticsViewNumber will go from 0 to <i>n-1</i> .
AnalyticsVideoSource	Optional	GUID for the actual video source being analyzed. For example, if an NVR is taking in video from an IP camera stream and performs video analytics, the source video stream (VideoSource) has a unique identifier as does the NVR platform itself (Device) along with the specified MetaSourceLocalID associated with analyzing the video source.
RuleList	Required	List of rules that were triggered to cause this alert to be sent. The RuleList contains one or more Rules.
Rule	Required	This describes one of the rules that triggered this alert. This is a required element of RuleList.
RuleID	Required	GUID issued by the analytics platform to uniquely identify the rule that generated the event. This is a required element of Rule.
RuleName	Required	The name of the rule provided by the user during rule configuration and associated with the RuleID. This is a required element of Rule.
RuleElementList	Optional	A rule element is the term used to define the “physical” aspect of the rule, such a line or region defined by the user associated with the full rule. The RuleElementList consists of one or more RuleElement objects, which include the RuleType (<i>Line, Area, Full</i>) and the Coordinates that define the location of the rule element. This is an optional element of Rule. The recommendation is Basic Rules do not include this information to keep the data to a minimum.
RuleElement	Optional	

Element Name	Requirement	Description
Action	Optional	<p>Open text string for the analytics platform to include the action or behavior associated with the rule (e.g., crossed line <u>from left to right</u>, or object <u>loitered</u> in an area).</p> <p>Action information can also be retrieved outside of the event by the application retrieving the detailed rule information based on the unique RuleID. Providing the Action with the full event simply enables providing this information with no assumptions that the application keeps a stored list of rule information.</p> <p>Note: This may become a defined list of actions.</p> <p>This is an optional element of Rule. The recommendation is Basic Rules do not include this information to keep the data to a minimum.</p>
Duration	Optional	<p>Integer value in milliseconds.</p> <p>Used by any rule/event that requires the communication of a time duration. For example, analytic vendors may want to pass the "leave behind" time for a left bag event. Additionally, analytic platforms may output an ongoing string of events associated with the same rule or conditions.</p> <p>There will likely be a vendor-specific way in which the Duration field will be used.</p> <p>This is an optional element of Rule.</p>
Message	Required	<p>Open text string used to pass any information or message associated with the event as entered by the user during rule configuration.</p> <p>This is a required element for both Basic and Full although it may be left blank.</p>
ObjectList	Optional	<p>The ObjectList provides detailed information about each object(s) associated with an event.</p> <p>See the full definition of ObjectList in section 5 on Metadata (rather than repeating here).</p> <p>The only notable difference is that the TimeOffset field may be non-zero to represent that this object description was at some point in the past.</p> <p>This is an optional element for Full alert events.</p>
ObjectMotionList	Optional	<p>Contains vendor specific tracking information. This information defines the path and speed of the object being tracked. Information provided in this field can be with reference to a pixel of global/calibrated coordinate system</p>
ObjectMotion	Required element of ObjectMotionList	<p>Provides the information on motion of a specific Object.</p>
ObjectID	Required element of ObjectMotion	<p>The ID of the Object in the ObjectList that this motion pertains to.</p>
PixelMotion	Optional element of ObjectMotion.	<p>Describes the object motion in pixel coordinates normalized to the video resolution.</p> <p>Either this and/or CalibratedMotion must be defined for each ObjectMotion.</p>

Element Name	Requirement	Description
Location	Required element of PixelMotion	The location of the motion in normalized pixel coordinates.
Speed	Required element of PixelMotion	The speed in pixels that an object is moving.
Path	Optional element of PixelMotion	The path that the object has taken. The number of points is left up to the device.
CalibratedMotion	Optional element of ObjectMotion.	Describes the object motion in calibrated coordinates. Either this and/or PixelMotion must be defined for each ObjectMotion.
Location	Required element of CalibratedMotion	The location of the motion in latitude and longitude.
CalibratedSpeed	Required element of CalibratedMotion	The speed of the object based on the calibration.
Units	Required element of CalibratedSpeed	The unit of speed taking into account distance and time. The value can be one of the pre-defined types. These are currently: FeetPerSecond, MetersPerSecond, MilesPerHour and KilometersPerHour.
Speed	Required element of CalibratedSpeed	The actual speed of the object expressed in Units.
CalibratedPath	Optional element of CalibratedMotion	The path that the object has taken in longitude and latitude. The number of points is left up to the device.
Confidence	Optional	Float value between 0 and 1 issued by the analytics platform as to its confidence that the event was accurately generated (0 is least confident and 1 is most confident).
Snapshots	Optional	The list of snapshots that is relevant to the Alert. The TimeOffset field allows for a Snapshot from the past to be shown.
Snapshot	Optional	A snapshot for this event. If this is not an embedded image, the Snapshot contains an xlink attribute containing the URI to retrieve the actual snapshot.
TimeOffset	Required element of Snapshot	The time offset into the past from the time of the alert. A 0 indicates this snapshot is from the time of the alert (MetaTime). This is in milliseconds. This may be used in conjunction with the Object TimeOffset to view the object placement in the past.
Width	Optional element of Snapshot	The width of the Snapshot in pixels.
Height	Optional element of Snapshot	The height of the Snapshot in pixels.
SizeinBytes	Optional element of Snapshot	The size in bytes of the Snapshot.

Element Name	Requirement	Description
Expiration	Optional element of Snapshot	The time that the snapshot is available to retrieve for non-embedded images. This is listed in seconds and is based on the current Alert time. For embedded images, this is not sent. Note that the device may keep the Snapshot for longer than this duration.
Image	Optional element of Snapshot	The Base-64 encoded image. If this is present, the Snapshot does not have xlink attributes. An example is shown following this table.

 Note that the snapshot URI listed in the XLINK attribute may or may not be rooted under the /PSIA/Metadata path. It is up to the device to provide a meaningful path based on its configuration. Regardless of where the URI is rooted, the authorization permissions to retrieve these snapshots should, at a minimum, allow the same caller who retrieved the alert to retrieve the snapshot.

If the **AlertConfiguration** allows snapshots to be in-line, the **Snapshot** body will contain a Base64-encoded image instead of the XLINK attribute as shown below (only the **Snapshots** property is shown).

```
<Snapshots>
  <Snapshot>
    <TimeOffset>0</TimeOffset>
    <Image>fhdsj...base64encoded...hfkjhjs=</Image>
  </Snapshot>
  <Snapshot>
    <TimeOffset>100</TimeOffset>
    <Image>sfhue...base64encoded...djhsk5e</Image>
  </Snapshot>
</Snapshots>
```

Snapshots with markup, which typically includes bounding box around the target slice and rule definition drawn graphically as an overlay, are generated by an application and not the device. This allows an application full control over the markup from different devices for consistency. An application can create the markup for a given alert by using the target slice information as well as the rule definition itself.

3.1.4 <AnalyticsCount> Event Document Definition

Counting events are similar to alert events in that they also have a “basic” and “full” level of information. Retrieving the full count information from the event would use the same query parameter as defined above for alerts. For event push, this is configured with the event push settings.

The count event format supports different types of counting rules. In some cases, a count is nothing more than a tally, and the “value” for that individual tally is ‘1’. Other counting rules output specific values, such as the occupancy in a defined area, and the “value” associated with it is the actual occupancy. This difference is handled through the use of the CountType, Count, and Duration elements defined in the event XSD.

To specifically request the analytic count data type, the URI is:

```
/PSIA/Metadata/stream?class=//VideoAnalytics/Count
```

By default, only the basic data is sent with count streaming. To request the full data, the client appends the 'output=full' query parameter on the URI as shown in the examples below. To request count data from certain channels on a device, the 'channel=<#>' parameter is appended, per the examples below.

```
/PSIA/Metadata/stream?class=//VideoAnalytics/Count?output=full  
/PSIA/Metadata/stream?class=//VideoAnalytics/Count?channel=<#>&channel=<#>
```

An example of an AnalyticsCount with the full set of data is shown below. For the complete schema definition refer to Appendix A: Analytics EVENT Schema Definition (XSD).

Example	<pre><?xml version="1.0" encoding="utf-8"?> <AnalyticsCount xmlns="urn:psialliance-org/analytics" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:psialliance-org/analytics psia-analytics- events-0.90.xsd" xmlns:xlink="http://www.w3.org/1999/xlink"> <MetadataHeader> <MetaVersion xmlns="">1.0</MetaVersion> <MetaID xmlns="">/psialliance.org/VideoAnalytics/Count// e2c3c2e9-99a2-4f12-bb42-038577915bfa</MetaID> <MetaSourceID xmlns="">{bd33f5e9-41ba-44ac-95ff- 57a2452ad8bd}</MetaSourceID> <MetaSourceLocalID xmlns="">1</MetaSourceLocalID> <MetaTime xmlns="">2009-07-08T21:41:08.2428064Z</MetaTime> <MetaPriority xmlns="">5</MetaPriority> <MetaLink xmlns="">{a7d66a49-e419-41df-8696- 4882e260cec7}</MetaLink> <MetaHdrExtension xmlns=""> <ExtensionName>/psialliance.org/VideoAnalytics</ExtensionName> <AnalyticsViewNumber>1</AnalyticsViewNumber> <AnalyticsVideoSource> 32ad512e-2f51-4b4c-bfc3- 48249283bcbc</AnalyticsVideoSource> </MetaHdrExtension> </MetadataHeader> <RuleList> <Rule> <RuleID>1d4fbd2a-6147-40a5-abba-aa46a63fa467</RuleID> <RuleName>Tally Rule</RuleName> <RuleElementList> <RuleElement> <RuleType>line</RuleType> <Coordinates> <Point> <X>0.5</X> <Y>0.5</Y> </Point> <Point> <X>0.4</X> <Y>0.4</Y> </Point> </Coordinates> </RuleElement> </RuleElementList> <Action><None></None></Action> </Rule> </RuleList> </CountRuleType>tally</CountRuleType></pre>
---------	--

	<pre> <Count>10</Count> <Duration>0</Duration> <Confidence>0</Confidence> <Direction>forward</Direction> <Snapshots> <Snapshot xlink:type="simple" xlink:href="/images/count1.jpg"> <TimeOffset>0</TimeOffset> <Width>320</Width> <Height>240</Height> <SizeInBytes>33123</SizeInBytes> </Snapshot> </Snapshots> </AnalyticsCount> </pre>
--	---

3.1.4.1 ANALYTICSCOUNT ELEMENTS

Element Name	Requirement	Description
MetadataHeader	Required	The common PSIA Metadata header. The usage of fields as they relate to video analytics is listed below. Refer to the Common Event Proposal document for more information on all the fields. Note that the Analytics Specification adds additional fields into this header. These start with <i>Analytics</i> . This is required for both Basic and Full.
MetaID	Required	This URI defines the unique identifier of the count.
MetaSourceID	Required	The GUID of the analytics device that generated this count.
MetaSourceLocalID	Required	The channel number on the device that generated this count.
MetaTime	Required	The time that the count occurred. The time is formatted in the RFC 3339 format.
MetaPriority	Optional	The priority of the message as defined within the CMEM specification. Typical values will be 5, which is Notification, and 6 which is Information.
MetaLink	Optional	Provides a reference back to another event.
AnalyticsViewNumber	Optional	Integer value used with analytic software that can support multiple views for a single channel (i.e., for PTZ presets). If the analytics software only supports a single view, then either a default value of 0 is used or this element is not included. For an analytics channel that supports n views, the AnalyticsViewNo will go from 0 to $n-1$.
AnalyticsVideoSource	Optional	GUID for the actual video source being analyzed. For example, if an NVR is taking in video from an IP camera stream and performs video analytics, the source video stream (VideoSource) has a unique identifier as does the NVR platform itself (Device) along with the specified MetaSourceLocalID associated with analyzing that video source.
RuleList	Required	List of rules that were triggered to cause this count to be sent. The RuleList contains one or more Rule elements.

Element Name	Requirement	Description
Rule		This describes one of the rules that triggered this count. This is a required element of RuleList.
RuleID	Required	GUID issued by the analytics platform to uniquely identify the rule that generated the event. This is a required element of Rule.
RuleName	Required	The name of the rule provided by the user during rule configuration and associated with the RuleID. This is a required element of Rule.
RuleElementList	Optional	A rule element is the term used to define the “physical” aspect of the rule, such a line or region defined by the user associated with the full rule. The RuleElementList consists of one or more RuleElement objects, which include the RuleType (<i>Line, Area, Full</i>) and the Coordinates that define the location of the rule element. This is an optional element of Rule. The recommendation is Basic Rules do not include this information to keep the data to a minimum.
Action	Optional	Open text string for the analytics platform to include the action or behavior associated with the rule (e.g., crossed line <u>from left to right</u> , or object <u>loitered</u> in an area). Action information can also be retrieved outside of the event by the application retrieving the detailed rule information based on the unique RuleID. Providing the Action with the full event simply enables providing this information with no assumptions that the application keeps a stored list of rule information. Note: This may become a defined list of actions. This is an optional element of Rule. The recommendation is Basic Rules do not include this information to keep the data to a minimum.
Duration	Optional	Integer value in milliseconds. Used by any rule/event that requires the communication of a time duration. For example, platforms may output an ongoing string of events associated with the same rule or conditions. There will likely be a vendor-specific way in which the Duration field will be used. This is an optional element of Rule.

Element Name	Requirement	Description
CountRuleType	Required	<p>Defines the type of the count being generated. The enumerated types are tally, occupancy, and dwell.</p> <p>Tally – indicates the rule is counting individual events that are occurring. This dictates that the Count value will be a simple tally value of '1' when streaming individual count events or some other integer depending on how analytics systems may accumulate counts before outputting.</p> <p>Occupancy – indicates the rule is counting the number of objects in a defined area. This dictates that the Count value will represent a total number of objects in the monitored area.</p> <p>Dwell – indicates the rule is assessing the dwell time for objects within the defined area. In this case, the Count value will be '1' and the dwell values (time) will be passed in the Duration field.</p>
Count	Required	<p>Value containing a numerical value (float) that is the current count. The behavior of this value will be defined based on the specified CountRuleType.</p> <p>For Tally rules, this represents the number of counted events. For analytics that output a count event for each instance, this value will be '1'. For analytic platforms that may accumulate events and output a count event or "average count" at certain times or regular intervals, this value could be any integer or non-integer value.</p> <p>For Occupancy rules, this represents the occupancy derived by the analytics (whether actual integer value or average float value) at the time of outputting the event.</p> <p>For Dwell rules, the Count value will be set to '1'.</p>
Duration	Required	<p>This is a time in milliseconds. For a tally rule, this time will by default be 0. For dwell rules, this will be the total time in the area (i.e., dwell or wait time).</p>
Confidence	Optional	<p>Float value between 0 and 1 issued by the analytics platform as to its confidence that the event was accurately generated (0 is least confident and 1 is most confident).</p>
Direction	Optional	<p>Provides the direction for the count. Allowed values are forward or reverse.</p>
Snapshots	Optional	<p>The list of snapshots that is relevant to the Count. The TimeOffset field allows for a Snapshot from the past to be shown.</p>
Snapshot	Optional	<p>A snapshot for this event. If this is not an embedded image, the Snapshot contains an xlink attribute containing the URI to retrieve the actual snapshot.</p>
TimeOffset	Required element of Snapshot	<p>The time offset into the past from the time of the count. A 0 indicates this snapshot is from the time of the count (MetaTime). This is in milliseconds.</p> <p>This may be used in conjunction with the Object TimeOffset to view the object placement in the past.</p>

Element Name	Requirement	Description
Width	Optional element of Snapshot	The width of the Snapshot in pixels.
Height	Optional element of Snapshot	The height of the Snapshot in pixels.
SizeinBytes	Optional element of Snapshot	The size in bytes of the Snapshot.
Expiration	Optional element of Snapshot	The time that the snapshot is available to retrieve for non-embedded images. This is listed in seconds and is based on the current Alert time. For embedded images, this is not sent. Note that the device may keep the Snapshot for longer than this duration.
Image	Optional element of Snapshot	The Base-64 encoded image. If this is present, the Snapshot does not have xlink attributes.

4.1.3 <AnalyticsStatus> Event Document Definition

Status events provide vendor specific messages regarding the status of the analytic operation.

The status event format allows for analytic engines to provide status updates, including component status, heartbeats and errors, to the client application.

To specifically request the analytic status data type, the URI is:

```
/PSIA/Metadata/stream?class=/VideoAnalytics/Status
```

For the complete schema definition refer to Appendix A: Analytics EVENT Schema Definition (XSD).

An example of an AnalyticsStatus is shown below.

Example	
	<pre><?xml version="1.0" encoding="utf-8"?> <AnalyticsStatus xmlns="urn:psialliance-org/analytics" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:psialliance-org/analytics psia-analytics- events-0.90.xsd" xmlns:xlink="http://www.w3.org/1999/xlink"> <MetadataHeader> <MetaVersion xmlns="">1.0</MetaVersion> <MetaID xmlns="">/psialliance.org/VideoAnalytics/Status// e2c3c2e9-99a2-4f12-bb42-038577915bfa</MetaID> <MetaSourceID xmlns="">{bd33f5e9-41ba-44ac-95ff- 57a2452ad8bd}</MetaSourceID> <MetaSourceLocalID xmlns="">1</MetaSourceLocalID> <MetaTime xmlns="">2009-07-08T21:41:08.2428064Z</MetaTime> <MetaPriority xmlns="">5</MetaPriority> <MetaLink xmlns="">{a7d66a49-e419-41df-8696- 4882e260cec7}</MetaLink> <MetaHdrExtension xmlns=""> <ExtensionName>/psialliance.org/VideoAnalytics</ExtensionName> <AnalyticsViewNumber>1</AnalyticsViewNumber> <AnalyticsVideoSource></pre>

	<pre> 32ad512e-2f51-4b4c-bfc3- 48249283bcbc</AnalyticsVideoSource> </MetaHdrExtension> </MetadataHeader> <AnalyticsStatus>Processing</AnalyticsStatus> <Message>Rule "Occupancy Rule" Activated</Message> </AnalyticsStatus> </pre>
--	---

3.1.4.2 ANALYTICSSTATUS ATTRIBUTES

Element Name	Requirement	Description
MetadataHeader	Required	The common PSIA Metadata header. The usage of fields as they relate to video analytics is listed below. Refer to the Common Event Proposal document for more information on all the fields. Note that the Analytics Specification adds additional fields into this header. These start with <i>Analytics</i> .
MetaID	Required	This URI defines the unique identifier of the status event.
MetaSourceID	Required	The GUID of the analytics device that generated this event.
MetaSourceLocalID	Required	The channel number on the device that generated this event.
MetaTime	Required	The time that the event occurred. The time is formatted in the RFC 3339 format.
MetaPriority	Optional	The priority of the message as defined within the CMEM specification. Typical values will be 5, which is Notification, and 6 which is Information.
MetaLink	Optional	Provides a reference back to another event.
AnalyticsViewNumber	Optional	Integer value used with analytic software that can support multiple views for a single channel (i.e., for PTZ presets). If the analytics software only supports a single view, then either a default value of 0 is used or this element is not included. For an analytics channel that supports <i>n</i> views, the AnalyticsViewNo will go from 0 to <i>n-1</i> .
AnalyticsVideoSource	Optional	GUID for the actual video source being analyzed. For example, if an NVR is taking in video from an IP camera stream and performs video analytics, the source video stream (VideoSource) has a unique identifier as does the NVR platform itself (Device) along with the specified MetaSourceLocalID associated with analyzing that video source.
AnalyticsStatus	Required	Text string providing status of the analytics processing for the specified video channel
Message	Optional	Text string containing a status message with additional information that can be interpreted by the receiver.

3.1.4.3 EVENT STREAMING URI SUMMARY

Operation	Method	URI	XML Data Type
Retrieve a stream of events for the device	GET	/PSIA/MetaData/stream?class="//VideoAnalytics/Alert" Or /PSIA/MetaData/stream?class="//VideoAnalytics/Count" Or /PSIA/MetaData/stream?class="//VideoAnalytics/Status"	AnalyticsAlert or AnalyticsCount or AnalyticsStatus
Retrieve a stream of events for one or more channels on the device	GET	/PSIA/MetaData/stream?class=/VideoAnalytics/<AnalyticsType>&channel=[id]&channel=[id]...&channel=[id]	AnalyticsAlert or AnalyticsCount or AnalyticsStatus
Retrieve a stream of events for a single channel	GET	/PSIA/MetaData/stream?class=/VideoAnalytics/<AnalyticsType>&channel=[id]	AnalyticsAlert or AnalyticsCount or AnalyticsStatus

4 Metadata Streaming

The VAS includes support for devices to output a standard, structured set of video analytic metadata. The device communicates to the application its ability to output metadata based on its channel-specific analytics capabilities (see Section 2.7). For a device outputting metadata, specific data will be treated as required while other data is optional.

For Metadata streaming, the application requests the metadata from a device, one or more specified analytic channels, or for a specific object based on its Object ID. Additionally, the application can request a metadata stream of common fields and/or make specific requests for individual optional data types.

URI	/PSIA/Metadata/stream		Type	Resource
Requirement Level	- All -			
Function	CMEM defined REST resource/object that describes how to stream ALL video analytics metadata from a device. The documents returned conforms to the AnalyticsMetadata schemas defined in the subsequent sections, which contains a list of attributes that define the relevant information contained within the video analytics metadata.			
Methods	Query String(s)	Inbound Data	Return Result	
GET	class=//videoanalytics/Metadata&streamID=[ID]	None	<ResponseStatus> w/error code OR <MetaSessionParms>	
PUT	N/A	N/A	<ResponseStatus w/error code>	
POST	class=//videoanalytics/Metadata	<MetaSessionParms>	<ResponseStatus w/error code>	
DELETE	N/A	N/A	<ResponseStatus w/error code>	
Notes	<p>The Analytics specification allows additional query parameters to be sent to extend the functionality of the CMEM specification. These parameters are used to setup streams of different types and also to request data from specific channels on a device. Examples are as follows:</p> <pre>/PSIA/Metadata/stream?class=//VideoAnalytics/Metadata</pre> <pre>/PSIA/Metadata/stream?class=//VideoAnalytics/Metadata?Channel=[#]</pre> <p>The resulting XML document is validated using the analytics event schema (see Appendix A: Analytics EVENT Schema Definition (XSD))</p>			
XML Example	<pre><?xml version="1.0" encoding="utf-8"?> <AnalyticsMetadata xmlns="urn:psialliance-org/analytics" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:psialliance-org/analytics psia-analytics- events-0.90.xsd" xmlns:xlink="http://www.w3.org/1999/xlink"> <MetadataHeader> <MetaVersion xmlns="">1.0</MetaVersion> <MetaID xmlns="">/psialliance.org/VideoAnalytics/Count//</pre>			

```

e2c3c2e9-99a2-4f12-bb42-038577915bfa</MetaID>
<MetaSourceID xmlns="">{bd33f5e9-41ba-44ac-95ff-
57a2452ad8bd}</MetaSourceID>
<MetaSourceLocalID xmlns="">1</MetaSourceLocalID>
<MetaTime xmlns="">2009-07-08T21:41:08.2428064Z</MetaTime>
<MetaPriority xmlns="">6</MetaPriority>
<MetaLink xmlns="">{a7d66a49-e419-41df-8696-
4882e260cec7}</MetaLink>
<MetaHdrExtension xmlns="">
  <ExtensionName>
    /psialliance.org/VideoAnalytics
  </ExtensionName>
  <AnalyticsViewNumber>1</AnalyticsViewNumber>
  <AnalyticsVideoSource>
    32ad512e-2f51-4b4c-bfc3-48249283bcbc
  </AnalyticsVideoSource>
</MetaHdrExtension>
</MetadataHeader>
<ObjectList>
  <Object>
    <ID>b4e2da9c-479b-43fd-88c7-6f028040a79b</ID>
    <BoundingBox>
      <X>0.728125</X>
      <Y>0.9041667</Y>
      <Width>0.06875</Width>
      <Height>0.0541666</Height>
    </BoundingBox>
    <PolygonF>
      <PointF>
        <X>0.2637</X>
        <Y>0.9372</Y>
      </PointF>
      <PointF>
        <X>0.6239</X>
        <Y>0.3958</Y>
      </PointF>
      <PointF>
        <X>0.2637</X>
        <Y>0.3958</Y>
      </PointF>
    </PolygonF>
    <EllipseF>
      <X>0.748237</X>
      <Y>0.2354</Y>
      <Major>0.23456</Major>
      <Minor>0.12961</Minor>
      <Angle>2.37</Angle>
    </EllipseF>
    <Footprint>
      <X>0.7625</X>
      <Y>0.9041667</Y>
    </Footprint>
    <Centroid>
      <X>0.7625</X>
      <Y>0.93125</Y>
    </Centroid>
    <CentroidVelocity>
      <X>0.7348</X>
      <Y>-0.3523</Y>
    </CentroidVelocity>
    <Classification>Human</Classification>
    <Color>Red</Color>
    <Parents>
      <ParentID>
        b4e2da9c-479b-43fd-88c7-6f028040a79c
      </ParentID>
      <ParentID>
        b4e2da9c-479b-43fd-88c7-6f028040a79d
      </ParentID>
    </Parents>
    <ObjectMask>jfie75jf873h...jr73j8==</ObjectMask>

```

```

</Object>
<Object>
  <ID>b4e2da9c-479b-43fd-88c7-6f028040a79c</ID>
  <BoundingBox>
    <X>0.438125</X>
    <Y>0.8941667</Y>
    <Width>0.04875</Width>
    <Height>0.0741666</Height>
  </BoundingBox>
  <Footprint>
    <X>0.4625</X>
    <Y>0.8941667</Y>
  </Footprint>
  <Centroid>
    <X>0.4625</X>
    <Y>0.93125</Y>
  </Centroid>
  <CentroidVelocity>
    <X>0.7348</X>
    <Y>-0.3523</Y>
  </CentroidVelocity>
  <Classification>Vehicle</Classification>
  <Color>Black</Color>
  <StationaryData>
    <State>Inserted</State>
    <Type>Active</Type>
    <ParentID>
      b4e2da9c-479b-43fd-88c7-6f028040a79c
    </ParentID>
    <DetectionDelay>1.5</DetectionDelay>
  </StationaryData>
  <ObjectMask>jfie75jf873h...jr73j8==</ObjectMask>
</Object>
</ObjectList>
</AnalyticsMetadata>

```

4.1 Metadata Properties

Element Name	Requirement	Description
MetadataHeader	Required	The common PSIA Metadata header. The usage of fields as they relate to video analytics is listed below. Refer to the Common Event Proposal document for more information on all the fields. Note that the Analytics Specification adds additional fields into this header. These start with Analytics.
MetaID	Required	This URI defines the class type of this frames worth of analytics metadata.
MetaSourceID	Required	The GUID of the analytics device that generated this metadata.
MetaSourceLocalID	Required	The channel number on the device that generated this metadata.
MetaTime	Required	The time that the metadata was generated. The time is formatted in the RFC 3339 format.
MetaPriority	Required	The priority of the message as defined within the CMEM specification. Typical values will be 5, which is Notification, and 6 which is Information.
MetaLink	Optional	Provides a reference back to other metadata.

Element Name	Requirement	Description
<i>AnalyticsViewNumber</i>	Optional	Integer number used with analytic software that can support multiple views for a single channel (i.e., for PTZ presets). If the analytics software only supports a single view, then either a default value of 0 is used or this element is not included. For an analytics channel that supports n views, the ViewNo will go from 0 to n-1.
<i>AnalyticsVideoSource</i>	Optional	GUID for the actual video source being analyzed. For example, if an NVR is taking in video from an IP camera stream and performs video analytics, the source video stream (VideoSource) has a unique identifier as does the NVR platform itself (Device) along with the specified ChannelNo associated with analyzing that video source.
ObjectList	Required	The AnalyticsObjectList provides detailed information about each object(s) associated with an event.
<i>Object</i>	Required	The AnalyticsObject definition.
<i>ObjectID</i>	Required	Unique identifier for each target being tracked and reporting metadata. Each Object has an ObjectID, which may change across frames. To assist in tracking the history of an individual object, the ParentIDs may be used.
<i>BoundingBox</i>	Required	Provides data on the bounding box that outlines the position of an object. BoundingBox information is provided with an (x,y) coordinate defining the upper left corner of the box, and associated height and width values for the box. All this data is provided as normalized (0 to 1) data so it can be applied to any video input format.
<i>PolygonF</i>	Optional	Provides a polygon that bounds the position of an object. Polygon information is provided as a set of multiple x and y coordinates, each representing a point on the polygon. All this data is provided as normalized data so it can be applied to any video input format. Polygons have a minimum of 3 Points and are limited to a maximum of 64 points.
<i>EllipseF</i>	Optional	Provides an ellipse that bounds the position of an object. Ellipse information is provided with x and y coordinates defining the center of the ellipse, and associated major and minor axis lengths. All this data is provided as normalized data so it can be applied to any video input format. Additionally, Ellipse information includes an Angle represented in radians as measured by the angle between the X axis and the major axis of the ellipse.
<i>Footprint</i>	Optional	The Footprint is where the analytics library estimates the bottom of the target is. The Footprint data is provided as a normalized point so it can be applied to any video input format.
<i>Centroid</i>	Optional	The Centroid is where the analytics library estimates the target's center of mass is. The Centroid data is provided as a normalized (0 to 1) coordinate point so it can be applied to any video input format.

Element Name	Requirement	Description
<i>CentroidVelocity</i>	Optional	Pixel-based velocity of the target expressed in x and y velocity vector components. All this data is provided as normalized (-1 to 1) data so it can be applied to any video input format.
<i>Classification</i>	Optional	Open text string to provide a classification label for the object (e.g., human, vehicle, etc.). Due to the variations in classification types across analytic platforms, this is an open field for the analytics to output any form of classification. In the future, this may become an identified list of known classification types.
<i>Color</i>	Optional	Open text string to provide the predominant color associated with the Object. It is up to the analytics vendor on how it intends to implement this optional piece of Object metadata, and how its analytics platform determines the dominant color. In the future, this may become an identified list of known colors or implement some other form of standard color list.
<i>TimeOffset</i>	Optional	This is the number of milliseconds from the AnalyticsMetadata's timestamp. In most cases this will be 0 to represent the object's location at the time of the AnalyticsMetadata. When the ObjectList is used within a different context, this maybe non-zero. If this is not specified, than a zero can be assumed.
<i>Parents</i>	Optional	The list of ParentIDs that this object has.
<i>ParentID</i>	Optional	The ParentID property allows tracking of the same target across different generations. The Parent ID is a list of one or more ObjectIDs that the current target was an immediate descendant of.
<i>StationaryData</i>	Optional	Set of Object information for objects that are or become stationary in the camera view.
<i>State</i>	Optional	The State is either Inserted or Removed. The Inserted state is when an object that was moving stops. The Removed state is when an object starts to move again (e.g. a parked car that moves) or is removed from the scene (e.g. a bag removed).
<i>Type</i>	Optional	The Type describes whether this target was an active target (such as a vehicle or person) or a passive target (such as a bag). The valid values are Active and Passive.
<i>ParentID</i>	Optional	The ParentID contains the ID of the object that left the stationary target. If a parent cannot be determined this will be the same as the TargetID.
<i>DetectionDelay</i>	Optional	DetectionDelay is the delay in seconds from when the target first stopped moving to when it was flagged as Stationary.
<i>ObjectMask</i>	Optional	Base64 encoded bit mask area representing the pixels that are included as part of the object within the bounding box.

4.2 Metadata Streaming Operations

Operation	Method	URI	XML Data Type
Retrieve a stream of metadata for the device	GET	/PSIA/Metadata/stream?class=//VideoAnalytics/Metadata	AnalyticsMetadata
Retrieve a stream of metadata for one or more channels on the device	GET	/PSIA/Metadata/stream?class=//VideoAnalytics/Metadata&channel=[id]&channel=[id]	AnalyticsMetadata
Retrieve a stream of metadata for a single channel	GET	/PSIA/Metadata/stream?class=//VideoAnalytics/Metadata&channel=[id]	AnalyticsMetadata
Retrieve metadata for a specified object	GET	/PSIA/MetaData/stream?class=//VideoAnalytics/Metadata&object=[id]	AnalyticsMetadata

5 Appendix A: Analytics EVENT Schema Definition (XSD)

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:psialliance-org/analytics" elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:cmem="urn:psialliance-org"
  xmlns="urn:psialliance-org/analytics" xmlns:xlink="http://www.w3.org/1999/xlink"
  version="1.0">
  <xs:import namespace="urn:psialliance-org" schemaLocation="metaHeader.xsd"/>
  <xs:import namespace="http://www.w3.org/1999/xlink" schemaLocation="xlink.xsd"/>
  <xs:annotation>
    <xs:documentation xml:lang="en">
      -- 03/26/2009 0.1 JC - Initial revision
      -- 04/13/2009 0.2 JC - Incorporated Initial group feedback
      - Changed ID and Reference ID to type GUID
      - Moved ID and Timestamp to EventHeader for each event
      type
      - Added minOccurs attribute to EventHeader
      - Added definition of the Priority SimpleType
      - Changed ReferenceID minOccurs to unbounded
      -- 04/15/2009 0.3 JC - Incorporated changes from 04/14/09 meeting
      - Created a base Event type from which all analytics
      events are derived
      - Allowed the Action type to be free format as
      defined by the application
      - Removed all completed TODO items
      -- 05/07/2009 0.4 JC - Removed Summary from Event type
      - Standardized naming convention for Events,
      removed Event Suffix
      - All enumeration values changed to lower case
      instead of mixed case
      - Changed RuleID from TOKEN to GUID
      - Removed Info suffix, all occurrences
      - Removed EventTypes since schema elements denote type
      - Added PixelPoint type providing a normalized
      pixel coordinate
      - Created a Calibrated Motion Vector and allowed
      both Pixel and Calibrated
      - Simplified AnalyticsStatus event
      -- 05/28/2009 0.5 JC - Changed AnalyticStatus to AnalyticsStatus
      - Changed ObjectID from xs:token to GUID
      - Fixed type changing CalibratedPath to CalibrationPoint
      in CalibratedPath type
      -- 06/24/2009 0.6 JC - Changed CalibratedPoint to use decimal Geospatial
      coordinates
      - Removed "{}" from GUID validation
      - Changed Calibrated points and path to CalibratedPoint
      and CalibratedPath respectively
      -- 12/13/2009 0.73 JC - Moved Motion outside of the Object model, now
      referenced via ObjectID
      -- 02/09/2010 0.74 JC - Validated Schema and Merged with CMEM specification
      -- 02/12/2010 0.77 GM - Updated based on the .77 version of the spec.
      Removed PixelPoint, Pixel, etc to simplify the
      types.
      - Reworked hierarchy based on the latest spec.
      -- 06/22/2010 0.80 JC - Updated the specification to conform to the review
      version of the specification.
      -- 09/08/2010 1.0 JC - Updated based on review comments, ready for release

    </xs:documentation>
  </xs:annotation>

  <!-- Analytics event -->
  <xs:complexType name="AnalyticsEvent" abstract="true">
    <xs:annotation>
      <xs:documentation xml:lang="en">
        Defines the basic analytic event type, there will be
```

three classes of events, Alerts, Counts and Status. All will have the same basic structure but slightly different header information

```
</xs:documentation>
</xs:annotation>
<xs:sequence>
  <xs:element name="MetadataHeader" minOccurs="1" maxOccurs="1"
             type="MetaDataHeader"/>
</xs:sequence>
</xs:complexType>

<xs:element name="AnalyticsAlert" type="AnalyticsAlert"/>
<xs:complexType name="AnalyticsAlert">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Header for Alert event types
    </xs:documentation>
  </xs:annotation>
  <xs:complexContent mixed="false">
    <xs:extension base="AnalyticsEvent">
      <xs:sequence>
        <xs:element name="RuleList" minOccurs="1" maxOccurs="1"
                   type="RuleSynopsisList"/>
        <xs:element name="Message" minOccurs="1" maxOccurs="1"
                   type="xs:string"/>
        <xs:element name="ObjectList" minOccurs="0" maxOccurs="1"
                   type="AnalyticsObjectList"/>
        <xs:element name="ObjectMotionList" minOccurs="0" maxOccurs="1"
                   type="AnalyticsObjectMotionList"/>
        <xs:element name="Confidence" minOccurs="0" maxOccurs="1"
                   type="Confidence"/>
        <xs:element name="Snapshots" minOccurs="0" maxOccurs="1"
                   type="Snapshots"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:element name="AnalyticsCount" type="AnalyticsCount"/>
<xs:complexType name="AnalyticsCount">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Header for count event types
    </xs:documentation>
  </xs:annotation>
  <xs:complexContent mixed="false">
    <xs:extension base="AnalyticsEvent">
      <xs:sequence>
        <xs:element name="RuleList" minOccurs="1" maxOccurs="1"
                   type="RuleSynopsisList"/>
        <xs:element name="CountRuleType" minOccurs="1" maxOccurs="1"
                   type="CountType"/>
        <xs:element name="Count" minOccurs="1" maxOccurs="1"
                   type="xs:float"/>
        <xs:element name="Duration" minOccurs="1" maxOccurs="1"
                   type="xs:int"/>
        <xs:element name="Confidence" minOccurs="0" maxOccurs="1"
                   type="Confidence"/>
        <xs:element name="Direction" minOccurs="0" maxOccurs="1"
                   type="Direction"/>
        <xs:element name="Snapshots" minOccurs="0" maxOccurs="1"
                   type="Snapshots"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:element name="AnalyticsStatus" type="AnalyticsStatus"/>
<xs:complexType name="AnalyticsStatus">
```

```

<xs:annotation>
  <xs:documentation xml:lang="en">
    Header for status event types
  </xs:documentation>
</xs:annotation>
<xs:complexContent mixed="false">
  <xs:extension base="AnalyticsEvent">
    <xs:sequence>
      <xs:element name="AnalyticsStatus" minOccurs="1" type="xs:string"/>
      <xs:element name="Message" minOccurs="0" maxOccurs="1"
        type="xs:string"/>
    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>

<!-- Element types defined below -->
<xs:complexType name="MetaDataHeader">
  <xs:complexContent mixed="false">
    <xs:extension base="cmem:metaHeader">
      <xs:sequence/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:complexType name="RuleSynopsisList">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Defines a list of rules that have been matched
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Rule" minOccurs="1" maxOccurs="unbounded"
      type="RuleSynopsis"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="RuleSynopsis">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Defines the rule information to be passed with the event
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="RuleID" minOccurs="1" maxOccurs="1" type="GUID"/>
    <xs:element name="RuleName" minOccurs="1" maxOccurs="1" type="xs:string"/>
    <xs:element name="RuleElementList" minOccurs="0" maxOccurs="1"
      type="RuleElementList"/>
    <xs:element name="Action" minOccurs="0" maxOccurs="1" type="Action"/>
    <xs:element name="Duration" minOccurs="0" maxOccurs="1" type="xs:int"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="RuleElementList">
  <xs:sequence>
    <xs:element name="RuleElement" minOccurs="1" maxOccurs="unbounded"
      type="RuleElement"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="RuleElement">
  <xs:sequence>
    <xs:element name="RuleType" minOccurs="1" maxOccurs="1" type="RuleType"/>
    <xs:element name="Coordinates" minOccurs="0" maxOccurs="1" type="PointList"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="Action">
  <xs:sequence>
    <xs:any processContents="lax"/>
  </xs:sequence>

```

```

</xs:complexType>

<xs:complexType name="AnalyticsObjectMotionList">
  <xs:sequence>
    <xs:element name="ObjectMotion" minOccurs="0" maxOccurs="unbounded"
      type="AnalyticsObjectMotion"/>
  </xs:sequence>
</xs:complexType>

<xs:group name="PixelMotionGroup">
  <xs:sequence>
    <xs:element name="PixelMotion" minOccurs="1" maxOccurs="1"
      type="PixelMotion"/>
  </xs:sequence>
</xs:group>

<xs:group name="CalibratedMotionGroup">
  <xs:sequence>
    <xs:element name="CalibratedMotion" minOccurs="1" maxOccurs="1"
      type="CalibratedMotion"/>
    <xs:element name="PixelMotion" minOccurs="1" maxOccurs="1"
      type="PixelMotion"/>
  </xs:sequence>
</xs:group>

<xs:complexType name="AnalyticsObjectMotion">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Defines a motion vector as a set of points. PixelMotion and/or
      CalibratedMotion must be defined.
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="ObjectID" type="GUID"/>
    <!-- Mandate that at least one is chosen or both -->
    <xs:choice>
      <xs:group ref="PixelMotionGroup"/>
      <xs:group ref="CalibratedMotionGroup"/>
    </xs:choice>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="PixelMotion">
  <xs:sequence>
    <xs:element name="Location" type="PointF"/>
    <xs:element name="Speed" type="xs:float"/>
    <xs:element name="Path" minOccurs="1" maxOccurs="1" type="PointList"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="CalibratedMotion">
  <xs:sequence>
    <xs:element name="Location" type="CalibratedPoint"/>
    <xs:element name="CalibratedSpeed" type="CalibratedSpeed"/>
    <xs:element name="CalibratedPath" minOccurs="1" maxOccurs="1"
      type="CalibratedPath"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="CalibratedPoint">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Defines a single X/Y point with long/lat.
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Longitude" type="Longitude"/>
    <xs:element name="Latitude" type="Latitude"/>
  </xs:sequence>
</xs:complexType>

```

```

<xs:complexType name="CalibratedSpeed">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Calibrated speed that has both the units of measurement and speed
      Value
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Units" type="CalibratedUnits"/>
    <xs:element name="Speed" type="xs:float"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="CalibratedPath">
  <xs:sequence>
    <xs:element name="CalibratedPoint" minOccurs="2" maxOccurs="unbounded"
      type="CalibratedPoint"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="Snapshots">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="Snapshot"
      nillable="true" type="Snapshot" />
  </xs:sequence>
</xs:complexType>

<xs:complexType name="Snapshot">
  <xs:sequence>
    <xs:element name="TimeOffset" minOccurs="1" maxOccurs="1" type="xs:int"/>
    <xs:element name="Width" minOccurs="0" maxOccurs="1" type="xs:int" />
    <xs:element name="Height" minOccurs="0" maxOccurs="1" type="xs:int"/>
    <xs:element name="SizeInBytes" minOccurs="0" maxOccurs="1" type="xs:int" />
    <xs:element name="Image" minOccurs="0" maxOccurs="unbounded" nillable="true"
      type="xs:base64Binary" />
  </xs:sequence>
  <xs:attributeGroup ref="xlinkAttributes"/>
</xs:complexType>

<xs:element name="AnalyticsMetadata" nillable="true" type="AnalyticsMetadata" />
<xs:complexType name="AnalyticsMetadata">
  <xs:sequence>
    <xs:element name="MetadataHeader" minOccurs="1" maxOccurs="1"
      type="MetaDataSourceHeader"/>
    <xs:element name="ObjectList" minOccurs="0" maxOccurs="1"
      type="AnalyticsObjectList" />
  </xs:sequence>
</xs:complexType>

<xs:element name="ObjectList" nillable="true" type="AnalyticsObjectList" />
<xs:complexType name="AnalyticsObjectList">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="Object" nillable="true"
      type="AnalyticsObject" />
  </xs:sequence>
</xs:complexType>

<xs:element name="AnalyticsObject" nillable="true" type="AnalyticsObject" />
<xs:complexType name="AnalyticsObject">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="ID" type="GUID"/>
    <xs:element minOccurs="1" maxOccurs="1" name="BoundingBox" type="RectF" />
    <xs:element minOccurs="0" maxOccurs="1" name="Polygon" type="PolygonF" />
    <xs:element minOccurs="0" maxOccurs="1" name="Ellipse" type="EllipseF" />
    <xs:element minOccurs="0" maxOccurs="1" name="Footprint" type="PointF" />
    <xs:element minOccurs="0" maxOccurs="1" name="Centroid" type="PointF" />
    <xs:element minOccurs="0" maxOccurs="1" name="CentroidVelocity" type="PointF"
  />

    <xs:element minOccurs="0" maxOccurs="1" name="Classification"
      type="xs:string" />
    <xs:element minOccurs="0" maxOccurs="1" name="TimeOffset" type="xs:int"/>
  </xs:sequence>
</xs:complexType>

```

```

        <xs:element minOccurs="0" maxOccurs="1" name="Parents" type="ArrayOfParents"
/>
        <xs:element minOccurs="0" maxOccurs="1" name="ObjectMask"
            type="xs:base64Binary" />
        <xs:element minOccurs="0" maxOccurs="1" name="Color" type="xs:token"/>
        <xs:element minOccurs="0" maxOccurs="1" name="StationaryData"
type="StationaryData"/>
    </xs:sequence>
</xs:complexType>

<xs:element name="ArrayOfParents" nillable="true" type="ArrayOfParents" />
<xs:complexType name="ArrayOfParents">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="ParentID"
            nillable="true" type="xs:token" />
    </xs:sequence>
</xs:complexType>

<xs:element name="StationaryData" nillable="true" type="StationaryData" />
<xs:complexType name="StationaryData">
    <xs:sequence>

        <xs:element name="State" type="StationaryStates" />
        <xs:element name="Type" type="StationaryTypes" />

        <xs:element name="ParentID" type="xs:token" />
        <xs:element name="DetectionDelay" type="xs:float" />
    </xs:sequence>
</xs:complexType>

<!--
    All simple types, enumerations and value restrictions
-->
<xs:simpleType name="CountType">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Defines the list of available count types
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="tally"/>
        <xs:enumeration value="occupancy"/>
        <xs:enumeration value="dwell"/>
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="Direction">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Defines the list of applicable directions for counting
            rules
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="forward"/>
        <xs:enumeration value="reverse"/>
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="CalibratedUnits">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Defines the list of applicable directions for counting
            rules
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="FeetPerSecond"/>
        <xs:enumeration value="MetersPerSecond"/>
    </xs:restriction>
</xs:simpleType>

```

```

        <xs:enumeration value="MilesPerHour" />
        <xs:enumeration value="KilometersPerHour" />
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="Confidence">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Defines Confidence as a normalized float value
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:float">
        <xs:minInclusive value="0" />
        <xs:maxInclusive value="1" />
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="Longitude">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Defines the range of decimal Longitude values
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:float">
        <xs:minInclusive value="-180" />
        <xs:maxInclusive value="180" />
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="Latitude">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Defines the range of decimal Latitude values
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:float">
        <xs:minInclusive value="-90" />
        <xs:maxInclusive value="90" />
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="GUID">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            The representation of a GUID, generally the id of an
            element.
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:pattern
            value="[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-
F0-9]{12}"
            />
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="RuleType">
    <xs:annotation>
        <xs:documentation xml:lang="en">
            Definitions of the available analytic rule types.
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="line" />
        <xs:enumeration value="area" />
        <xs:enumeration value="full" />
    </xs:restriction>
</xs:simpleType>

<xs:complexType name="PointF">

```

```

<xs:annotation>
  <xs:documentation xml:lang="en">
    Defines a single X/Y point in normalized form.</xs:documentation>
  </xs:annotation>
</xs:sequence>
  <xs:element name="X" type="xs:float"/>
  <xs:element name="Y" type="xs:float"/>
</xs:sequence>
</xs:complexType>

<xs:complexType name="RectF">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Defines a rectangle in normalized form.
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="X" type="xs:float"/>
    <xs:element name="Y" type="xs:float"/>
    <xs:element name="Width" type="xs:float"/>
    <xs:element name="Height" type="xs:float"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="PolygonF">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Defines a closed shape that consist of a set of 3 or more points in
      normalized form.
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Point" minOccurs="3" maxOccurs="64" type="PointF"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="EllipseF">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      Defines the ellipse in normalized form.
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="X" type="xs:float"/>
    <xs:element name="Y" type="xs:float"/>
    <xs:element name="Major" type="xs:float"/>
    <xs:element name="Minor" type="xs:float"/>
    <xs:element name="Angle" type="xs:float"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="PointList">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      A list of 2 or more points.
    </xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Point" minOccurs="2" maxOccurs="unbounded" type="PointF"/>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="StationaryStates">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      The state of the stationary object, inserted or removed
    </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="inserted"/>
    <xs:enumeration value="removed"/>
  </xs:restriction>
</xs:simpleType>

```

```
</xs:restriction>
</xs:simpleType>

<xs:simpleType name="StationaryTypes">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      The type of the stationary object, active or passive. Active objects
      Were moving and then stopped, passive are objects that are static. i.e
      Vehicle and bag respectively.
    </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="active"/>
    <xs:enumeration value="passive"/>
  </xs:restriction>
</xs:simpleType>

<xs:attributeGroup name="xlinkAttributes">
  <xs:attribute ref="xlink:type"/>
  <xs:attribute ref="xlink:href"/>
</xs:attributeGroup>
</xs:schema>
```

6 Appendix B: References

6.1 REST

http://en.wikipedia.org/wiki/Representational_State_Transfer
<http://www.xml.com/pub/a/2004/12/01/restful-web.html>

6.2 HTTP

<http://www.w3.org/Protocols/rfc2616/rfc2616.html>
<http://www.ietf.org/rfc/rfc2617.txt>
<http://www.ietf.org/rfc/rfc3986.txt>

6.3 MIME

<http://tools.ietf.org/html/rfc2045>
<http://tools.ietf.org/html/rfc2046>
<http://tools.ietf.org/html/rfc2046>

6.4 XML

<http://www.w3.org/TR/xml>
<http://www.w3.org/TR/xlink>

6.5 Data Encodings

<http://www.ietf.org/rfc/rfc3548.txt>

6.6 Time Format

<http://www.ietf.org/rfc/rfc3339.txt>

6.7 SHA-1 Hashing Algorithm

<http://www.ietf.org/rfc/rfc3174.txt>

6.8 ZLIB Compression

<http://www.ietf.org/rfc/rfc1950.txt>

6.9 DNS-SD

<http://www.dns-sd.org/>
<http://developer.apple.com/networking/bonjour/>
<http://files.dns-sd.org/draft-cheshire-dnsext-dns-sd.txt>
<http://www.dns-sd.org/ServiceTypes.html>