INFRASTRUCTURES

CITYBOX

Using SLV API Rest (North)

Eclairage Public

Object :

The object of this document is to describe each function of the SLV API and how to use them.

Document State :

Author	Version	Date	Update object	Validation	Date
G.DEFOUR	1.0	03/08/2014	Creation	G.DEFOUR	

Contents Page

1. Introduction

1.1. Overview table

Part	Page	Fonction	Category	Function
2.1.	14	AknowledgeAlarm	Alarm	Acknowledge the given alarm.
2.2.	15	AknowledgeAlarms	Alarm	Acknowledge the given alarms.
2.3.	17	CreateSendEmailActionAlarmDefinition	Alarm	Create a send email alarm definition.
2.4.	19	CreateSingleActionAlarmDefinition	Alarm	Create a single action alarm definition.
2.5.	21	DeleteAlarmDefinition	Alarm	Delete the given alarm definition.
2.6.	22	GetAlarmDefinition	Alarm	Return an alarm definition.
2.7.	26	GetAlarmDefinitionEditablePropertyDescriptors	Alarm	Return the editable property descriptors for the given alarm definition.
2.8.	34	GetAllAlarmDefinitions	Alarm	Return an array of alarm definitions.
2.9.	38	GetGeoZoneAlarms	Alarm	Return an array of alarms for the given geozone.
2.10.	40	GetRegisteredAlarmStateChangeActionImplementations	Alarm	Return an array of alarm state change actions.
2.11.	41	GetRegisteredTriggerConditionImplementations	Alarm	Return an array of trigger conditions.
2.12.	44	GetVirtualSingleActionAlarmDefinitionEditablePropertyDescritors	Alarm	Return an array of editable property descriptors for a virtual single action alarm definition.
2.13.	50	UpdateAlarmDefinition	Alarm	Update the given alarm definition.
3.1.	53	ExecuteCommand	Commands	Excute a command on the given device.
3.2.	54	ExecuteCommands	Commands	Execute a command on the given devices.
3.3.	55	ExitManualModes	Commands	Exit manual mode for the given streetlights.
3.4.	57	GetDeviceCommand	Commands	Return the given command for the given device.

3.5.	58	GetDeviceCommands	Commands	Return an array of commands for the given device.
3.6.	60	GetDeviceExecutable Commands	Commands	Return an array of executable commands for the given device.
3.7.	61	GetDeviceFailureDescriptors	Commands	Return an array of failure descriptors for the given device.
3.8.	64	GetDeviceFailureOnValues	Commands	Return the failure state of each device for the given failure names.
3.9.	64	GetDeviceFormattedMeteringValues	Commands	Return an array of metering values for the given device.
3.10.	66	GetDeviceMeteringValueDescriptors	Commands	Return an array of metering value descriptors for the given device.
3.11.	67	GetDevicesFailureOnValues	Commands	Return the failure state of each device for the given failure names.
3.12.	68	GetDevicesHasOneFailureOn	Commands	Return the failure state for each device.
3.13.	69	GetDimmingLevel	Commands	Return the dimming level for the given streetlight.
3.14.	70	GetDimmingLevels	Commands	Return an array of dimming levels for the given streetlights.
3.15.	71	GetDimmingMode	Commands	Return an array of dimming modes for the given streetlights.
3.16.	72	GetGeoZoneFailures	Commands	Return an array of failures for the given geozone.
3.17.	74	HasDeviceCommand	Commands	Check out that the given device has the given command.
3.18.	75	ReplaceLamps	Commands	Replace the lamps for the given streetlights.
3.19.	76	ReplaceOLCs	Commands	Replace the given streetlights.
3.20.	76	SetDevicesMeteringValues	Commands	Update the metering values for the given devices.
3.21.	77	SetDimmingLevel	Commands	Update the dimming level for the given streetlight.
3.22.	78	SetDimmingLevels	Commands	Update the dimming levels for the given streetlights.
3.23.	80	SwitchOff	Commands	Switch off the given streetlight.
3.24.	81	SwitchOn	Commands	Switch on the given streetlight.

4.1.	82	CommissionControllerAsync	Controller	Commission a controller asynchronously (only compatible with starsense RF).
4.2.	83	CreateController	Controller	Create a new controller.
4.3.	86	ExitOuputManualMode	Controller	Exit manual mode for an output of the given controller.
4.4.	86	GetAllControllers	Controller	Return an array of all controllers.
4.5.	90	GetControllerDeviceDevice	Controller	Return the given controller as device.
4.6.	90	GetCumulatedDatalogSize	Controller	Return the datalog size for the given controller between date range.
4.7.	91	GetOutputIndicesControllingMain	Controller	Return an array of output indices controlling main.
4.8.	92	GetOutputIndicesControllingMains	Controller	Return arrays of output indices controlling mains.
4.9.	94	GetSystemTimeStringInLocalTime	Controller	Return the system time as local time for the given controller.
4.10.	95	GetTimeZone	Controller	Return the timezone for the given controller.
4.11.	96	GetTimeZoneId	Controller	Return the timezone identifier for the given controller.
4.12.	97	ReadControllerInputState	Controller	Return the state of an input on the given controller.
4.13.	98	ReadControllerOutputState	Controller	Return the state of an output on the given controller.
4.14.	99	SendControllerDataNow	Controller	Tell to the given controller to send its datalogs.
4.15.	99	SetSystemTime	Controller	Update the system time for the given controller.
4.16.	100	SetSystemTimeStringInLocalTime	Controller	Update the system time for the given controller.
4.17.	101	SetTimeZoneId	Controller	Update the timezone for the given controller.
4.18.	102	SwitchControllerOutput	Controller	Switch an output of the given controller.
5.1.	104	CreateBrand	Inventory	Create a new brand.
5.2.	105	CreateCategoryDevice	Inventory	Create a new device.
5.3.	107	CreateGeoZone	Inventory	Create a new geozone.

5.4.	109	CreateProvider	Inventory	Create a new provider.
5.5.	110	DeleteBrand	Inventory	Delete a brand.
5.6.	111	DeleteDevices	Inventory	Delete devices.
5.7.	112	DeleteGeoZone	Inventory	Delete a geozone.
5.8.	113	DeleteProvider	Inventory	Delete a provider.
5.9.	114	GetAllBrands	Inventory	Return an array of all brands.
5.10.	116	GetAllDevicesValueDescriptors	Inventory	Return an array of device value descriptors.
5.11.	119	GetAllLampTypes	Inventory	Return an array of all lamp types.
5.12.	121	GetAllProviders	Inventory	Return an array of all providers.
5.13.	122	GetAllValueDescriptors	Inventory	Return an array of value descriptors.
5.14.	125	GetBatchResult	Inventory	Return the batch result for the given batch identifier.
5.15.	127	GetCategories	Inventory	Return an array of device categories.
5.16.	128	GetCategoryControllerDevices	Inventory	Return an array of devices for the given controller and device category.
5.17.	131	GetControllerDevices	Inventory	Return an array of all devices for the given controller.
5.18.	134	GetControllerDevicesLastValuesAsArray	Inventory	Return an array of device last values for the given controller.
5.19.	135	GetDefaultValueDescriptors	Inventory	Return an array of default value descriptors for the given config file path.
5.20.	136	GetDevice	Inventory	Return the device for the given device identifier.
5.21.	139	GetDevicesInBounds	Inventory	Return all the devices located inside a box defined by four points as follow:
5.22.	142	GetDevicesLastValuesAsArray	Inventory	Return an array of last values for the given devices.
5.23.	143	GetDevicesLogValuesAsArray	Inventory	Return an array of log values for the given devices between date range. The log values are the last values stored in the Streetlight.Vision database.
5.24.	143	GetDeviceValueDescriptors	Inventory	Return an array of value descriptors for the given device.

5.25.	145	GetDistinctExistingValues	Inventory	Return the distinct existing values of the given meaning.
5.26.	148	GetGeoZoneChildrenGeoZones	Inventory	Return an array of all sub geozones for the given geozone.
5.27.	150	GetGeoZoneDevices	Inventory	Return an array of devices for the given geozone.
5.28.	153	GetGeoZoneDevicesLastValuesAsArray	Inventory	Return an array of device last values for the given geozone.
5.29.	153	GetGeoZoneRoot	Inventory	Return root geozone for the current logged user.
5.30.	155	GetGeoZoneValueDescriptors	Inventory	Return the value descriptors of the given geozone.
5.31.	165	GetVirtualDeviceValueDescriptors	Inventory	Return an array of device value descriptors.
5.32.	167	ImportDevicesFromCsvFileAsync	Inventory	Import devices from a CSV file.
5.33.	168	ImportDevicesFromSdpFileAsync	Inventory	Import devices from a SDP file.
5.34.	170	MoveDevices	Inventory	Move location of devices on the map.
5.35.	172	RenameDevices	Inventory	Rename devices.
5.36.	173	SearchDevices	Inventory	Return all devices having their name containing the freeform parameter.
5.37.	176	SearchGeoZones	Inventory	Return an array of geozones.
5.38.	178	SetDevicesGeoZone	Inventory	Move the given devices into the target geozone.
5.39.	180	SetDevicesValues	Inventory	Update the device values.
5.40.	181	SetDeviceValue	Inventory	Update the device value.
5.41.	181	SetDeviceValues	Inventory	Update the device values.
5.42.	182	UpdateBrand	Inventory	Update a brand.
5.43.	184	UpdateGeoZone	Inventory	Update the geozone properties.
5.44.	186	UpdateProvider	Inventory	Update the given provider.
6.1.	188	CreateScheduledReportTemplateDefinition	Report	Create a new scheduled report definition.
6.2.	190	DeleteScheduledReportDefinition	Report	Delete the given scheduled report definition.

6.3.	191	GetAllScheduledReportTemplateDefinitions	Report	Return all the scheduled report template definitions.
6.4.	193	GetDevicesLastValues	Report	Return an array of last values for the given devices. The last values are read on the controller in realtime.
6.5.	195	GetDevicesLogValues	Report	Return an array of log values for the given devices between date range. The log values are the last values stored in the Streetlight.Vision database.
6.6.	199	GetGroupEnergy	Report	Return the energy consumption for the given geozone between date range.
6.7.	200	GetGroupEnergySaved	Report	Computes the energy consumed and the energy saved comparing to a theorical energy consumption for a given geozone between date range.
6.8.	200	GetRegisteredScheduledReportTemplateImplementations	Report	Return the scheduled reports templates.
6.9.	202	GetReport	Report	Return the report for the given geozone and report type.
6.10.	211	GetScheduledReportTemplateDefinition	Report	Return a scheduled report template definition.
6.11.	213	GetScheduledReportTemplateEditablePropertyDescriptors	Report	Return a scheduled report template editable property descriptors.
6.12.	222	GetVirtualScheduledReportTemplateEditablePropertyDescriptors	Report	Return a virtual scheduled report template editable property descriptors.
6.13.	234	UpdateScheduledReportTemplateDefinition	Report	Update a scheduled report definition.
7.1.	237	CreateProfil	User	Create a new profile.
7.2.	241	CreateUser	User	Create a new user.
7.3.	243	DeleteProfil	User	Delete a profile.
7.4.	244	DeleteUser	User	Delete an user.
7.5.	245	GetAllProfils	User	Return an array of all profiles.
7.6.	249	GetAllUsers	User	Return an array of all users.
7.7.	250	GetGeoZoneProfils	User	Return an array of profiles for the given geozone.
7.8.	255	GetGeoZoneUsers	User	Return an array of users for the given geozone.

7.9.	256	GetProfil	User	Return the profile for the given profile name.
7.10.	260	GetProfilProperties	User	Return an array of profile properties for the current logged user.
7.11.	264	GetUser	User	Return the user for the given user login.
7.12.	265	UpdateProfilProperties	User	Update the profile properties.
7.13.	266	UpdateUser	User	Update the user properties.

2. Alarm

2.1. AknowledgeAlarm

Acknowledge the given alarm.

2.1.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmAPI?methodName=acknowledgeAlarm

2.1.2. Supported Formats

XML, JSON

2.1.3. Supported request methods

GET

2.1.4. Parameters

alarmId	Identifier of alarm	Required
comment	Reason of acknowledgment	Optional

2.1.5. Example requests

This request returns a SLVAlarm object.

2.1.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVAlarm>
  <creationTime class="sql-timestamp">2010-08-19 12:10:10.0</creationTime>
  <generator>Failures Alarms Cagnes-1657</generator>
  <groupId>107</groupId>
  <id>3104</id>
  <lastStateChangeInfo>Comment</lastStateChangeInfo>
  <lastStateChangeSrc>admin</lastStateChangeSrc>
  <lastStateChangeTime class="sql-timestamp">2010-08-20 07:42:10.0</lastStateChangeTime>
  <name>Failures Alarms Cagnes</name>
  <priority>1</priority>
  <stateKey>ACKNOWLEGED</stateKey>
</com.dotv.streetlightserver.api.data.SLVAlarm>
```

Example with JSON format :

```
 {
   "creationTime" : "2010-08-19 12:10:10.0",
   "description" : null,
   "generator" : "Failures Alarms Cagnes-1657",
   "groupId" : 107,
   "id" : 3104,
   "lastStateChangeInfo" : "Comment",
   "lastStateChangeSrc" : "admin",
   "lastStateChangeTime" : "2010-08-20 07:42:10.0",
   "name" : "Failures Alarms Cagnes",
   "priority" : 0,
   "stateKey" : "ACKNOWLEGED"
}
```

2.2. AknowledgeAlarms

Acknowledge the given alarms.

2.2.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmAPI?methodName=acknowledgeAlarms

2.2.2. Supported Formats

XML, JSON

2.2.3. Supported request methods

GET

2.2.4. Parameters

alarmIds	Array of alarm identifiers	Required
comment	Reason of acknowledgment	Optional

2.2.5. Example requests

This request returns an array of SLVAlarm objects.

2.2.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <errorCode>0</errorCode>
  <value class="com.dotv.streetlightserver.api.data.SLVAlarm-array">
    <com.dotv.streetlightserver.api.data.SLVAlarm>
      <creationTime class="sql-timestamp">2010-12-21 07:35:51.0</creationTime>
      <generator>Alarme Luminaire - Peupliers-2655</generator>
      <groupId>193</groupId>
      <id>3750</id>
      <lastStateChangeInfo>comment</lastStateChangeInfo>
      <lastStateChangeSrc>admin</lastStateChangeSrc>
      <lastStateChangeTime>2012-01-24 13:47:06.283</lastStateChangeTime>
      <name>Alarme Luminaire - Peupliers</name>
      <priority>4</priority>
      <stateKey>ACKNOWLEGED</stateKey>
    </com.dotv.streetlightserver.api.data.SLVAlarm>
    <com.dotv.streetlightserver.api.data.SLVAlarm>
      <creationTime class="sql-timestamp">2010-12-27 08:55:51.0</creationTime>
      <generator>Alarme Luminaire - Peupliers-2659</generator>
      <groupId>193</groupId>
      <id>3766</id>
      <lastStateChangeInfo>comment</lastStateChangeInfo>
      <lastStateChangeSrc>admin</lastStateChangeSrc>
      <lastStateChangeTime>2012-01-24 13:47:06.298</lastStateChangeTime>
      <name>Alarme Luminaire - Peupliers</name>
      <priority>4</priority>
      <stateKey>ACKNOWLEGED</stateKey>
    </com.dotv.streetlightserver.api.data.SLVAlarm>
  </value>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : null,
   "value" :
      [
    {
      "creationTime" : "2011-07-06T17:06:45.000",
      "description" : null,
      "generator" : "failures alarms Rouen-1038",
      "groupId" : 80,
      "id" : 3927,
      "lastStateChangeInfo" : "comment",
      "lastStateChangeSrc" : "admin",
      "lastStateChangeTime" : "2012-01-24T13:52:10.405",
      "name" : "failures alarms Rouen",
      "priority" : 1,
      "stateKey" : "ACKNOWLEGED"
    },
    {
      "creationTime" : "2011-07-06T17:07:08.000",
      "description" : null,
      "generator" : "failures alarms Rouen-1041",
      "groupId" : 80,
      "id" : 3930,
      "lastStateChangeInfo" : "comment",
      "lastStateChangeSrc" : "admin",
      "lastStateChangeTime" : "2012-01-24T13:52:10.405",
      "name" : "failures alarms Rouen",
      "priority" : 1,
      "stateKey" : "ACKNOWLEGED"
    }
  ]
}
```

2.3. CreateSendEmailActionAlarmDefinition

Create a send email alarm definition.

2.3.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmManagementAPI?methodName=createSendEmailActionAlarmDefinition

2.3.2. Supported Formats

XML, JSON

2.3.3. Supported request methods

GET

2.3.4. Parameters

alarmDefinitionId	String identifier of the new alarm defintion	Required
triggerConditionImplClassName	Class name of the trigger condition	Required
geoZoneId	Identifier of the geozone	Required

2.3.5. Example requests

This request returns a SLVAlarmDefinition object.

2.3.5.1. Example with XML format :

```
 <SLVAlarmDefinition>
  <alarmStateChangeActionImplClassNames>
    <string>com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction</string>
  </alarmStateChangeActionImplClassNames>
  <id>DemoSendEmailAlarm</id>
  <geoZoneId>383</geoZoneId>
<triggerConditionImplClassName>com.dotv.streetlightserver.plugin.alarm.MonoAlarmMultiDeviceFailureCondition</triggerConditionImplClassName>
  <propertyValues>
    <entry>
      <string>action[0].message</string>
      <string>${TC}</string>
    </entry>
    <entry>
      <string>newAlarmWhenAcknowledged</string>
      <boolean>true</boolean>
    </entry>
    <entry>
      <string>triggerCondition.criticalCount</string>
      <int>1</int>
    </entry>
    <entry>
      <string>alarmPriority</string>
      <int>0</int>
    </entry>
  </propertyValues>
</SLVAlarmDefinition>
```

Example with JSON format :

```
 {
   "alarmStateChangeActionImplClassNames" :
      [
         "com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction"
      ],
   "geoZoneId" : 318,
   "id" : "DemoSendEmailAlarm",
   "propertyValues" :
      {
         "action[0].message" : "${TC}",
         "newAlarmWhenAcknowledged" : true,
         "alarmPriority" : 0,
         "triggerCondition.hoursDelay" : 36
      },
   "triggerConditionImplClassName"                                                            :
"com.dotv.streetlightserver.plugin.alarm.ControllerUpdateTimeAlarm"
}
```

2.4. CreateSingleActionAlarmDefinition

Create a single action alarm definition.

2.4.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmManagementAPI?methodName=createSingleActionAlarmDefinition

2.4.2. Supported Formats

XML, JSON

2.4.3. Supported request methods

GET

2.4.4. Parameters

alarmDefinitionId	String identifier of the new alarm defintion	Required
triggerConditionImplClassName	Class name of the trigger condition	Required
alarmStateChangeActionImplClassName	Class name of the state change action	Required
geoZoneId	Identifier of the geozone	Required

2.4.5. Example requests

This request returns a SLVAlarmDefinition object.

2.4.5.1. Example with XML format :

```
 <SLVAlarmDefinition>
  <alarmStateChangeActionImplClassNames>
    <string>com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction</string>
  </alarmStateChangeActionImplClassNames>
  <id>jmeterSingleActionAlarm</id>
  <geoZoneId>344</geoZoneId>

<triggerConditionImplClassName>com.dotv.streetlightserver.plugin.alarm.GroupFailureRatioCondition
</triggerConditionImplClassName>
  <propertyValues>
    <entry>
      <string>action[0].message</string>
      <string>${TC}</string>
    </entry>
    <entry>
      <string>triggerCondition.criticalFailureRatio</string>
      <float>0.0</float>
    </entry>
    <entry>
      <string>newAlarmWhenAcknowledged</string>
      <boolean>true</boolean>
    </entry>
    <entry>
      <string>alarmPriority</string>
      <int>0</int>
    </entry>
  </propertyValues>
</SLVAlarmDefinition>
```

Example with JSON format :

```
 {
   "alarmStateChangeActionImplClassNames" :
      [
         "com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction"
      ],
   "geoZoneId" : 269,
   "id" : "jmeterSingleActionAlarm",
   "propertyValues" :
      {
         "action[0].message" : "${TC}",
         "newAlarmWhenAcknowledged" : true,
         "triggerCondition.criticalCount" : 1,
         "alarmPriority" : 0
      },
   "triggerConditionImplClassName"                                        :
"com.dotv.streetlightserver.plugin.alarm.MonoAlarmMultiDeviceFailureCondition"
}
```

2.5. DeleteAlarmDefinition

Delete the given alarm definition.

2.5.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmManagementAPI?methodName=deleteAlarmDefinition

2.5.2. Supported Formats

XML, JSON

2.5.3. Supported request methods

GET

2.5.4. Parameters

alarmDefinitionId	String identifier of alarm defintion	Required

2.5.5. Example requests

This request returns a SLVResult object.

2.5.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <status>OK</status>
  <errorCode>0</errorCode>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : "OK",
   "value" : null
}
```

2.6. GetAlarmDefinition

Return an alarm definition.

2.6.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmManagementAPI?methodName=getAlarmDefinition

2.6.2. Supported Formats

XML, JSON

2.6.3. Supported request methods

GET

2.6.4. Parameters

alarmDefinitionId	String identifier of alarm definition	Required

2.6.5. Example requests

This request returns a SLVAlarmDefinition object.

2.6.5.1. Example with XML format :

```
 <SLVAlarmDefinition>
  <alarmStateChangeActionImplClassNames>
    <string>com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction</string>
  </alarmStateChangeActionImplClassNames>
  <id>Segment-Citylone_Defaut Lampe</id>
  <geoZoneId>191</geoZoneId>

<triggerConditionImplClassName>com.dotv.streetlightserver.plugin.alarm.MonoAlarmMultiDeviceFailureCondition</triggerConditionImplClassName>
  <propertyValues>
    <entry>
      <string>action[0].message</string>
      <string>${TC}</string>
    </entry>
    <entry>
      <string>action[0].to</string>
      <string-array>
        <string>support@streetlight-vision.com</string>
        <string>mailuser@streetlight-vision.com</string>
      </string-array>
    </entry>
    <entry>
      <string>triggerCondition.fullFilledMessageTemplate</string>
      <string>${FDC} points lumineux ont actuellement une panne de lampe</string>
    </entry>
    <entry>
  <string>newAlarmWhenAcknowledged</string>
  <boolean>true</boolean>
</entry>
<entry>
  <string>triggerCondition.deviceIds</string>
  <int-array>
    <int>2612</int>
    <int>2614</int>
    <int>2615</int>
    <int>2617</int>
    <int>2618</int>
    <int>2620</int>
    <int>2621</int>
    <int>2623</int>
    <int>2624</int>
    <int>2626</int>
    <int>2627</int>
    <int>2629</int>
    <int>2630</int>
    <int>2632</int>
    <int>2633</int>
    <int>2635</int>
    <int>2636</int>
    <int>2638</int>
    <int>2639</int>
    <int>2641</int>
    <int>2642</int>
    <int>2644</int>
    <int>2645</int>
    <int>2647</int>
    <int>2648</int>
    <int>2650</int>
    <int>2651</int>
    <int>2652</int>
    <int>2653</int>
    <int>2654</int>
    <int>2655</int>
    <int>2656</int>
    <int>2657</int>
    <int>2658</int>
```

```
        <int>2659</int>
        <int>2660</int>
        <int>2661</int>
        <int>2662</int>
        <int>2663</int>
        <int>2664</int>
      </int-array>
    </entry>
    <entry>
      <string>triggerCondition.criticalCount</string>
      <int>2</int>
    </entry>
    <entry>
      <string>alarmPriority</string>
      <int>2</int>
    </entry>
    <entry>
      <string>action[0].subject</string>
      <string>Alarme Sous Segment - Citylone : Defaut de lampe</string>
    </entry>
    <entry>
      <string>triggerCondition.triggeringMeaningsStrIds</string>
      <string-array>
        <string>Default8</string>
        <string>LampFailure</string>
      </string-array>
    </entry>
    <entry>
      <string>action[0].from</string>
      <string>Notification@streetlightmonitoring.com</string>
    </entry>
  </propertyValues>
</SLVAlarmDefinition>
```

Example with JSON format :

```
 {
   "alarmStateChangeActionImplClassNames" :
      [
         "com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction"
```

```
    ],
  "geoZoneId" : 290,
  "id" : "Alarme Retour Contacteur3 sur le Poste GRENIER",
  "propertyValues" :
    {
      "action[0].message" : "${TC}",
      "triggerCondition.inputName" : "InputModbus3",
      "action[0].to" :
        [
          "support@streetlight-vision.com",
          "mailuser@streetlight-vision.com"
        ],
      "triggerCondition.fullFilledMessageTemplate" : "Bonjour,\r\n\r\nUn défaut sur le contacteur N°3 a été enregistré il y a moins d'une heure.\r\n\r\nVérifiez qu'il n'y ait pas un dysfonctionnement du départ\r\n\r\nCordialement,\r\nVotre service de télégestion.",
      "newAlarmWhenAcknowledged" : true,
      "triggerCondition.hoursCheckInterval" : 1,
      "triggerCondition.inputValue" : true,
      "triggerCondition.controllerIds" :
        [
          "fr-boulogne-cimetiere-ilon5-MCCB"
        ],
      "alarmPriority" : 0,
      "action[0].subject" : "Alarme Retour Contacteur3 sur le Poste GRENIER",
      "action[0].from" : "Notification@streetlightmonitoring.com"
    },
  "triggerConditionImplClassName" : "com.dotv.streetlightserver.plugin.alarm.ControllerInputCondition"
}
```

2.7. GetAlarmDefinitionEditablePropertyDescriptors

Return the editable property descriptors for the given alarm definition.

2.7.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmManagementAPI?methodName=getAlarmDefinitionEditablePropertyDescriptors

2.7.2. Supported Formats

XML, JSON

2.7.3. Supported request methods

GET

2.7.4. Parameters

alarmDefinitionId	String identifier of alarm definition	Required

2.7.5. Example requests

This request returns an array SLVPropertyDescriptor objects.

2.7.5.1. Example with XML format :

```
 <SLVPropertyDescriptor-array>
  <SLVPropertyDescriptor>
    <name>alarmPriority</name>
    <labelKey></labelKey>
    <format>min=0&amp;max=9</format>
    <type>integer</type>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>newAlarmWhenAcknowledged</name>
    <labelKey></labelKey>
    <type>boolean</type>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>triggerCondition.fullFilledMessageTemplate</name>
    <label>Modèle Info</label>
    <labelKey>jsp.alarms.label.property.fullFilledMessageTemplate</labelKey>
    <type>string</type>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>triggerCondition.controllerIds</name>
    <label>Contrôleurs</label>
    <labelKey>jsp.alarms.label.property.controllerIds</labelKey>
    <type>select</type>
    <enumValues class="SLVLabelledValue-array">
    <SLVLabelledValue>
      <value class="string">Cimetiere_ILON1</value>
      <label>Cimetiere_ILON1</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="string">Cimetiere_ILON2</value>
      <label>Cimetiere_ILON2</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="string">Cimetiere_ILON3</value>
      <label>Cimetiere_ILON3</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="string">Cimetiere_ILON4</value>
      <label>Cimetiere_ILON4</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="string">fr-boulogne-cimetiere-ilon5-MCCB</value>
      <label>fr-boulogne-cimetiere-ilon5-MCCB</label>
    </SLVLabelledValue>
  </enumValues>
</SLVPropertyDescriptor>
<SLVPropertyDescriptor>
  <name>triggerCondition.inputName</name>
  <label>Entrée</label>
  <labelKey>jsp.alarms.label.property.inputName</labelKey>
  <type>select</type>
  <enumValues class="SLVLabelledValue-array">
    <SLVLabelledValue>
      <value class="string">DigitalInput1</value>
      <label>Input 1</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="string">DigitalInput2</value>
      <label>Input 2</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="string">InputModbus1</value>
      <label>1-Modbus input 1</label>
    </SLVLabelledValue>
```

```
        <SLVLabelledValue>
          <value class="string">InputModbus2</value>
          <label>2-Modbus input 2</label>
        </SLVLabelledValue>
        <SLVLabelledValue>
          <value class="string">InputModbus3</value>
          <label>3-Modbus input 3</label>
        </SLVLabelledValue>
        <SLVLabelledValue>
          <value class="string">InputModbus4</value>
          <label>4-Modbus input 4</label>
        </SLVLabelledValue>
      </enumValues>
    </SLVPropertyDescriptor>
    <SLVPropertyDescriptor>
      <name>triggerCondition.inputValue</name>
      <label>Valeur</label>
      <labelKey>jsp.alarms.label.property.inputValue</labelKey>
      <type>select</type>
      <enumValues class="SLVLabelledValue-array">
        <SLVLabelledValue>
          <value class="string">true</value>
          <label>ON</label>
        </SLVLabelledValue>
        <SLVLabelledValue>
          <value class="string">false</value>
          <label>OFF</label>
        </SLVLabelledValue>
      </enumValues>
    </SLVPropertyDescriptor>
    <SLVPropertyDescriptor>
      <name>triggerCondition.hoursCheckInterval</name>
      <label>Vérifier depuis</label>
      <labelKey>jsp.alarms.label.property.hoursCheckInterval</labelKey>
      <type>select</type>
      <enumValues class="SLVLabelledValue-array">
        <SLVLabelledValue>
          <value class="string">1</value>
          <label>1 hour</label>
        </SLVLabelledValue>
        <SLVLabelledValue>
          <value class="string">6</value>
          <label>6 hours</label>
        </SLVLabelledValue>
        <SLVLabelledValue>
          <value class="string">12</value>
          <label>12 hours</label>
        </SLVLabelledValue>
      </enumValues>
    </SLVPropertyDescriptor>
    <SLVPropertyDescriptor>
      <name>action[0].from</name>
      <label>De</label>
      <labelKey>jsp.alarms.label.property.from</labelKey>
      <format>^[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,4}$</format>
      <type>string</type>
    </SLVPropertyDescriptor>
    <SLVPropertyDescriptor>
      <name>action[0].to</name>
      <label>A</label>
      <labelKey>jsp.alarms.label.property.to</labelKey>
      <type>select</type>
      <enumValues class="SLVLabelledValue-array"/>
    </SLVPropertyDescriptor>
    <SLVPropertyDescriptor>
      <name>action[0].subject</name>
      <label>Sujet</label>
      <labelKey>jsp.alarms.label.property.subject</labelKey>
      <type>string</type>
    </SLVPropertyDescriptor>
    <SLVPropertyDescriptor>
      <name>action[0].message</name>
      <label>Message</label>
      <labelKey>jsp.alarms.label.property.message</labelKey>
      <type>string</type>
    </SLVPropertyDescriptor>
  </SLVPropertyDescriptor-array>
```

Example with JSON format :

```
[
  {
    "enumValues" : null,
    "format" : "min=0&max=9",
    "label" : null,
    "labelKey" : "",
    "name" : "alarmPriority",
    "type" : "integer"
  },
  {
    "enumValues" : null,
    "format" : null,
    "label" : null,
    "labelKey" : "",
    "name" : "newAlarmWhenAcknowledged",
    "type" : "boolean"
  },
  {
    "enumValues" : null,
    "format" : null,
    "label" : "Modèle Info",
    "labelKey" : "jsp.alarms.label.property.fullFilledMessageTemplate",
    "name" : "triggerCondition.fullFilledMessageTemplate",
    "type" : "string"
  },
  {
    "enumValues" :
      [
        {
          "label" : "Allumé en journée [litduringday]",
          "properties" : null,
          "value" : "litduringday"
        },
        {
          "label" : "Autre problème [otherfailure]",
          "properties" : null,
          "value" : "otherfailure"
        },
        {
          "label" : "Backup Allumage Local [BackupScheduler]",
        "properties" : null,
        "value" : "BackupScheduler"
      },
      {
        "label" : "Bouton de maintenance [DigiInA]",
        "properties" : null,
        "value" : "DigiInA"
      }
    ],
  "format" : null,
  "label" : "Défauts déclencheurs",
  "labelKey" : "jsp.alarms.label.property.triggeringMeaningsStrIds",
  "name" : "triggerCondition.triggeringMeaningsStrIds",
  "type" : "select"
},
{
  "enumValues" :
    [
      {
        "label" : "BO06G054a.OLC[0] [@fr-boulogne-postecim]",
        "properties" : null,
        "value" : "2917"
      },
      {
        "label" : "BO06G049a.OLC[0] [@fr-boulogne-postecim]",
        "properties" : null,
        "value" : "2918"
      },
      {
        "label" : "BO06G028a.OLC[0] [@fr-boulogne-postecim]",
        "properties" : null,
        "value" : "2919"
      },
      {
        "label" : "BO06G041a.OLC[0] [@fr-boulogne-postecim]",
        "properties" : null,
        "value" : "2920"
      }
    ],
  "format" : null,
  "label" : "Equipements",
  "labelKey" : "jsp.alarms.label.property.deviceIds",
  "name" : "triggerCondition.deviceIds",
  "type" : "select"
},
{
  "enumValues" : null,
  "format" : null,
  "label" : "Nombre critique d'équipement en défaut",
  "labelKey" : "jsp.alarms.label.property.criticalCount",
  "name" : "triggerCondition.criticalCount",
  "type" : "int"
},
{
  "enumValues" : null,
  "format" : "^[A-Z0-9._%+-]+@[A-Z0-9.-]+\\.[A-Z]{2,4}$",
  "label" : "De",
  "labelKey" : "jsp.alarms.label.property.from",
  "name" : "action[0].from",
  "type" : "string"
},
{
  "enumValues" :
    [
    ],
  "format" : null,
  "label" : "A",
  "labelKey" : "jsp.alarms.label.property.to",
  "name" : "action[0].to",
  "type" : "select"
},
{
  "enumValues" : null,
  "format" : null,
  "label" : "Sujet",
  "labelKey" : "jsp.alarms.label.property.subject",
  "name" : "action[0].subject",
  "type" : "string"
},
{
      "enumValues" : null,
      "format" : null,
      "label" : "Message",
      "labelKey" : "jsp.alarms.label.property.message",
      "name" : "action[0].message",
      "type" : "string"
    }
]
```

2.8. GetAllAlarmDefinitions

Return an array of alarm definitions.

2.8.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmManagementAPI?methodName=getAllAlarmDefinitions

2.8.2. Supported Formats

XML, JSON

2.8.3. Supported request methods

GET

2.8.4. Parameters

propertyDescriptors	True to retrieve every propertyDescriptor. False by default.	Optional

2.8.5. Example requests

This request returns an array of SLVAlarmDefinition objects.

2.8.5.1. Example with XML format :

```
 <SLVAlarmDefinition-array>
  <SLVAlarmDefinition>
    <alarmStateChangeActionImplClassNames>
      <string>com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction</string>
    </alarmStateChangeActionImplClassNames>
    <id>Alarme Luminaire - Dassault</id>
    <geoZoneId>192</geoZoneId>
<triggerConditionImplClassName>com.dotv.streetlightserver.plugin.alarm.MultiAlarmMultiDeviceFailureCondition</triggerConditionImplClassName>
    <propertyValues>
      <entry>
        <string>action[0].message</string>
        <string>${TC}</string>
      </entry>
      <entry>
        <string>action[0].to</string>
        <string-array>
          <string>support@streetlight-vision.com</string>
        </string-array>
      </entry>
      <entry>
        <string>triggerCondition.fullFilledMessageTemplate</string>
        <string>Alarme  : ${FMS}&#xd;
Sur ${FD}</string>
      </entry>
      <entry>
        <string>triggerCondition.deviceIds</string>
        <int-array>
          <int>2661</int>
          <int>2663</int>
        </int-array>
      </entry>
      <entry>
        <string>action[0].subject</string>
        <string>Alarme Luminaire</string>
      </entry>
      <entry>
        <string>triggerCondition.triggeringMeaningsStrIds</string>
        <string-array>
          <string>LowLampCurrent</string>
          <string>HighLampCurrent</string>
          <string>LowCurrent</string>
          <string>HighCurrent</string>
          <string>Default13</string>
          <string>CapacitorFailure</string>
          <string>Default8</string>
          <string>LampFailure</string>
          <string>DeviceFailure</string>
          <string>RelayFailure</string>
          <string>LowPowerFactor</string>
          <string>InternalComFailure</string>
          <string>BallastFailure</string>
          <string>ExternalComFailure</string>
          <string>Default0</string>
          <string>LowPower</string>
          <string>HighPower</string>
          <string>HighOLCTemperature</string>
          <string>HighBallastTemperature</string>
          <string>Default3</string>
          <string>Default4</string>
          <string>Default5</string>
          <string>Default7</string>
          <string>LowLampVoltage</string>
          <string>HighLampVoltage</string>
          <string>VoltageBelowSpecs</string>
          <string>LowVoltage</string>
          <string>HighVoltage</string>
        </string-array>
      </entry>
      <entry>
        <string>action[0].from</string>
        <string>Notification@streetlightmonitoring.com</string>
      </entry>
    </propertyValues>
  </SLVAlarmDefinition>
  ...
</SLVAlarmDefinition-array>
```

Example with JSON format :

```
 [
   null,
   null,
   null,
null,
null,
null,
{
  "alarmStateChangeActionImplClassNames" :
    [
      "com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction"
    ],
  "geoZoneId" : 192,
  "id" : "Alarme Luminaire - Dassault",
  "propertyValues" :
    {
      "action[0].message" : "${TC}",
      "action[0].to" :
        [
          "support@streetlight-vision.com"
        ],
      "triggerCondition.fullFilledMessageTemplate" : "Alarme  : ${FMS}\r\nSur ${FD}",
      "triggerCondition.deviceIds" :
        [
          2661,
          2663
        ],
      "action[0].subject" : "Alarme Luminaire",
      "triggerCondition.triggeringMeaningsStrIds" :
        [
          "LowLampCurrent",
          "HighLampCurrent",
          "LowCurrent",
          "HighCurrent",
          "Default13",
          "CapacitorFailure",
          "Default8",
          "LampFailure",
          "DeviceFailure",
          "RelayFailure",
          "LowPowerFactor",
          "InternalComFailure",
          "BallastFailure",
          "ExternalComFailure",
            "Default0",
            "LowPower",
            "HighPower",
            "HighOLCTemperature",
            "HighBallastTemperature",
            "Default3",
            "Default4",
            "Default5",
            "Default7",
            "LowLampVoltage",
            "HighLampVoltage",
            "VoltageBelowSpecs",
            "LowVoltage",
            "HighVoltage"
          ],
        "action[0].from" : "Notification@streetlightmonitoring.com"
      },
    "triggerConditionImplClassName"                                        :
"com.dotv.streetlightserver.plugin.alarm.MultiAlarmMultiDeviceFailureCondition"
  },
  ...
]
```

2.9. GetGeoZoneAlarms

Return an array of alarms for the given geozone.

2.9.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmAPI?methodName=getGeoZoneAlarms

2.9.2. Supported Formats

XML, JSON

2.9.3. Supported request methods

GET

2.9.4. Parameters

geoZoneId	Identifier of geozone	Required
minPriority	Minimal priority of alarms. 0 by default.	Optional
includeAcknowledgedAlarms	True to include acknowledged alarms. False by default.	Optional
recurseSubGeoZones	True to include alarms of sub geozones. False by default.	Optional

2.9.5. Example requests

This request returns an array of SLVAlarm objects.

2.9.5.1. Example with XML format :

```
<com.dotv.streetlightserver.api.data.SLVAlarm-array>
    <com.dotv.streetlightserver.api.data.SLVAlarm>
  <creationTime class="sql-timestamp">2010-08-19 09:10:09.0</creationTime>
  <generator>Failures Alarm Strasbourg-2231</generator>
  <groupId>215</groupId>
  <id>3074</id>
  <lastStateChangeInfo></lastStateChangeInfo>
  <lastStateChangeSrc>admin</lastStateChangeSrc>
  <lastStateChangeTime class="sql-timestamp">2010-08-19 11:59:16.0</lastStateChangeTime>
  <name>Failures Alarm Strasbourg</name>
  <priority>1</priority>
  <stateKey>ACKNOWLEGED</stateKey>
 </com.dotv.streetlightserver.api.data.SLVAlarm>
 <com.dotv.streetlightserver.api.data.SLVAlarm>
  <creationTime class="sql-timestamp">2010-08-19 09:10:10.0</creationTime>
  <generator>Lost node Strasbourg</generator>
  <groupId>215</groupId>
  <id>3075</id>
  <lastStateChangeInfo></lastStateChangeInfo>
  <lastStateChangeSrc>admin</lastStateChangeSrc>
  <lastStateChangeTime class="sql-timestamp">2010-08-19 11:59:16.0</lastStateChangeTime>
  <name>Lost node Strasbourg</name>
  <priority>1</priority>
  <stateKey>ACKNOWLEGED</stateKey>
 </com.dotv.streetlightserver.api.data.SLVAlarm>
 ...
```

```
</com.dotv.streetlightserver.api.data.SLVAlarm-array>
```

Example with JSON format :

```
 [
  {
    "creationTime" : "2010-08-20T19:10:10.000",
    "description" : null,
    "generator" : "Lamp Failure ETDE citylone-2624",
    "groupId" : 191,
    "id" : 3209,
    "lastStateChangeInfo" : "",
    "lastStateChangeSrc" : "admin",
    "lastStateChangeTime" : "2010-08-23T03:54:32.000",
    "name" : "Panne Lampes ETDE citylone",
    "priority" : 2,
    "stateKey" : "ACKNOWLEGED"
  },
  {
    "creationTime" : "2010-08-20T19:10:10.000",
    "description" : null,
    "generator" : "Lamp Failure ETDE citylone-2642",
    "groupId" : 191,
    "id" : 3210,
    "lastStateChangeInfo" : "",
    "lastStateChangeSrc" : "admin",
    "lastStateChangeTime" : "2010-08-23T03:54:32.000",
    "name" : "Panne Lampes ETDE citylone",
    "priority" : 2,
    "stateKey" : "ACKNOWLEGED"
  },
  ...
]
```

2.10. GetRegisteredAlarmStateChangeActionImplementations

Return an array of alarm state change actions.

2.10.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmManagementAPI?methodName=getRegisteredAlarmStateChangeActionImplementations

2.10.2. Supported Formats

XML, JSON

2.10.3. Supported request methods

GET

2.10.4. Example requests

This request returns an array of SLVLabelledValue objects.

2.10.4.1. Example with XML format :

```
 <list>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction</value>
    <label>Envoi d&apos;E-Mail</label>
  </SLVLabelledValue>
</list>
```

Example with JSON format :

```
 [
   {
      "label" : "Envoi d'E-Mail",
      "properties" : null,
      "value" : "com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction"
   }
]
```

2.11. GetRegisteredTriggerConditionImplementations

Return an array of trigger conditions.

2.11.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmManagementAPI?methodName=getRegisteredTriggerConditionImplementations

2.11.2. Supported Formats

XML, JSON

2.11.3. Supported request methods

GET

2.11.4. Example requests

This request returns an array of SLVLabelledValue objects.

2.11.4.1. Example with XML format :

```
 <list>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.alarm.GroupFailureRatioCondition</value>
    <label>Taux de défaut critique dans un Groupe</label>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.alarm.DeviceFailureCondition</value>
    <label>Défaut mono-équipement</label>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.alarm.MonoAlarmMultiDeviceFailureCondition</value>
    <label>Alarme unique sur plusieurs équipements</label>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.alarm.MultiAlarmMultiDeviceFailureCondition</value>
    <label>Alarmes multiples sur plusieurs équipements</label>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.alarm.ControllerInputCondition</value>
    <label>Alarme entrée du contrôleur</label>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.alarm.ControllerInputStateCondition</value>
    <label>Alarme état entrée du contrôleur</label>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.alarm.ControllerUpdateTimeAlarm</value>
    <label>Absence de donnée sur un contrôleur</label>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <value
class="string">com.dotv.streetlightserver.plugin.alarm.DevicesInAreaTriggersGenerator</value>
    <label>Trop de pannes dans une zone</label>
  </SLVLabelledValue>
</list>
```

Example with JSON format :

```
 [
   {
     "label" : "Taux de défaut critique dans un Groupe",
     "properties" : null,
     "value" : "com.dotv.streetlightserver.plugin.alarm.GroupFailureRatioCondition"
   },
   {
     "label" : "Défaut mono-équipement",
     "properties" : null,
     "value" : "com.dotv.streetlightserver.plugin.alarm.DeviceFailureCondition"
   },
   {
     "label" : "Alarme unique sur plusieurs équipements",
     "properties" : null,
     "value" : "com.dotv.streetlightserver.plugin.alarm.MonoAlarmMultiDeviceFailureCondition"
   },
   {
     "label" : "Alarmes multiples sur plusieurs équipements",
     "properties" : null,
     "value" : "com.dotv.streetlightserver.plugin.alarm.MultiAlarmMultiDeviceFailureCondition"
   },
   {
     "label" : "Alarme entrée du contrôleur",
     "properties" : null,
     "value" : "com.dotv.streetlightserver.plugin.alarm.ControllerInputCondition"
   },
   {
     "label" : "Alarme état entrée du contrôleur",
     "properties" : null,
     "value" : "com.dotv.streetlightserver.plugin.alarm.ControllerInputStateCondition"
   },
   {
     "label" : "Absence de donnée sur un contrôleur",
     "properties" : null,
     "value" : "com.dotv.streetlightserver.plugin.alarm.ControllerUpdateTimeAlarm"
   },
   {
     "label" : "Trop de pannes dans une zone",
     "properties" : null,
     "value" : "com.dotv.streetlightserver.plugin.alarm.DevicesInAreaTriggersGenerator"
   }
 ]
```

2.12. GetVirtualSingleActionAlarmDefinitionEditablePropertyDescritors

Return an array of editable property descriptors for a virtual single action alarm definition.

2.12.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmManagementAPI?methodName=GetVirtualSingleActionAlarmDefinitionEditablePropertyDescritors

2.12.2. Supported Formats

XML, JSON

2.12.3. Supported request methods

GET

2.12.4. Parameters

triggerConditionImplClassName	Class name of the trigger condition	Required
alarmStateChangeActionImplClassName	Class name of the state change action	Required
geoZoneId	Identifier of the geozone	Required

2.12.5. Example requests

This request returns an array of SLVPropertyDescriptor objects.

2.12.5.1. Example with XML format :

```
<SLVPropertyDescriptor-array>
  <SLVPropertyDescriptor>
    <name>alarmPriority</name>
    <labelKey></labelKey>
    <format>min=0&amp;max=9</format>
    <type>integer</type>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>newAlarmWhenAcknowledged</name>
    <labelKey></labelKey>
    <type>boolean</type>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>triggerCondition.fullFilledMessageTemplate</name>
    <label>Modèle Info</label>
    <labelKey>jsp.alarms.label.property.fullFilledMessageTemplate</labelKey>
    <type>string</type>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>triggerCondition.triggeringMeaningsStrIds</name>
    <label>Défauts déclencheurs</label>
    <labelKey>jsp.alarms.label.property.triggeringMeaningsStrIds</labelKey>
    <type>select</type>
    <enumValues class="SLVLabelledValue-array">
      <SLVLabelledValue>
        <value class="string">litduringday</value>
        <label>Allumé en journée [litduringday]</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="string">otherfailure</value>
        <label>Autre problème [otherfailure]</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="string">BackupScheduler</value>
        <label>Backup Allumage Local [BackupScheduler]</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="string">DigiInA</value>
        <label>Bouton de maintenance [DigiInA]</label>
      </SLVLabelledValue>
      ...
    </enumValues>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>triggerCondition.radius</name>
    <label>Rayon</label>
    <labelKey>jsp.alarms.label.property.radius</labelKey>
    <type>int</type>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>triggerCondition.threshold</name>
    <label>Seuil</label>
    <labelKey>jsp.alarms.label.property.threshold</labelKey>
    <type>int</type>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>action[0].from</name>
    <label>De</label>
    <labelKey>jsp.alarms.label.property.from</labelKey>
    <format>^[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,4}$</format>
    <type>string</type>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>action[0].to</name>
    <label>A</label>
    <labelKey>jsp.alarms.label.property.to</labelKey>
```

```
    <type>select</type>
    <enumValues class="SLVLabelledValue-array"/>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>action[0].subject</name>
    <label>Sujet</label>
    <labelKey>jsp.alarms.label.property.subject</labelKey>
    <type>string</type>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>action[0].message</name>
    <label>Message</label>
    <labelKey>jsp.alarms.label.property.message</labelKey>
    <type>string</type>
  </SLVPropertyDescriptor>
</SLVPropertyDescriptor-array>
```

Example with JSON format :

```
 [
  {
    "enumValues" : null,
    "format" : "min=0&max=9",
    "label" : null,
    "labelKey" : "",
    "name" : "alarmPriority",
    "type" : "integer"
  },
  {
    "enumValues" : null,
    "format" : null,
    "label" : null,
    "labelKey" : "",
    "name" : "newAlarmWhenAcknowledged",
    "type" : "boolean"
  },
  {
    "enumValues" : null,
    "format" : null,
    "label" : "Modèle Info",
    "labelKey" : "jsp.alarms.label.property.fullFilledMessageTemplate",
    "name" : "triggerCondition.fullFilledMessageTemplate",
    "type" : "string"
  },
  {
    "enumValues" :
      [
        {
          "label" : "Allumé en journée [litduringday]",
          "properties" : null,
          "value" : "litduringday"
        },
        {
          "label" : "Autre problème [otherfailure]",
          "properties" : null,
          "value" : "otherfailure"
        },
        {
          "label" : "Backup Allumage Local [BackupScheduler]",
          "properties" : null,
          "value" : "BackupScheduler"
        },
        {
          "label" : "Bouton de maintenance [DigiInA]",
          "properties" : null,
          "value" : "DigiInA"
        },
        ...
      ],
    "format" : null,
    "label" : "Défauts déclencheurs",
    "labelKey" : "jsp.alarms.label.property.triggeringMeaningsStrIds",
    "name" : "triggerCondition.triggeringMeaningsStrIds",
    "type" : "select"
  },
  {
    "enumValues" : null,
    "format" : null,
    "label" : "Rayon",
    "labelKey" : "jsp.alarms.label.property.radius",
    "name" : "triggerCondition.radius",
    "type" : "int"
  },
  {
    "enumValues" : null,
    "format" : null,
    "label" : "Seuil",
    "labelKey" : "jsp.alarms.label.property.threshold",
    "name" : "triggerCondition.threshold",
    "type" : "int"
  },
  {
    "enumValues" : null,
    "format" : "^[A-Z0-9._%+-]+@[A-Z0-9.-]+\\.[A-Z]{2,4}$",
    "label" : "De",
    "labelKey" : "jsp.alarms.label.property.from",
    "name" : "action[0].from",
    "type" : "string"
  },
  {
    "enumValues" :
      [
      ],
    "format" : null,
    "label" : "A",
    "labelKey" : "jsp.alarms.label.property.to",
    "name" : "action[0].to",
    "type" : "select"
  },
  {
    "enumValues" : null,
    "format" : null,
    "label" : "Sujet",
    "labelKey" : "jsp.alarms.label.property.subject",
    "name" : "action[0].subject",
    "type" : "string"
  },
  {
    "enumValues" : null,
    "format" : null,
    "label" : "Message",
    "labelKey" : "jsp.alarms.label.property.message",
    "name" : "action[0].message",
    "type" : "string"
  }
]
```

2.13. UpdateAlarmDefinition

Update the given alarm definition.

2.13.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAlarmManagementAPI?methodName=updateAlarmDefinition

2.13.2. Supported Formats

XML, JSON

2.13.3. Supported request methods

GET

2.13.4. Parameters

alarmDefinitionId	String identifier of alarm defintion	Required
propertyNames	Array of property names	Required
propertyValues	Array of property values	Required

2.13.5. Example requests

This request returns a SLVAlarmDefinition object.

2.13.5.1. Example with XML format :

```
<SLVAlarmDefinition>
 <alarmStateChangeActionImplClassNames>
   <string>com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction</string>
 </alarmStateChangeActionImplClassNames>
```

```
  <id>jmeterSingleActionAlarm</id>
  <geoZoneId>159</geoZoneId>
<triggerConditionImplClassName>com.dotv.streetlightserver.plugin.alarm.DevicesInAreaTriggersGenerator</triggerConditionImplClassName>
  <propertyValues>
    <entry>
      <string>action[0].message</string>
      <string>${TC}</string>
    </entry>
    <entry>
      <string>newAlarmWhenAcknowledged</string>
      <boolean>true</boolean>
    </entry>
    <entry>
      <string>triggerCondition.threshold</string>
      <int>3</int>
    </entry>
    <entry>
      <string>alarmPriority</string>
      <int>2</int>
    </entry>
    <entry>
      <string>triggerCondition.radius</string>
      <int>100</int>
    </entry>
  </propertyValues>
</SLVAlarmDefinition>
```

Example with JSON format :

```
{
  "alarmStateChangeActionImplClassNames" :
    [
      "com.dotv.streetlightserver.plugin.alarm.action.SendEMailAction"
    ],
  "geoZoneId" : 250,
  "id" : "jmeterSingleActionAlarm",
  "propertyValues" :
    {
      "action[0].message" : "${TC}",
      "newAlarmWhenAcknowledged" : true,
      "triggerCondition.threshold" : 3,
      "alarmPriority" : 2,
      "triggerCondition.radius" : 100
    },
  "triggerConditionImplClassName" : "com.dotv.streetlightserver.plugin.alarm.DevicesInAreaTriggersGenerator"
}
```

3. Commands

3.1. ExecuteCommand

Excute a command on the given device.

3.1.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=executeCommand

3.1.2. Supported Formats

XML, JSON

3.1.3. Supported request methods

GET

3.1.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required
commandName	Name of command to execute	Required

3.1.5. Example requests

This request returns a SLVResult object.

3.1.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <status>OK</status>
  <errorCode>0</errorCode>
  <value class="string">30.0 1</value>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : "OK",
   "value" : "30.0 1"
}
```

3.2. ExecuteCommands

Execute a command on the given devices.

3.2.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=executeCommands

3.2.2. Supported Formats

XML, JSON

3.2.3. Supported request methods

GET

3.2.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of device identifiers on controller	Required
commandName	Name of command to execute	Required

3.2.5. Example requests

This request returns an array of SLVResult objects.

3.2.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult-array>
  <com.dotv.streetlightserver.api.data.SLVResult>
    <status>OK</status>
    <errorCode>0</errorCode>
    <value class="string">0.0 0</value>
  </com.dotv.streetlightserver.api.data.SLVResult>
  <com.dotv.streetlightserver.api.data.SLVResult>
    <status>OK</status>
    <errorCode>0</errorCode>
    <value class="string">0.0 0</value>
  </com.dotv.streetlightserver.api.data.SLVResult>
  <com.dotv.streetlightserver.api.data.SLVResult>
    <status>OK</status>
    <errorCode>0</errorCode>
    <value class="string">100.0 1</value>
  </com.dotv.streetlightserver.api.data.SLVResult>
  ...
</com.dotv.streetlightserver.api.data.SLVResult-array>
```

Example with JSON format :

```
 [
  {
    "errorCode" : "0",
    "status" : "OK",
    "value" : "0.0 0"
  },
  {
    "errorCode" : "0",
    "status" : "OK",
    "value" : "0.0 0"
  },
  {
    "errorCode" : "0",
    "status" : "OK",
    "value" : "100.0 1"
  }
]
```

3.3. ExitManualModes

Exit manual mode for the given streetlights.

3.3.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVDimmingAPI?methodName=exitManualModes

3.3.2. Supported Formats

XML, JSON

3.3.3. Supported request methods

GET

3.3.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of streetlight identifiers on controller	Required

3.3.5. Example requests

This request returns a SLVResult object.

3.3.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <errorCode>0</errorCode>
  <value class="string-array">
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    ...
  </value>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : null,
   "value" :
    [
      "OK",
      "OK",
      "OK",
      ...
    ]
}
```

3.4. GetDeviceCommand

Return the given command for the given device.

3.4.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=getDeviceCommand

3.4.2. Supported Formats

XML, JSON

3.4.3. Supported request methods

GET

3.4.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required
commandName	Command name	Required

3.4.5. Example requests

This request returns a SLVCommand object.

3.4.5.1. Example with XML format :

```
<SLVCommand>
 <name>LampLevel.Dim50</name>
```

```
  <label>LampLevel.Dim50</label>
  <category>dimmingControl</category>
</SLVCommand>
```

Example with JSON format :

```
 {
   "category" : "dimmingControl",
   "label" : "LampLevel.Dim50",
   "name" : "LampLevel.Dim50"
}
```

3.5. GetDeviceCommands

Return an array of commands for the given device.

3.5.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=getDeviceCommands

3.5.2. Supported Formats

XML, JSON

3.5.3. Supported request methods

GET

3.5.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required

3.5.5. Example requests

This request returns an array of SLVCommand objects.

3.5.5.1. Example with XML format :

```
 <SLVCommand-array>
  <SLVCommand>
    <name>AutomaticLampLevel.Write</name>
    <label>AutomaticLampLevel.Write</label>
  </SLVCommand>
  <SLVCommand>
    <name>GroupControllerLampLevel.Write</name>
    <label>GroupControllerLampLevel.Write</label>
  </SLVCommand>
  <SLVCommand>
    <name>LampLevel.BackToAutoMode</name>
    <label>LampLevel.BackToAutoMode</label>
    <category>dimmingModeControl</category>
  </SLVCommand>
  ...
</SLVCommand-array>
```

Example with JSON format :

```
 [
   {
      "category" : null,
      "label" : "AutomaticLampLevel.Write",
      "name" : "AutomaticLampLevel.Write"
   },
   {
      "category" : null,
      "label" : "GroupControllerLampLevel.Write",
      "name" : "GroupControllerLampLevel.Write"
   },
   {
      "category" : dimmingModeControl,
      "label" : "LampLevel.BackToAutoMode",
      "name" : "LampLevel.BackToAutoMode"
   },
   ...
]
```

3.6. GetDeviceExecutableCommands

Return an array of executable commands for the given device.

3.6.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=getDeviceExecutableCommands

3.6.2. Supported Formats

XML, JSON

3.6.3. Supported request methods

GET

3.6.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required

3.6.5. Example requests

This request returns an array of SLVCommand objects.

3.6.5.1. Example with XML format :

```
 <SLVCommand-array>
  <SLVCommand>
    <name>LampLevel.BackToAutoMode</name>
    <label>LampLevel.BackToAutoMode</label>
    <category>dimmingModeControl</category>
  </SLVCommand>
  <SLVCommand>
    <name>LampLevel.Dim30</name>
    <label>LampLevel.Dim30</label>
    <category>dimmingControl</category>
  </SLVCommand>
  <SLVCommand>
    <name>LampLevel.Dim50</name>
    <label>LampLevel.Dim50</label>
    <category>dimmingControl</category>
  </SLVCommand>
  ...
</SLVCommand-array>
```

Example with JSON format :

```
 [
  {
    "category" : dimmingModeControl,
    "label" : "LampLevel.BackToAutoMode",
    "name" : "LampLevel.BackToAutoMode"
  },
  {
    "category" : dimmingControl,
    "label" : "LampLevel.Dim30",
    "name" : "LampLevel.Dim30"
  },
  {
    "category" : dimmingControl,
    "label" : "LampLevel.Dim50",
    "name" : "LampLevel.Dim50"
  },
  ...
]
```

3.7. GetDeviceFailureDescriptors

Return an array of failure descriptors for the given device.

3.7.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=getDeviceFailureDescriptors

3.7.2. Supported Formats

XML, JSON

3.7.3. Supported request methods

GET

3.7.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required

3.7.5. Example requests

This request returns an array of SLVValueDescriptor objects.

3.7.5.1. Example with XML format :

```
 <SLVValueDescriptor-array>
  <SLVValueDescriptor>
    <name>streetlight.failure.status.philipsLLC7030.mainCurrentHigh</name>
    <label>Courant élevé</label>
    <labelKey>streetlight.failure.status.philipsLLC7030.mainCurrentHigh</labelKey>
    <failure>false</failure>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  <SLVValueDescriptor>
    <name>streetlight.failure.status.philipsLLC7030.mainCurrentLow</name>
    <label>Courant faible</label>
    <labelKey>streetlight.failure.status.philipsLLC7030.mainCurrentLow</labelKey>
    <failure>false</failure>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  <SLVValueDescriptor>
    <name>streetlight.failure.status.philipsLLC7030.mainVoltageHigh</name>
    <label>Tension élevée</label>
    <labelKey>streetlight.failure.status.philipsLLC7030.mainVoltageHigh</labelKey>
    <failure>false</failure>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  ...
</SLVValueDescriptor-array>
```

Example with JSON format :

```
 [
  {
    "category" : null,
    "criticity" : 0,
    "dataFormat" : null,
    "failure" : false,
    "help" : null,
    "label" : "Courant élevé",
    "labelKey" : "streetlight.failure.status.philipsLLC7030.mainCurrentHigh",
    "name" : "streetlight.failure.status.philipsLLC7030.mainCurrentHigh",
    "type" : null,
    "unit" : null
  },
  {
    "category" : null,
    "criticity" : 0,
    "dataFormat" : null,
    "failure" : false,
    "help" : null,
    "label" : "Courant faible",
    "labelKey" : "streetlight.failure.status.philipsLLC7030.mainCurrentLow",
    "name" : "streetlight.failure.status.philipsLLC7030.mainCurrentLow",
    "type" : null,
    "unit" : null
  },
  {
    "category" : null,
    "criticity" : 0,
    "dataFormat" : null,
    "failure" : false,
    "help" : null,
    "label" : "Tension élevé",
    "labelKey" : "streetlight.failure.status.philipsLLC7030.mainVoltageHigh",
    "name" : "streetlight.failure.status.philipsLLC7030.mainVoltageHigh",
    "type" : null,
    "unit" : null
  },
 ...
```

]

3.8. GetDeviceFailureOnValues

Return the failure state of each device for the given failure names.

3.8.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=getDeviceFailureOnValues

3.8.2. Supported Formats

XML, JSON

3.8.3. Supported request methods

GET

3.8.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required
failureName	Array of failure names	Required
maxAge	Max age of iLON request	Required

3.8.5. Example requests

This request returns an array of Boolean objects.

3.9. GetDeviceFormattedMeteringValues

Return an array of metering values for the given device.

3.9.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=getDeviceFormattedMeteringValues

3.9.2. Supported Formats

XML, JSON

3.9.3. Supported request methods

GET

3.9.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required
meteringValueName	Array of metering value names	Required
maxAge	Max age of iLON request	Required

3.9.5. Example requests

This request returns an array of String objects.

3.9.5.1. Example with XML format :

```
 <string-array>
  <string>1631h</string>
  <string>0.0mA</string>
  ...
</string-array>
```

Example with JSON format :

```
 [
   "1631h",
   "0.0mA",
   ...
]
```

3.10. GetDeviceMeteringValueDescriptors

Return an array of metering value descriptors for the given device.

3.10.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=getDeviceMeteringValueDescriptors

3.10.2. Supported Formats

XML, JSON

3.10.3. Supported request methods

GET

3.10.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required

3.10.5. Example requests

This request returns an array of SLVValueDescriptor objects.

3.10.5.1. Example with XML format :

```
 <SLVValueDescriptor-array>
  <SLVValueDescriptor>
    <name>Temperature</name>
    <label>OLC Température</label>
    <labelKey>metering.Temperature.label</labelKey>
    <failure>false</failure>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  <SLVValueDescriptor>
    <name>SensorInput</name>
    <label>Entrée capteur</label>
    <labelKey>metering.SensorInput.label</labelKey>
    <failure>false</failure>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  ...
<SLVValueDescriptor-array>
```

Example with JSON format :

```
 [
  {
     "category" : null,
     "criticity" : 0,
     "dataFormat" : null,
     "failure" : false,
     "help" : null,
     "label" : "OLC Température",
     "labelKey" : "metering.Temperature.label",
     "name" : "Temperature",
     "type" : null,
     "unit" : null
  },
  {
     "category" : null,
     "criticity" : 0,
     "dataFormat" : null,
     "failure" : false,
     "help" : null,
     "label" : "Entrée capteur",
     "labelKey" : "metering.SensorInput.label",
     "name" : "SensorInput",
     "type" : null,
     "unit" : null
  },
  ...
]
```

3.11. GetDevicesFailureOnValues

Return the failure state of each device for the given failure names.

3.11.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=getDevicesFailureOnValues

3.11.2. Supported Formats

XML, JSON

3.11.3. Supported request methods

GET

3.11.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of device identifiers on controller	Required
failureName	Array of failure names	Required
maxAge	Max age of iLON request	Required

3.11.5. Example requests

This request returns an array of Boolean objects.

3.12. GetDevicesHasOneFailureOn

Return the failure state for each device.

3.12.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=getDevicesHasOneFailureOn

3.12.2. Supported Formats

XML, JSON

3.12.3. Supported request methods

GET

3.12.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of device identifiers on controller	Required
maxAge	Max age of iLON request	Required

3.12.5. Example requests

This request returns an array of Boolean objects.

3.13. GetDimmingLevel

Return the dimming level for the given streetlight.

3.13.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVDimmingAPI?methodName=getDimmingLevel

3.13.2. Supported Formats

XML, JSON

3.13.3. Supported request methods

GET

3.13.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of streetlight on controller	Required
maxAge	Max age of iLON request	Required

3.13.5. Example requests

This request returns a Float object.

3.13.5.1. Example with XML format :

```
<float>100.0</float>
```

Example with JSON format :

```
100.0
```

3.14. GetDimmingLevels

Return an array of dimming levels for the given streetlights.

3.14.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVDimmingAPI?methodName=getDimmingLevels

3.14.2. Supported Formats

XML, JSON

3.14.3. Supported request methods

GET

3.14.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of streetlight identifiers on controller	Required
maxAge	Max age of iLON request	Required

3.14.5. Example requests

This request returns an array of Float objects.

3.14.5.1. Example with XML format :

```
 <float-array>
  <float>0.0</float>
  <float>100.0</float>
  ...
</float-array>
```

Example with JSON format :

```
 [
   0.0,
   100.0,
   ...
]
```

3.15. GetDimmingMode

Return an array of dimming modes for the given streetlights.

3.15.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVDimmingAPI?methodName=getDimmingMode

3.15.2. Supported Formats

XML, JSON

3.15.3. Supported request methods

GET

3.15.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of streetlight identifiers on controller	Required
maxAge	Max age of iLON request	Required

3.15.5. Example requests

This request returns an array of String objects.

3.15.5.1. Example with XML format :

```
 <string-array>
  <string>AUTOMATIC</string>
  <string>MANUAL</string>
</string-array>
```

Example with JSON format :

```
 [
   "AUTOMATIC",
   "MANUAL"
]
```

3.16. GetGeoZoneFailures

Return an array of failures for the given geozone.

3.16.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVFailureAPI?methodName=getGeoZoneFailures

3.16.2. Supported Formats

XML, JSON

3.16.3. Supported request methods

GET

3.16.4. Parameters

geoZoneId	Identifier of geozone	Required

deviceCategoryStrId	Category of equipment (streetlight, controllerdevice, electricalCounter, cameraip, ...).	Optional
recurseGroupHierarchy	True to get failures recursively from sub geozones, otherwise false. Default value is True.	Optional
minCriticity	Minimum criticity of failures to return. Default value is 0.	Optional

3.16.5. Example requests

This request returns an array of SLVFailure objects.

3.16.5.1. Example with XML format :

```
 <SLVFailure-array>
  <SLVFailure>
    <deviceId>638</deviceId>
    <deviceName>device device[9] on SC_of_Dublin.2[1]</deviceName>
    <name>Lamp Fault</name>
    <strId>Default8</strId>
    <criticity>1</criticity>
    <timestamp class="sql-timestamp">2009-12-26 02:51:00.0</timestamp>
    <meaningLabel>Défaut de lampe</meaningLabel>
  </SLVFailure>
  <SLVFailure>
    <deviceId>641</deviceId>
    <deviceName>device device[12] on SC_of_Dublin.2[1]</deviceName>
    <name>Lamp Fault</name>
    <strId>Default8</strId>
    <criticity>1</criticity>
    <timestamp class="sql-timestamp">2009-12-26 04:34:00.0</timestamp>
    <meaningLabel>Défaut de lampe</meaningLabel>
  </SLVFailure>
  ...
<SLVFailure-array>
```

Example with JSON format :

```
 [
   {
    "criticity" : 1,
    "deviceId" : 638,
    "deviceName" : "device device[9] on SC_of_Dublin.2[1]",
    "meaningLabel" : "Défaut de lampe",
    "name" : "Lamp Fault",
    "strId" : "Default8",
    "timestamp" : "2009-12-26 02:51:00.0"
  },
  {
    "criticity" : 1,
    "deviceId" : 641,
    "deviceName" : "device device[12] on SC_of_Dublin.2[1]",
    "meaningLabel" : "Défaut de lampe",
    "name" : "Lamp Fault",
    "strId" : "Default8",
    "timestamp" : "2009-12-26 04:34:00.0"
  },
  ...
]
```

3.17. HasDeviceCommand

Check out that the given device has the given command.

3.17.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=hasDeviceCommand

3.17.2. Supported Formats

XML, JSON

3.17.3. Supported request methods

GET

3.17.4. Parameters

controllerStrId	Identifier of controller	Required

idOnController	Identifier of device on controller	Required
commandName	Command name	Required

3.17.5. Example requests

This request returns a Boolean object.

3.17.5.1. Example with XML format :

```
<boolean>true</boolean>
```

Example with JSON format :

```
true
```

3.18. ReplaceLamps

Replace the lamps for the given streetlights.

3.18.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringManagementAPI?methodName=replaceLamps

3.18.2. Supported Formats

XML, JSON

3.18.3. Supported request methods

GET

3.18.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of streetlight identifiers on controller	Required

3.18.5. Example requests

This request returns a SLVResult object.

3.19. ReplaceOLCs

Replace the given streetlights.

3.19.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringManagementAPI?methodName=replaceOLCs

3.19.2. Supported Formats

XML, JSON

3.19.3. Supported request methods

GET

3.19.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of streetlight identifiers on controller	Required
newNetworkIds	Array of new network identifiers	Required

3.19.5. Example requests

This request returns a SLVResult object.

3.20. SetDevicesMeteringValues

Update the metering values for the given devices.

3.20.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVMonitoringAPI?methodName=setDevicesMeteringValues

3.20.2. Supported Formats

XML, JSON

3.20.3. Supported request methods

GET

3.20.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of device identifiers on controller	Required
meteringValueName	Name of metering value	Required
meteringValue	Array of metering values	Required
priority	Array of priorities	Optional

3.20.5. Example requests

This request returns a SLVResult object.

3.21. SetDimmingLevel

Update the dimming level for the given streetlight.

3.21.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVDimmingAPI?methodName=setDimmingLevel

3.21.2. Supported Formats

XML, JSON

3.21.3. Supported request methods

GET

3.21.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of streetlight on controller	Required
dimmingLevel	Dimming level (float)	Required

3.21.5. Example requests

This request returns OK status.

3.21.5.1. Example with XML format :

```
<string>OK</string>
```

Example with JSON format :

```
"OK"
```

3.22. SetDimmingLevels

Update the dimming levels for the given streetlights.

3.22.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVDimmingAPI?methodName=setDimmingLevels

3.22.2. Supported Formats

XML, JSON

3.22.3. Supported request methods

GET

3.22.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of streetlight identifiers on controller	Required
dimmingLevel	Array of dimming levels (float)	Required

3.22.5. Example requests

This request returns a SLVResult object.

3.22.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <status>OK</status>
  <errorCode>0</errorCode>
  <value class="object-array">
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    ...
  </value>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : null,
   "value" :
      [
         "OK",
         "OK",
         "OK",
        "OK",
        "OK",
        "OK",
        "OK",
        "OK",
        ...
    ]
}
```

3.23. SwitchOff

Switch off the given streetlight.

3.23.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVDimmingAPI?methodName=switchOff

3.23.2. Supported Formats

XML, JSON

3.23.3. Supported request methods

GET

3.23.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of streetlight on controller	Required

3.23.5. Example requests

This request returns OK status.

3.23.5.1. Example with XML format :

```
<string>OK</string>
```

Example with JSON format :

```
"OK"
```

3.24. SwitchOn

Switch on the given streetlight.

3.24.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVDimmingAPI?methodName=switchOn

3.24.2. Supported Formats

XML, JSON

3.24.3. Supported request methods

GET

3.24.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of streetlight on controller	Required

3.24.5. Example requests

This request returns OK status.

3.24.5.1. Example with XML format :

```
<string>OK</string>
```

Example with JSON format :

```
"OK"
```

4. Controller

4.1. CommissionControllerAsync

Commission a controller asynchronously (only compatible with starsense RF).

4.1.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerManagementAPI?methodName=commissionControllerAsync

4.1.2. Supported Formats

XML, JSON

4.1.3. Supported request methods

GET

4.1.4. Parameters

controllerStrId	Identifier of controller	Required
flags	Integer indicating what will be commissioned (Bits : 4 = OLCs, 2 = Schedulers, 1 = Controller config).	Required

4.1.5. Example requests

This request returns a SLVBatchResult object.

4.1.5.1. Example with XML format :

```
 <SLVBatchResult>
  <status>OK</status>
  <errorCode>0</errorCode>
  <value class="com.dotv.streetlightserver.api.data.SLVBatchStepsResult">
    <stepResults/>
  </value>
  <batchRunning>true</batchRunning>
  <batchId>1329840341444</batchId>
  <batchProgressValue>-1</batchProgressValue>
```

```
</SLVBatchResult>
```

Example with JSON format :

```
 {
   "batchId" : "1329840375468",
   "batchProgressMessage" : null,
   "batchProgressValue" : -1,
   "batchRunning" : true,
   "errorCode" : "0",
   "status" : "OK",
   "value" :
      {
         "login" : "admin"
      }
}
```

4.2. CreateController

Create a new controller.

4.2.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=createController

4.2.2. Supported Formats

XML, JSON

4.2.3. Supported request methods

GET

4.2.4. Parameters

controllerStrId	Identifier of controller	Required
geoZoneId	Identifier of geozone	Required
lat	Latitude of controller	Required

lng	Longitude of controller	Required

4.2.5. Example requests

This request returns a SLVController object.

4.2.5.1. Example with XML format :

```
 <SLVController>
  <type>controller</type>
  <name>newController</name>
  <strId>newController</strId>
  <installDate>2011-06-27 15:55:33.331</installDate>
  <controllerDevice>
    <id>28</id>
    <type>device</type>
    <name>newController</name>
    <controllerStrId>newController</controllerStrId>
    <idOnController>controllerdevice</idOnController>
    <categoryStrId>controllerdevice</categoryStrId>
    <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
    <lat>1.0</lat>
    <lng>2.0</lng>
    <properties>
      <SLVKeyValuePair>
        <key>power</key>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>powerCorrection</key>
      </SLVKeyValuePair>
    </properties>
  </controllerDevice>
  <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
</SLVController>
```

Example with JSON format :

```
 {
   "controllerDevice" :
    {
      "address" : null,
      "categoryStrId" : "controllerdevice",
      "controllerStrId" : "newController",
      "functionId" : null,
      "geoZoneNamesPath" : "GeoZones/Streetlight.Vision Lab",
      "id" : 60,
      "idOnController" : "controllerdevice",
      "lat" : 1.0,
      "lng" : 2.0,
      "modelName" : null,
      "name" : "newController",
      "nodeTypeStrId" : null,
      "properties" :
        [
          {
            "key" : "power",
            "value" : null
          },
          {
            "key" : "powerCorrection",
            "value" : null
          }
        ],
      "technologyStrId" : null,
      "type" : "device"
    },
  "geoZoneNamesPath" : "GeoZones/Streetlight.Vision Lab",
  "host" : null,
  "id" : null,
  "installDate" : "2011-07-05T15:55:33.867",
  "model" : null,
  "name" : "newController",
  "strId" : "newController",
  "type" : "controller",
  "updateDate" : null
}
```

4.3. ExitOuputManualMode

Exit manual mode for an output of the given controller.

4.3.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=exitOuputManualMode

4.3.2. Supported Formats

XML, JSON

4.3.3. Supported request methods

GET

4.3.4. Parameters

controllerStrId	Identifier of controller	Required
ouputIndex	Index of output	Required

4.3.5. Example requests

This request returns OK status.

4.3.5.1. Example with XML format :

```
<string>OK</string>
```

Example with JSON format :

```
"OK"
```

4.4. GetAllControllers

Return an array of all controllers.

4.4.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getAllControllers

4.4.2. Supported Formats

XML, JSON

4.4.3. Supported request methods

GET

4.4.4. Example requests

This request returns an array of SLVController objects.

4.4.4.1. Example with XML format :

```
 <SLVController-array>
  <SLVController>
    <type>controller</type>
    <name>SLV_Demo</name>
    <strId>SLV_Demo</strId>
    <host>streetlight.vision.com:8080</host>
    <model>ILON - adsl</model>
    <installDate class="sql-timestamp">2011-06-24 00:00:00.0</installDate>
    <controllerDevice>
      <id>1</id>
      <type>device</type>
      <name>SLV_Demo</name>
      <controllerStrId>SLV_Demo</controllerStrId>
      <idOnController>controllerdevice</idOnController>
      <categoryStrId>controllerdevice</categoryStrId>
      <address></address>
      <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
      <lat>48.872814</lat>
      <lng>2.321686</lng>
      <properties>
        <SLVKeyValuePair>
          <key>power</key>
        </SLVKeyValuePair>
        <SLVKeyValuePair>
          <key>powerCorrection</key>
        </SLVKeyValuePair>
        <SLVKeyValuePair>
          <key>userproperty.DigitalInput1Info</key>
          <value class="string"></value>
        </SLVKeyValuePair>
        <SLVKeyValuePair>
          <key>userproperty.DigitalInput2Info</key>
          <value class="string"></value>
        </SLVKeyValuePair>
        <SLVKeyValuePair>
          <key>userproperty.DigitalOutput1Info</key>
          <value class="string">Output 1</value>
        </SLVKeyValuePair>
        <SLVKeyValuePair>
          <key>userproperty.DigitalOutput2Info</key>
          <value class="string">Output 2</value>
        </SLVKeyValuePair>
      </properties>
    </controllerDevice>
    <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
  </SLVController>
  ...
</SLVController-array>
```

Example with JSON format :

```
 [
   {
      "controllerDevice" :
         {
            "address" : "",
            "categoryStrId" : "controllerdevice",
            "controllerStrId" : "SLV_Demo",
            "functionId" : null,
            "geoZoneNamesPath" : "GeoZones/Streetlight.Vision Lab",
            "id" : 1,
            "idOnController" : "controllerdevice",
            "lat" : 48.872814,
            "lng" : 2.321686,
            "modelName" : null,
    "name" : "SLV_Demo",
    "nodeTypeStrId" : null,
    "properties" :
      [
        {
          "key" : "power",
          "value" : null
        },
        {
          "key" : "powerCorrection",
          "value" : null
        },
        {
          "key" : "userproperty.DigitalInput1Info",
          "value" : ""
        },
        {
          "key" : "userproperty.DigitalInput2Info",
          "value" : ""
        },
        {
          "key" : "userproperty.DigitalOutput1Info",
          "value" : "Output 1"
        },
        {
          "key" : "userproperty.DigitalOutput2Info",
          "value" : "Output 2"
        }
      ],
    "technologyStrId" : null,
    "type" : "device"
  },
"geoZoneNamesPath" : "GeoZones/Streetlight.Vision Lab",
"host" : "streetlight.vision.com:8080",
"id" : null,
"installDate" : "2011-06-24 00:00:00.0",
"model" : "ILON - adsl",
"name" : "SLV_Demo",
"strId" : "SLV_Demo",
"type" : "controller",
        "updateDate" : null
    },
    ...
]
```

4.5. GetControllerDeviceDevice

Return the given controller as device.

4.5.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=getControllerDeviceDevice

4.5.2. Supported Formats

XML, JSON

4.5.3. Supported request methods

GET

4.5.4. Parameters

controllerStrId	Identifier of controller	Required

4.5.5. Example requests

This request returns a SLVDevice object.

4.6. GetCumulatedDatalogSize

Return the datalog size for the given controller between date range.

4.6.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=getCumulatedDatalogSize

4.6.2. Supported Formats

XML, JSON

4.6.3. Supported request methods

GET

4.6.4. Parameters

controllerStrId	Identifier of controller	Required
from	Date of the oldest datalog (dd/MM/yyyy HH:mm:ss)	Required
to	Date of the newest datalog (dd/MM/yyyy HH:mm:ss)	Required

4.6.5. Example requests

This request returns a SLVResult object.

4.6.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <errorCode>0</errorCode>
  <value class="long">113</value>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : null,
   "value" : 113
}
```

4.7. GetOutputIndicesControllingMain

Return an array of output indices controlling main.

4.7.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=getOutputIndicesControllingMain

4.7.2. Supported Formats

XML, JSON

4.7.3. Supported request methods

GET

4.7.4. Parameters

controllerStrId	Identifier of controller	Required

4.7.5. Example requests

This request returns an Integer array.

4.7.5.1. Example with XML format :

```
 <java.lang.Integer-array>
  <int>1</int>
</java.lang.Integer-array>
```

Example with JSON format :

```
 [
   1
]
```

4.8. GetOutputIndicesControllingMains

Return arrays of output indices controlling mains.

4.8.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=getOutputIndicesControllingMains

4.8.2. Supported Formats

XML, JSON

4.8.3. Supported request methods

GET

4.8.4. Parameters

controllerStrIds	Array of controller string identifiers	Required

4.8.5. Example requests

This request returns an SLVResult object.

4.8.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <errorCode>0</errorCode>
  <value class="java.lang.Integer-array-array">
    <java.lang.Integer-array/>
    <java.lang.Integer-array/>
    <java.lang.Integer-array/>
    <java.lang.Integer-array/>
    <java.lang.Integer-array/>
    <java.lang.Integer-array>
      <int>1</int>
    </java.lang.Integer-array>
  </value>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
  "errorCode" : "0",
  "status" : null,
  "value" :
    [
      [
      ],
      [
        1
      ],
      [
        1
      ],
      [
      ],
      [
        1
      ],
      [
      ]
    ]
}
```

4.9. GetSystemTimeStringInLocalTime

Return the system time as local time for the given controller.

4.9.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=getSystemTimeStringInLocalTime

4.9.2. Supported Formats

XML, JSON

4.9.3. Supported request methods

GET

4.9.4. Parameters

controllerStrId	Identifier of controller	Required

4.9.5. Example requests

This request returns a String object.

4.9.5.1. Example with XML format :

```
<string>2011-06-27 16:55:31</string>
```

Example with JSON format :

```
"2011-06-27 16:55:31"
```

4.10. GetTimeZone

Return the timezone for the given controller.

4.10.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=getTimeZone

4.10.2. Supported Formats

XML, JSON

4.10.3. Supported request methods

GET

4.10.4. Parameters

controllerStrId	Identifier of controller	Required

4.10.5. Example requests

This request returns a SLVTimeZone object.

4.10.5.1. Example with XML format :

```
 <SLVTimeZone>
  <id>(GMT+0100) Brussels</id>
  <displayName>(GMT+01:00) Brussels, Copenhagen, Madrid, Paris</displayName>
  <dstSaving>0</dstSaving>
  <rawOffset>3600000</rawOffset>
</SLVTimeZone>
```

Example with JSON format :

```
 {
   "displayName" : "(GMT+01:00) Brussels, Copenhagen, Madrid, Paris",
   "dstSaving" : 0,
   "id" : "(GMT+0100) Brussels",
   "rawOffset" : 3600000
}
```

4.11. GetTimeZoneId

Return the timezone identifier for the given controller.

4.11.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=getTimeZoneId

4.11.2. Supported Formats

XML, JSON

4.11.3. Supported request methods

GET

4.11.4. Parameters

controllerStrId	Identifier of controller	Required

4.11.5. Example requests

This request returns a String object.

4.11.5.1. Example with XML format :

```
<string>(GMT+0100) Brussels</string>
```

Example with JSON format :

```
"(GMT+0100) Brussels"
```

4.12. ReadControllerInputState

Return the state of an input on the given controller.

4.12.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=readControllerInputState

4.12.2. Supported Formats

XML, JSON

4.12.3. Supported request methods

GET

4.12.4. Parameters

controllerStrId	Identifier of controller	Required
inputIndex	Index of input	Required

4.12.5. Example requests

This request returns a Boolean object.

4.12.5.1. Example with XML format :

<boolean>false</boolean>

Example with JSON format :

false

4.13. ReadControllerOutputState

Return the state of an output on the given controller.

4.13.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=readControllerOutputState

4.13.2. Supported Formats

XML, JSON

4.13.3. Supported request methods

GET

4.13.4. Parameters

controllerStrId	Identifier of controller	Required
outputIndex	Index of output	Required

4.13.5. Example requests

This request returns a Boolean object.

4.13.5.1. Example with XML format :

<boolean>true</boolean>

Example with JSON format :

true

4.14. SendControllerDataNow

Tell to the given controller to send its datalogs.

4.14.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=sendControllerDataNow

4.14.2. Supported Formats

XML, JSON

4.14.3. Supported request methods

GET

4.14.4. Parameters

controllerStrId	Identifier of controller	Required

4.14.5. Example requests

This request returns OK status.

4.14.5.1. Example with XML format :

```
<string>OK</string>
```

Example with JSON format :

```
"OK"
```

4.15. SetSystemTime

Update the system time for the given controller.

4.15.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=setSystemTime

4.15.2. Supported Formats

XML, JSON

4.15.3. Supported request methods

GET

4.15.4. Parameters

controllerStrId	Identifier of controller	Required
time	New system time	Required

4.15.5. Example requests

This request returns OK status.

4.15.5.1. Example with XML format :

```
<string>OK</string>
```

Example with JSON format :

```
"OK"
```

4.16. SetSystemTimeStringInLocalTime

Update the system time for the given controller.

4.16.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=setSystemTimeStringInLocalTime

4.16.2. Supported Formats

XML, JSON

4.16.3. Supported request methods

GET

4.16.4. Parameters

controllerStrId	Identifier of controller	Required
systemTimeString	Time string	Required

4.16.5. Example requests

This request returns OK status.

4.17. SetTimeZoneId

Update the timezone for the given controller.

4.17.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=setTimeZoneId

4.17.2. Supported Formats

XML, JSON

4.17.3. Supported request methods

GET

4.17.4. Parameters

controllerStrId	Identifier of controller	Required
timeZoneId	New timezone identifier	Required

4.17.5. Example requests

This request returns OK status.

4.17.5.1. Example with XML format :

```
 <string>OK</string>
```

Example with JSON format :

```
 "OK"
```

4.18. SwitchControllerOutput

Switch an output of the given controller.

4.18.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVControllerAPI?methodName=switchControllerOutput

4.18.2. Supported Formats

XML, JSON

4.18.3. Supported request methods

GET

4.18.4. Parameters

controllerStrId	Identifier of controller	Required
ouputIndex	Index of output	Required
switchState	True to switch on, False to switch off.	Required

4.18.5. Example requests

This request returns OK status.

4.18.5.1. Example with XML format :

```
<string>OK</string>
```

Example with JSON format :

```
"OK"
```

5. Inventory

5.1. CreateBrand

Create a new brand.

5.1.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=createBrand

5.1.2. Supported Formats

XML, JSON

5.1.3. Supported request methods

GET

5.1.4. Parameters

name	Name of the new brand	Required
description	Description of the new brand	Optional
lifeTime	Brand lifetime	Optional

5.1.5. Example requests

This request returns the SLVLabelledValue object of new brand.

5.1.5.1. Example with XML format :

```
 <SLVLabelledValue>
  <label>newBrand</label>
  <value class="int">106</value>
  <properties>
    <entry>
      <string>description</string>
      <string>newBrand</string>
    </entry>
    <entry>
      <string>name</string>
      <string>newBrand</string>
    </entry>
    <entry>
      <string>lifeTime</string>
      <string>null</string>
    </entry>
  </properties>
</SLVLabelledValue>
```

Example with JSON format :

```
 {
   "label" : "newBrand",
   "properties" :
      {
         "description" : "newBrand",
         "name" : "newBrand",
         "lifeTime" : "null"
      },
   "value" : 106
}
```

5.2. CreateCategoryDevice

Create a new device.

5.2.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=createCategoryDevice

5.2.2. Supported Formats

XML, JSON

5.2.3. Supported request methods

GET

5.2.4. Parameters

categoryStrId	Identifier of category	Required
controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required
userName	Name of user	Required
geoZoneId	Identifier of geozone	Required
lat	Latitude of device	Required
lng	Longitude of device	Required

5.2.5. Example requests

This request returns the SLVDevice object.

5.2.5.1. Example with XML format :

```
 <SLVDevice>
  <id>22</id>
  <type>device</type>
  <name>newDevice</name>
  <controllerStrId>SLV_Demo</controllerStrId>
  <idOnController>newDevice</idOnController>
  <categoryStrId>undefined</categoryStrId>
  <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
  <lat>1.0</lat>
  <lng>2.0</lng>
  <properties>
    <SLVKeyValuePair>
      <key>power</key>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>powerCorrection</key>
    </SLVKeyValuePair>
  </properties>
</SLVDevice>
```

Example with JSON format :

```
 {
   "address" : null,
   "categoryStrId" : "undefined",
   "controllerStrId" : "EN01",
   "functionId" : null,
   "geoZoneNamesPath" : "GeoZones/Streetlight.Vision Lab",
   "id" : 22,
   "idOnController" : "newDevice",
   "lat" : 1.0,
   "lng" : 2.0,
   "modelName" : null,
   "name" : "newDevice",
   "nodeTypeStrId" : null,
   "properties" :
     [
       {
         "key" : "power",
         "value" : null
       },
       {
         "key" : "powerCorrection",
         "value" : null
       }
     ],
   "technologyStrId" : null,
   "type" : "device"
 }
```

5.3. CreateGeoZone

Create a new geozone.

5.3.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=createGeoZone

5.3.2. Supported Formats

XML, JSON

5.3.3. Supported request methods

GET

5.3.4. Parameters

name	Name of geozone	Required
parentId	Identifier of parent geozone	Required
latMin	Minimum latitude of geozone	Required
latMax	Maximum latitude of geozone	Required
lngMin	Minimum longitude of geozone	Required
lngMax	Maximum longitude of geozone	Required

5.3.5. Example requests

This request returns a SLVGeoZone object.

5.3.5.1. Example with XML format :

```
 <SLVGeoZone>
    <id>101</id>
    <type>geozone</type>
    <name>MyGeoZone</name>
    <namesPath>MyParentGeoZone/MyGeoZone</namesPath>
    <idsPath>100/101</idsPath>
    <childrenCount>0</childrenCount>
    <devicesCount>0</devicesCount>
    <latMax>45.799207</latMax>
    <latMin>45.708545</latMin>
    <lngMax>4.914481</lngMax>
    <lngMin>4.775693</lngMin>
</SLVGeoZone>
```

Example with JSON format :

```
 {
   "childrenCount" : 0,
   "devicesCount" : 0,
   "id" : 101,
   "idsPath" : "100/101",
   "latMax" : 45.799207,
   "latMin" : 45.708545,
   "lngMax" : 4.914481,
   "lngMin" : 4.775693,
   "name" : "MyGeoZone",
   "namesPath" : "MyParentGeoZone/MyGeoZone",
   "type" : "geozone"
}
```

5.4. CreateProvider

Create a new provider.

5.4.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=createProvider

5.4.2. Supported Formats

XML, JSON

5.4.3. Supported request methods

GET

5.4.4. Parameters

name	Name of the provider	Required
parentId	Identifier of the parent provider	Optional
pollutionRate	Pollution rate	Optional

5.4.5. Example requests

This request returns the SLVProvider object of new provider.

5.4.5.1. Example with XML format :

```
 <SLVProvider>
  <id>20</id>
  <type>provider</type>
  <name>newProvider</name>
  <pollution>0.11</pollution>
</SLVProvider>
```

Example with JSON format :

```
 {
   "id" : 20,
   "name" : "newProvider",
   "pollution" : 0.11,
   "type" : "provider"
}
```

5.5. DeleteBrand

Delete a brand.

5.5.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=deleteBrand

5.5.2. Supported Formats

XML, JSON

5.5.3. Supported request methods

GET

5.5.4. Parameters

id	Id of the brand	Required

5.5.5. Example requests

This request returns a SLVResult object.

5.5.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <status>OK</status>
  <errorCode>0</errorCode>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : "OK"
}
```

5.6. DeleteDevices

Delete devices.

5.6.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=deleteDevices

5.6.2. Supported Formats

XML, JSON

5.6.3. Supported request methods

GET

5.6.4. Parameters

controllerStrId	Array of identifiers of controller	Required
idOnController	Array of identifiers of device on controller	Required

5.6.5. Example requests

This request returns a SLVResult object.

5.6.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <status>OK</status>
  <errorCode>0</errorCode>
  <value class="object-array">
    <string>OK</string>
    <string>OK</string>
    ...
  </value>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : "OK",
   "value" :
      [
         "OK",
         "OK",
         ...
      ]
}
```

5.7. DeleteGeoZone

Delete a geozone.

5.7.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=deleteGeoZone

5.7.2. Supported Formats

XML, JSON

5.7.3. Supported request methods

GET

5.7.4. Parameters

geoZoneId	Identifier of geozone	Required
pullUpContent	True to pull up the content of geozone (sub geozones and devices) into parent geozone, otherwise the content is deleted. Default value is True.	Optional

5.7.5. Example requests

This request returns OK status.

5.7.5.1. Example with XML format :

```
<string>OK</string>
```

Example with JSON format :

```
"OK"
```

5.8. DeleteProvider

Delete a provider.

5.8.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=deleteProvider

5.8.2. Supported Formats

XML, JSON

5.8.3. Supported request methods

GET

5.8.4. Parameters

id	Id of the provider	Required

5.8.5. Example requests

This request returns a SLVResult object.

5.8.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <status>OK</status>
  <errorCode>0</errorCode>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : "OK"
}
```

5.9. GetAllBrands

Return an array of all brands.

5.9.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getAllBrands

5.9.2. Supported Formats

XML, JSON

5.9.3. Supported request methods

GET

5.9.4. Example requests

This request returns an array of SLVLabelledValue objects.

5.9.4.1. Example with XML format :

```
 <SLVLabelledValue-array>
  <SLVLabelledValue>
    <label>Brand 1</label>
    <value class="int">1</value>
    <properties>
      <entry>
        <string>description</string>
        <string>Brand 1</string>
      </entry>
      <entry>
        <string>lifeTime</string>
        <string>null</string>
      </entry>
    </properties>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <label>HPS 70W Ferro</label>
    <value class="int">2</value>
    <properties>
      <entry>
        <string>description</string>
        <string>HPS 70W Ferro</string>
      </entry>
      <entry>
        <string>lifeTime</string>
        <string>20000</string>
      </entry>
    </properties>
  </SLVLabelledValue>
  ...
</SLVLabelledValue-array>
```

Example with JSON format :

```
[
  {
    "label" : "Brand 1",
    "properties" :
      {
        "description" : "Brand 1",
        "lifeTime" : "null"
      },
    "value" : 1
  },
  {
    "label" : "HPS 70W Ferro",
    "properties" :
      {
        "description" : "HPS 70W Ferro",
        "lifeTime" : "20000"
      },
    "value" : 2
  },
  ...
]
```

5.10. GetAllDevicesValueDescriptors

Return an array of device value descriptors.

5.10.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getAllDevicesValueDescriptors

5.10.2. Supported Formats

XML, JSON

5.10.3. Supported request methods

GET

5.10.4. Example requests

This request returns an array of SLVValueDescriptor objects.

5.10.4.1. Example with XML format :

```
 <SLVValueDescriptor-array>
  <SLVValueDescriptor>
    <name>MaintenanceButton</name>
    <label>Bouton Maintenance</label>
    <labelKey>db.failuremeaning.maintenancebutton.label</labelKey>
    <type>boolean</type>
    <failure>true</failure>
    <category>standard</category>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  <SLVValueDescriptor>
    <name>CapMode</name>
    <label>Cap Mode</label>
    <labelKey>db.meaning.capmode.label</labelKey>
    <type>boolean</type>
    <failure>true</failure>
    <category>standard</category>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  <SLVValueDescriptor>
    <name>NoLight</name>
    <label>Pas de lumière</label>
    <labelKey>db.failuremeaning.nolight.label</labelKey>
    <type>boolean</type>
    <failure>true</failure>
    <category>standard</category>
    <criticity>1</criticity>
  </SLVValueDescriptor>
  ...
</SLVValueDescriptor-array>
```

Example with JSON format :

```
[
  {
    "category" : "standard",
    "criticity" : 0,
    "dataFormat" : null,
    "failure" : true,
    "help" : null,
    "label" : "Bouton Maintenance",
    "labelKey" : "db.failuremeaning.maintenancebutton.label",
    "name" : "MaintenanceButton",
    "type" : "boolean",
    "unit" : null
  },
  {
    "category" : "standard",
    "criticity" : 0,
    "dataFormat" : null,
    "failure" : true,
    "help" : null,
    "label" : "Cap Mode",
    "labelKey" : "db.meaning.capmode.label",
    "name" : "CapMode",
    "type" : "boolean",
    "unit" : null
  },
  {
    "category" : "standard",
    "criticity" : 1,
    "dataFormat" : null,
    "failure" : true,
    "help" : null,
    "label" : "Pas de lumière",
    "labelKey" : "db.failuremeaning.nolight.label",
    "name" : "NoLight",
    "type" : "boolean",
    "unit" : null
  },
  ...
]
```

5.11. GetAllLampTypes

Return an array of all lamp types.

5.11.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getAllLampTypes

5.11.2. Supported Formats

XML, JSON

5.11.3. Supported request methods

GET

5.11.4. Example requests

This request returns an array of SLVLabelledValue objects.

5.11.4.1. Example with XML format :

```
<SLVLabelledValue-array>
  <SLVLabelledValue>
    <label>Brand 1</label>
    <value class="int">1</value>
    <properties>
      <entry>
        <string>description</string>
        <string>Brand 1</string>
      </entry>
      <entry>
        <string>lifeTime</string>
        <string>null</string>
      </entry>
    </properties>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <label>HPS 70W Ferro</label>
    <value class="int">2</value>
    <properties>
      <entry>
        <string>description</string>
        <string>HPS 70W Ferro</string>
      </entry>
      <entry>
        <string>lifeTime</string>
        <string>20000</string>
      </entry>
    </properties>
  </SLVLabelledValue>
  ...
</SLVLabelledValue-array>
```

Example with JSON format :

```
 [
  {
    "label" : "Brand 1",
    "properties" :
      {
        "description" : "Brand 1",
        "lifeTime" : "null"
      },
    "value" : 1
  },
  {
    "label" : "HPS 70W Ferro",
    "properties" :
      {
        "description" : "HPS 70W Ferro",
        "lifeTime" : "20000"
      },
    "value" : 2
  },
  ...
]
```

5.12. GetAllProviders

Return an array of all providers.

5.12.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getAllProviders

5.12.2. Supported Formats

XML, JSON

5.12.3. Supported request methods

GET

5.12.4. Example requests

This request returns an array of SLVProvider objects.

5.12.4.1. Example with XML format :

```
<SLVProvider-array>
 <SLVProvider>
   <id>3</id>
   <type>provider</type>
   <name>Providers</name>
 </SLVProvider>
 <SLVProvider>
   <id>4</id>
   <type>provider</type>
   <name>EDF</name>
   <pollution>0.11</pollution>
 </SLVProvider>
 <SLVProvider>
   <id>5</id>
   <type>provider</type>
   <name>RWE</name>
   <pollution>0.783</pollution>
 </SLVProvider>
 …
```

```
</SLVProvider-array>
```

Example with JSON format :

```
 [
  {
    "id" : 3,
    "name" : "Providers",
    "pollution" : null,
    "type" : "provider"
  },
  {
    "id" : 4,
    "name" : "EDF",
    "pollution" : 0.11,
    "type" : "provider"
  },
  {
    "id" : 5,
    "name" : "RWE",
    "pollution" : 0.783,
    "type" : "provider"
  },
  ...
]
```

5.13. GetAllValueDescriptors

Return an array of value descriptors.

5.13.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getAllValueDescriptors

5.13.2. Supported Formats

XML, JSON

5.13.3. Supported request methods

GET

5.13.4. Example requests

This request returns an array of SLVValueDescriptor objects.

5.13.4.1. Example with XML format :

```
<SLVValueDescriptor-array>
 <SLVValueDescriptor>
   <name>MaintenanceButton</name>
   <label>Bouton Maintenance</label>
   <labelKey>db.failuremeaning.maintenancebutton.label</labelKey>
   <type>boolean</type>
   <failure>true</failure>
   <category>standard</category>
   <criticity>0</criticity>
 </SLVValueDescriptor>
 <SLVValueDescriptor>
   <name>CapMode</name>
   <label>Cap Mode</label>
   <labelKey>db.meaning.capmode.label</labelKey>
   <type>boolean</type>
   <failure>true</failure>
   <category>standard</category>
   <criticity>0</criticity>
 </SLVValueDescriptor>
 <SLVValueDescriptor>
   <name>NoLight</name>
   <label>Pas de lumière</label>
   <labelKey>db.failuremeaning.nolight.label</labelKey>
   <type>boolean</type>
   <failure>true</failure>
   <category>standard</category>
   <criticity>1</criticity>
 </SLVValueDescriptor>
 ...
```

```
</SLVValueDescriptor-array>
```

Example with JSON format :

```
 [
  {
    "category" : "standard",
    "criticity" : 0,
    "dataFormat" : null,
    "failure" : true,
    "help" : null,
    "label" : "Bouton Maintenance",
    "labelKey" : "db.failuremeaning.maintenancebutton.label",
    "name" : "MaintenanceButton",
    "type" : "boolean",
    "unit" : null
  },
  {
    "category" : "standard",
    "criticity" : 0,
    "dataFormat" : null,
    "failure" : true,
    "help" : null,
    "label" : "Cap Mode",
    "labelKey" : "db.meaning.capmode.label",
    "name" : "CapMode",
    "type" : "boolean",
    "unit" : null
  },
  {
    "category" : "standard",
    "criticity" : 1,
    "dataFormat" : null,
    "failure" : true,
    "help" : null,
    "label" : "Pas de lumière",
    "labelKey" : "db.failuremeaning.nolight.label",
    "name" : "NoLight",
    "type" : "boolean",
    "unit" : null
  },
  ...
]
```

5.14. GetBatchResult

Return the batch result for the given batch identifier.

5.14.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVBatchAPI?methodName=getBatchResult

5.14.2. Supported Formats

XML, JSON

5.14.3. Supported request methods

GET

5.14.4. Parameters

Batch	Identifier of batch	Required

5.14.5. Example requests

This request returns a SLVBatchResult object.

5.14.5.1. Example with XML format :

```
 <SLVBatchResult>
  <status>OK</status>
  <errorCode>0</errorCode>
  <value class="com.dotv.streetlightserver.importer.StringsArrayImportResult">
    <running>false</running>
    <lineResults>
      <com.dotv.streetlightserver.importer.LineResult>
        <index>0</index>
        <messages>
          <string>Header Line</string>
        </messages>
        <state>1</state>
      </com.dotv.streetlightserver.importer.LineResult>
      <com.dotv.streetlightserver.importer.LineResult>
        <index>1</index>
        <messages>
          <string>Device &apos;controllerdevice&apos; found on controller &apos;MatController&apos;. Update it.</string>
          <string>Device &apos;MatController&apos; done.</string>
        </messages>
        <properties>
          <entry>
            <string>action</string>
            <string>update</string>
          </entry>
        </properties>
        <state>1</state>
      </com.dotv.streetlightserver.importer.LineResult>
      <com.dotv.streetlightserver.importer.LineResult>
        <index>2</index>
        <messages>
          <string>Device &apos;MatEm24&apos; found on controller &apos;MatController&apos;. Update it.</string>
          <string>Device &apos;New Electrical Counter&apos; done.</string>
        </messages>
        <properties>
          <entry>
            <string>action</string>
            <string>update</string>
          </entry>
        </properties>
        <state>1</state>
      </com.dotv.streetlightserver.importer.LineResult>
    </lineResults>
  </value>
  <batchRunning>false</batchRunning>
  <batchId>1329823791032</batchId>
  <batchProgressValue>100</batchProgressValue>
  <batchProgressMessage>Device &apos;New Electrical Counter&apos; done.</batchProgressMessage>
</SLVBatchResult>
```

Example with JSON format :

```
 {
   "batchId" : "1329824470491",
   "batchProgressMessage" : "Device 'controllerdevice' found on controller 'MatController'. Update it.",
   "batchProgressValue" : 66,
   "batchRunning" : true,
   "errorCode" : "0",
   "status" : "OK",
   "value" :
     {
       "running" : false
     }
}
```

5.15. GetCategories

Return an array of device categories.

5.15.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getCategories

5.15.2. Supported Formats

XML, JSON

5.15.3. Supported request methods

GET

5.15.4. Parameters

includeHidden	True to include hidden categories. True by default.	Optional

5.15.5. Example requests

This request returns an array of SLVCategory objects.

5.15.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVCategory-array>
   <com.dotv.streetlightserver.api.data.SLVCategory>
       <strId>undefined</strId>
       <displayName>Undefined</displayName>
       <hidden>false</hidden>
   </com.dotv.streetlightserver.api.data.SLVCategory>
   ...
</com.dotv.streetlightserver.api.data.SLVCategory-array>
```

Example with JSON format :

```
 [
   {
     "displayName" : "Undefined",
     "hidden" : false,
     "strId" : "undefined"
   },
   ...
]
```

5.16. GetCategoryControllerDevices

Return an array of devices for the given controller and device category.

5.16.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getCategoryControllerDevices

5.16.2. Supported Formats

XML, JSON

5.16.3. Supported request methods

GET

5.16.4. Parameters

controllerStrId	Identifier of controller	Required
categoryStrId	Identifier of category	Required

5.16.5. Example requests

This request returns an array of SLVDevice objects.

5.16.5.1. Example with XML format :

```
<SLVDevice-array>
  <SLVDevice>
    <id>15</id>
    <type>device</type>
    <name>SLV_olc7030.Advert switch</name>
    <controllerStrId>SLV_Demo</controllerStrId>
    <idOnController>SLV_olc7030.Advert switch</idOnController>
    <categoryStrId>switch</categoryStrId>
    <nodeTypeStrId>philips.llc7030.dali.two+fpm</nodeTypeStrId>
    <technologyStrId>lonworks</technologyStrId>
    <modelName>PHILIPS LLC7030/DALI With Two Outputs</modelName>
    <functionId>lampActuator</functionId>
    <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
    <lat>48.873102</lat>
    <lng>2.322765</lng>
    <properties>
      <SLVKeyValuePair>
        <key>power</key>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>powerCorrection</key>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DimmingGroupName</key>
        <value class="string">Luminaires</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.MacAddress</key>
        <value class="string">0501ACDC7A00</value>
      </SLVKeyValuePair>
    </properties>
  </SLVDevice>
  ...
</SLVDevice-array>
```

Example with JSON format :

```
 [
  {
    "address" : null,
    "categoryStrId" : "controllerdevice",
    "controllerStrId" : "SC_of_zone.2[1]",
    "functionId" : null,
    "geoZoneNamesPath" : "Some European Cities/World-Wide Demo Centers/City of Lyon/Lyon 4ème Arrondissement/Croix Rousse",
    "id" : 6,
    "idOnController" : "controllerdevice",
    "lat" : null,
    "lng" : null,
    "modelName" : null,
    "name" : "SC_of_zone.2[1]",
    "nodeTypeStrId" : null,
    "properties" :
      [
        {
          "key" : "power",
          "value" : null
        },
        {
          "key" : "powerCorrection",
          "value" : null
        },
        {
          "key" : "userproperty.DigitalInput1Info",
          "value" : null
        },
        {
          "key" : "userproperty.DigitalInput2Info",
          "value" : null
        },
        {
          "key" : "userproperty.DigitalOutput1Info",
          "value" : null
        },
        {
          "key" : "userproperty.DigitalOutput2Info",
          "value" : null
        },
        {
          "key" : "userproperty.Latitude",
          "value" : null
        },
        {
          "key" : "userproperty.Longitude",
          "value" : null
        },
        {
          "key" : "userproperty.MacAddress",
          "value" : null
        }
      ],
    "technologyStrId" : null,
    "type" : "device"
  },
  ...
]
```

5.17. GetControllerDevices

Return an array of all devices for the given controller.

5.17.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getControllerDevices

5.17.2. Supported Formats

XML, JSON

5.17.3. Supported request methods

GET

5.17.4. Parameters

controllerStrId	Identifier of controller	Required

5.17.5. Example requests

This request returns an array of SLVDevice objects.

5.17.5.1. Example with XML format :

```
<SLVDevice-array>
  <SLVDevice>
    <id>15</id>
    <type>device</type>
    <name>SLV_olc7030.Advert switch</name>
    <controllerStrId>SLV_Demo</controllerStrId>
    <idOnController>SLV_olc7030.Advert switch</idOnController>
    <categoryStrId>switch</categoryStrId>
    <nodeTypeStrId>philips.llc7030.dali.two+fpm</nodeTypeStrId>
    <technologyStrId>lonworks</technologyStrId>
    <modelName>PHILIPS LLC7030/DALI With Two Outputs</modelName>
    <functionId>lampActuator</functionId>
    <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
    <lat>48.873102</lat>
    <lng>2.322765</lng>
    <properties>
      <SLVKeyValuePair>
        <key>power</key>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>powerCorrection</key>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DimmingGroupName</key>
        <value class="string">Luminaires</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.MacAddress</key>
        <value class="string">0501ACDC7A00</value>
      </SLVKeyValuePair>
    </properties>
  </SLVDevice>
  ...
</SLVDevice-array>
```

Example with JSON format :

```
 [
  {
    "address" : null,
    "categoryStrId" : "streetlight",
    "controllerStrId" : "SC_of_zone.3",
    "functionId" : "Lamp",
    "geoZoneNamesPath" : "Some European Cities/World-Wide Demo Centers/City of Lyon/Lyon 6ème Arrondissement/Rue de Sèze",
    "id" : 10,
    "idOnController" : "device",
    "lat" : 48.4583333333333,
    "lng" : 1.80416666666667,
    "modelName" : null,
    "name" : "device device on SC_of_zone.3",
    "nodeTypeStrId" : "philips.llc7020",
    "properties" :
      [
        {
          "key" : "power",
          "value" : null
        },
        {
          "key" : "powerCorrection",
          "value" : null
        },
        {
          "key" : "lampTypeName",
          "value" : "Brand 1"
        },
        {
          "key" : "lampLifeTime",
          "value" : null
        },
        {
          "key" : "userproperty.DimmingGroupName",
          "value" : null
        },
        {
          "key" : "userproperty.MacAddress",
          "value" : null
        }
      ],
    "technologyStrId" : null,
    "type" : "device"
  },
  ...
]
```

5.18. GetControllerDevicesLastValuesAsArray

Return an array of device last values for the given controller.

5.18.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getControllerDevicesLastValuesAsArray

5.18.2. Supported Formats

XML, JSON

5.18.3. Supported request methods

GET

5.18.4. Parameters

controllerStrId	Identifier of controller	Required
categoryStrId	Array of identifiers of categories	Required
valueName	Array of value names	Required
concat	Array of concats	Optional
columnName	Array of column names	Optional

5.18.5. Example requests

This request returns a SLVArray object.

5.19. GetDefaultValueDescriptors

Return an array of default value descriptors for the given config file path.

5.19.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getDefaultValueDescriptors

5.19.2. Supported Formats

XML, JSON

5.19.3. Supported request methods

GET

5.19.4. Parameters

configFilePath	Full path of config file	Required

5.19.5. Example requests

This request returns an array of SLVValueDescriptor objects.

5.20. GetDevice

Return the device for the given device identifier.

5.20.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getDevice

5.20.2. Supported Formats

XML, JSON

5.20.3. Supported request methods

GET

5.20.4. Parameters

deviceId	identifier of device	Required

5.20.5. Example requests

This request returns a SLVDevice object.

5.20.5.1. Example with XML format :

```
 <SLVDevice>
    <id>10</id>
    <type>device</type>
    <name>SLV_Comtec_v1.Lamp1</name>
    <controllerStrId>SLV_Demo</controllerStrId>
    <idOnController>SLV_Comtec_v1.Lamp1</idOnController>
    <categoryStrId>streetlight</categoryStrId>
    <nodeTypeStrId>citylone2tor</nodeTypeStrId>
    <technologyStrId>lonworks</technologyStrId>
    <modelName>Citylone Double ON/OFF</modelName>
    <functionId>Lamp1</functionId>
    <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
```

```
  <lat>48.873464</lat>
  <lng>2.322021</lng>
  <properties>
    <SLVKeyValuePair>
      <key>power</key>
      <value class="float">100.0</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>powerCorrection</key>
      <value class="float">20.0</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>lampTypeName</key>
      <value class="string">Master SON 100W Plus - dim 1-10V</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>lampLifeTime</key>
      <value class="int">28800</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>userproperty.DimmingGroupName</key>
      <value class="string">Luminaires</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>userproperty.MacAddress</key>
      <value class="string">050159150800</value>
    </SLVKeyValuePair>
  </properties>
</SLVDevice>
```

Example with JSON format :

```
 {
   "address" : null,
   "categoryStrId" : "streetlight",
   "controllerStrId" : "SC_of_zone.3",
   "functionId" : "Lamp",
   "geoZoneNamesPath" : "Some European Cities/World-Wide Demo Centers/City of Lyon/Lyon 6ème Arrondissement/Rue de Sèze",
   "id" : 20,
  "idOnController" : "device[10]",
  "lat" : 48.375,
  "lng" : 1.8375,
  "modelName" : null,
  "name" : "device device[10] on SC_of_zone.3",
  "nodeTypeStrId" : "philips.llc7020",
  "properties" :
    [
      {
        "key" : "power",
        "value" : null
      },
      {
        "key" : "powerCorrection",
        "value" : null
      },
      {
        "key" : "lampTypeName",
        "value" : "Brand 1"
      },
      {
        "key" : "lampLifeTime",
        "value" : null
      },
      {
        "key" : "userproperty.DimmingGroupName",
        "value" : null
      },
      {
        "key" : "userproperty.MacAddress",
        "value" : null
      }
    ],
  "technologyStrId" : null,
  "type" : "device"
}
```

5.21. GetDevicesInBounds

Return all the devices located inside a box defined by four points as follow:

NW - NE / SW - SE

SW : South-West corner is defined by minimum latitude and minimum longitude

SE : South-East corner is defined by minimum latitude and maximum longitude

NE : North-East corner is defined by maximum latitude and maximum longitude

NW : North-West corner is defined by maximum latitude and minimum longitude

5.21.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getDevicesInBounds

5.21.2. Supported Formats

XML, JSON

5.21.3. Supported request methods

GET

5.21.4. Parameters

latMin	Minimal latitude (in degrees) of the bounding box	Required
latMax	Maximal latitude (in degrees) of the bounding box	Required
lngMin	Minimal longitude (in degrees) of the bounding box	Required
lngMax	Maximal longitude (in degrees) of the bounding box	Required

5.21.5. Example requests

This request returns an array of SLVDevice objects.

5.21.5.1. Example with XML format :

```
<SLVDevice-array>
  <SLVDevice>
```

```
    <id>1</id>
    <type>device</type>
    <name>SLV_Demo</name>
    <controllerStrId>SLV_Demo</controllerStrId>
    <idOnController>controllerdevice</idOnController>
    <categoryStrId>controllerdevice</categoryStrId>
    <address></address>
    <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
    <lat>48.872814</lat>
    <lng>2.321686</lng>
    <properties>
      <SLVKeyValuePair>
        <key>power</key>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>powerCorrection</key>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DigitalInput1Info</key>
        <value class="string"></value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DigitalInput2Info</key>
        <value class="string"></value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DigitalOutput1Info</key>
        <value class="string">Output 1</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DigitalOutput2Info</key>
        <value class="string">Output 2</value>
      </SLVKeyValuePair>
    </properties>
  </SLVDevice>
  ...
</SLVDevice-array>
```

Example with JSON format :

```
[
  {
    "address" : null,
    "categoryStrId" : "streetlight",
    "controllerStrId" : "SC_of_zone.3",
    "functionId" : "Lamp",
    "geoZoneNamesPath" : "Some European Cities/World-Wide Demo Centers/City of Lyon/Lyon 6ème Arrondissement/Rue de Sèze",
    "id" : 10,
    "idOnController" : "device",
    "lat" : 48.4583333333333,
    "lng" : 1.80416666666667,
    "modelName" : null,
    "name" : "device device on SC_of_zone.3",
    "nodeTypeStrId" : "philips.llc7020",
    "properties" :
      [
        {
          "key" : "power",
          "value" : null
        },
        {
          "key" : "powerCorrection",
          "value" : null
        },
        {
          "key" : "lampTypeName",
          "value" : "Brand 1"
        },
        {
          "key" : "lampLifeTime",
          "value" : null
        },
        {
          "key" : "userproperty.DimmingGroupName",
          "value" : null
        },
        {
          "key" : "userproperty.MacAddress",
          "value" : null
        }
      ],
    "technologyStrId" : null,
    "type" : "device"
  },
  ...
]
```

5.22. GetDevicesLastValuesAsArray

Return an array of last values for the given devices.

5.22.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getDevicesLastValuesAsArray

5.22.2. Supported Formats

XML, JSON

5.22.3. Supported request methods

GET

5.22.4. Parameters

deviceId	Array of device identifiers	Required
valueName	Array of value names	Required
concat	Array of concats	Optional
columnName	Array of column names	Optional

5.22.5. Example requests

This request returns a SLVArray object.

5.23. GetDevicesLogValuesAsArray

Return an array of log values for the given devices between date range. The log values are the last values stored in the Streetlight.Vision database.

5.23.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getDevicesLogValuesAsArray

5.23.2. Supported Formats

XML, JSON

5.23.3. Supported request methods

GET

5.23.4. Parameters

deviceId	Array of device identifiers	Required
name	Array of value names	Required
from	Date of oldest log value (dd/MM/yyyy HH:mm:ss)	Required
to	Date of newest log value (dd/MM/yyyy HH:mm:ss)	Required

5.23.5. Example requests

This request returns a SLVArray object.

5.24. GetDeviceValueDescriptors

Return an array of value descriptors for the given device.

5.24.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getDeviceValueDescriptors

5.24.2. Supported Formats

XML, JSON

5.24.3. Supported request methods

GET

5.24.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required

5.24.5. Example requests

This request returns an array of SLVValueDescriptor objects.

5.24.5.1. Example with XML format :

```
<SLVValueDescriptor-array>
 <SLVValueDescriptor>
   <name>LampFailure</name>
   <label>Panne de lampe</label>
   <labelKey>db.failuremeaning.lampfailure.label</labelKey>
   <type>boolean</type>
   <failure>true</failure>
   <category>standard</category>
   <criticity>1</criticity>
 </SLVValueDescriptor>
 <SLVValueDescriptor>
   <name>LampSwitch</name>
   <label>Allumage - Retour Etat</label>
   <labelKey>db.meaning.lampswitch.label</labelKey>
   <type>boolean</type>
   <failure>false</failure>
   <category>standard</category>
   <criticity>0</criticity>
 </SLVValueDescriptor>
 ...
```

```
</SLVValueDescriptor-array>
```

Example with JSON format :

```
 [
   {
      "category" : "standard",
      "criticity" : 1,
      "dataFormat" : null,
      "failure" : true,
      "help" : null,
      "label" : "Panne de lampe",
      "labelKey" : "db.failuremeaning.lampfailure.label",
      "name" : "LampFailure",
      "type" : "boolean",
      "unit" : null
   },
   {
      "category" : "standard",
      "criticity" : 0,
      "dataFormat" : null,
      "failure" : false,
      "help" : null,
      "label" : "Allumage - Retour Etat",
      "labelKey" : "db.meaning.lampswitch.label",
      "name" : "LampSwitch",
      "type" : "boolean",
      "unit" : null
   },
   ...
]
```

5.25. GetDistinctExistingValues

Return the distinct existing values of the given meaning.

5.25.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getDistinctExistingValues

5.25.2. Supported Formats

XML, JSON

5.25.3. Supported request methods

GET

5.25.4. Parameters

valueName	Name of the meaning	Required
quick	False to search in more tables. True by default.	Required
includeUndefinedValues	True to include undefined values. False by default.	Required

5.25.5. Example requests

This request returns an array of SLVTaggedValue objects.

5.25.5.1. Example with XML format :

```
<SLVTaggedValue-array>
 <SLVTaggedValue>
   <value class="string">Luminaires</value>
   <properties>
     <entry>
       <string>count</string>
       <string>1</string>
     </entry>
   </properties>
 </SLVTaggedValue>
 <SLVTaggedValue>
   <value class="string">dimmingGroup1</value>
   <properties>
     <entry>
       <string>count</string>
       <string>1</string>
```

```
      </entry>
    </properties>
  </SLVTaggedValue>
  <SLVTaggedValue>
    <value class="string">dimmingGroup2</value>
    <properties>
      <entry>
        <string>count</string>
        <string>1</string>
      </entry>
    </properties>
  </SLVTaggedValue>
  <SLVTaggedValue>
    <value class="string">dimmingGroup3</value>
    <properties>
      <entry>
        <string>count</string>
        <string>3</string>
      </entry>
    </properties>
  </SLVTaggedValue>
</SLVTaggedValue-array>
```

Example with JSON format :

```
 [
   {
      "properties" :
         {
            "count" : "1"
         },
      "value" : "Luminaires"
   },
   {
      "properties" :
         {
            "count" : "1"
         },
      "value" : "dimmingGroup1"
   },
  {
    "properties" :
      {
        "count" : "1"
      },
    "value" : "dimmingGroup2"
  },
  {
    "properties" :
      {
        "count" : "3"
      },
    "value" : "dimmingGroup3"
  }
]
```

5.26. GetGeoZoneChildrenGeoZones

Return an array of all sub geozones for the given geozone.

5.26.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getGeoZoneChildrenGeoZones

5.26.2. Supported Formats

XML, JSON

5.26.3. Supported request methods

GET

5.26.4. Parameters

geoZoneId	Identifier of parent geozone	Required

5.26.5. Example requests

This request returns an array of SLVGeoZone objects.

5.26.5.1. Example with XML format :

```
 <SLVGeoZone-array>
   <SLVGeoZone>
       <id>35</id>
       <type>geozone</type>
       <name>Barcelona</name>
       <namesPath>World-Wide Demo Centers/Barcelona</namesPath>
       <idsPath>69/35</idsPath>
       <childrenCount>4</childrenCount>
       <devicesCount>0</devicesCount>
       <latMax>41.439665</latMax>
       <latMin>41.324814</latMin>
       <lngMax>2.251635</lngMax>
       <lngMin>2.056388</lngMin>
     </SLVGeoZone>
     ...
</SLVGeoZone-array>
```

Example with JSON format :

```
 [
   {
     "childrenCount" : 4,
     "devicesCount" : 0,
     "id" : 35,
     "idsPath" : "69/35",
     "latMax" : 41.439665,
     "latMin" : 41.324814,
     "lngMax" : 2.251635,
     "lngMin" : 2.056388,
     "name" : "Barcelona",
     "namesPath" : "World-Wide Demo Centers/Barcelona",
     "type" : "geozone"
   },
   ...
]
```

5.27. GetGeoZoneDevices

Return an array of devices for the given geozone.

5.27.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getGeoZoneDevices

5.27.2. Supported Formats

XML, JSON

5.27.3. Supported request methods

GET

5.27.4. Parameters

geoZoneId	Identifier of geozone	Required
recurse	True to do recursive get. False by default.	Optional
categoryStrIds	Array of categoryStrId.	Optional

5.27.5. Example requests

This request returns an array of SLVDevice objects.

5.27.5.1. Example with XML format :

```
<SLVDevice-array>
  <SLVDevice>
     <id>1</id>
     <type>device</type>
     <name>SLV_Demo</name>
     <controllerStrId>SLV_Demo</controllerStrId>
     <idOnController>controllerdevice</idOnController>
     <categoryStrId>controllerdevice</categoryStrId>
     <address></address>
     <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
     <lat>48.872814</lat>
     <lng>2.321686</lng>
```

```
    <properties>
      <SLVKeyValuePair>
        <key>power</key>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>powerCorrection</key>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DigitalInput1Info</key>
        <value class="string"></value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DigitalInput2Info</key>
        <value class="string"></value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DigitalOutput1Info</key>
        <value class="string">Output 1</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DigitalOutput2Info</key>
        <value class="string">Output 2</value>
      </SLVKeyValuePair>
    </properties>
  </SLVDevice>
  ...
</SLVDevice-array>
```

Example with JSON format :

```
 [
  {
    "address" : null,
    "categoryStrId" : "streetlight",
    "controllerStrId" : "SC_of_zone.3",
    "functionId" : "Lamp",
    "geoZoneNamesPath" : "Some European Cities/World-Wide Demo Centers/City of Lyon/Lyon 6ème Arrondissement/Rue de Sèze",
    "id" : 10,
    "idOnController" : "device",
    "lat" : 48.4583333333333,
    "lng" : 1.80416666666667,
    "modelName" : null,
    "name" : "device device on SC_of_zone.3",
    "nodeTypeStrId" : "philips.llc7020",
    "properties" :
      [
        {
          "key" : "power",
          "value" : null
        },
        {
          "key" : "powerCorrection",
          "value" : null
        },
        {
          "key" : "lampTypeName",
          "value" : "Brand 1"
        },
        {
          "key" : "lampLifeTime",
          "value" : null
        },
        {
          "key" : "userproperty.DimmingGroupName",
          "value" : null
        },
        {
          "key" : "userproperty.MacAddress",
          "value" : null
        }
      ],
    "technologyStrId" : null,
    "type" : "device"
  },
  ...
]
```

5.28. GetGeoZoneDevicesLastValuesAsArray

Return an array of device last values for the given geozone.

5.28.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getGeoZoneDevicesLastValuesAsArray

5.28.2. Supported Formats

XML, JSON

5.28.3. Supported request methods

GET

5.28.4. Parameters

geoZoneId	Identier of geozone	Required
categoryStrId	Array of categories identifiers. False by default.	Required
valueName	Array of value names	Required
concat	Array of concats	Optional
columnName	Array of column names	Optional
recurse	True to do a recursive get	Optional

5.28.5. Example requests

This request returns a SLVArray object.

5.29. GetGeoZoneRoot

Return root geozone for the current logged user.

5.29.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getGeoZoneRoot

5.29.2. Supported Formats

XML, JSON

5.29.3. Supported request methods

GET

5.29.4. Example requests

This request returns a SLVGeoZone object.

5.29.4.1. Example with XML format :

```
 <SLVGeoZone>
   <id>69</id>
   <type>geozone</type>
   <name>World-Wide Demo Centers</name>
   <namesPath>World-Wide Demo Centers</namesPath>
   <idsPath>69</idsPath>
   <childrenCount>8</childrenCount>
   <devicesCount>0</devicesCount>
   <latMax>60.0</latMax>
   <latMin>10.0</latMin>
   <lngMax>126.0</lngMax>
   <lngMin>-12.0</lngMin>
</SLVGeoZone>
```

Example with JSON format :

```
 {
   "childrenCount" : 8,
   "devicesCount" : 0,
   "id" : 69,
   "idsPath" : "1",
   "latMax" : 60.0,
   "latMin" : 10.0,
   "lngMax" : 126.0,
   "lngMin" : -12.0,
   "name" : "World-Wide Demo Centers",
   "namesPath" : "World-Wide demo Centers",
   "type" : "geozone"
}
```

5.30. GetGeoZoneValueDescriptors

Return the value descriptors of the given geozone.

5.30.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=getGeoZoneValueDescriptors

5.30.2. Supported Formats

XML, JSON

5.30.3. Supported request methods

GET

5.30.4. Parameters

geoZoneId	Identifier of geozone	Required
configFilePath	Full path of config file	Optional

5.30.5. Example requests

This request returns an array of SLVValueDescriptors objects.

5.30.5.1. Example with XML format :

```
 <SLVValueDescriptor-array>
  <SLVValueDescriptor>
    <name>virtual.energy.consumption.jan</name>
    <label>Jan KWh</label>
    <labelKey>db.meaning.virtual.energy.consumption.jan.label</labelKey>
    <type>int</type>
    <failure>false</failure>
    <dataFormat class="linked-hash-map">
      <entry>
        <string>min</string>
        <string>0</string>
      </entry>
    </dataFormat>
    <category>standard</category>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  <SLVValueDescriptor>
    <name>virtual.energy.consumption.feb</name>
    <label>Feb KWh</label>
    <labelKey>db.meaning.virtual.energy.consumption.feb.label</labelKey>
    <type>int</type>
    <failure>false</failure>
    <dataFormat class="linked-hash-map">
      <entry>
        <string>min</string>
        <string>0</string>
      </entry>
    </dataFormat>
    <category>standard</category>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  <SLVValueDescriptor>
    <name>virtual.energy.consumption.mar</name>
    <label>March KWh</label>
    <labelKey>db.meaning.virtual.energy.consumption.mar.label</labelKey>
    <type>int</type>
    <failure>false</failure>
    <dataFormat class="linked-hash-map">
      <entry>
        <string>min</string>
        <string>0</string>
      </entry>
    </dataFormat>
    <category>standard</category>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  <SLVValueDescriptor>
    <name>virtual.energy.consumption.apr</name>
```

```
      <label>April KWh</label>
      <labelKey>db.meaning.virtual.energy.consumption.apr.label</labelKey>
      <type>int</type>
      <failure>false</failure>
      <dataFormat class="linked-hash-map">
        <entry>
          <string>min</string>
          <string>0</string>
        </entry>
      </dataFormat>
      <category>standard</category>
      <criticity>0</criticity>
    </SLVValueDescriptor>
    <SLVValueDescriptor>
      <name>virtual.energy.consumption.may</name>
      <label>May KWh</label>
      <labelKey>db.meaning.virtual.energy.consumption.may.label</labelKey>
      <type>int</type>
      <failure>false</failure>
      <dataFormat class="linked-hash-map">
        <entry>
          <string>min</string>
          <string>0</string>
        </entry>
      </dataFormat>
      <category>standard</category>
      <criticity>0</criticity>
    </SLVValueDescriptor>
    <SLVValueDescriptor>
      <name>virtual.energy.consumption.jun</name>
      <label>June KWHh</label>
      <labelKey>db.meaning.virtual.energy.consumption.jun.label</labelKey>
      <type>int</type>
      <failure>false</failure>
      <dataFormat class="linked-hash-map">
        <entry>
          <string>min</string>
          <string>0</string>
        </entry>
      </dataFormat>
```

```
  <category>standard</category>
  <criticity>0</criticity>
</SLVValueDescriptor>
<SLVValueDescriptor>
  <name>virtual.energy.consumption.jul</name>
  <label>July KWHh</label>
  <labelKey>db.meaning.virtual.energy.consumption.jul.label</labelKey>
  <type>int</type>
  <failure>false</failure>
  <dataFormat class="linked-hash-map">
    <entry>
      <string>min</string>
      <string>0</string>
    </entry>
  </dataFormat>
  <category>standard</category>
  <criticity>0</criticity>
</SLVValueDescriptor>
<SLVValueDescriptor>
  <name>virtual.energy.consumption.aug</name>
  <label>Aug KWh</label>
  <labelKey>db.meaning.virtual.energy.consumption.aug.label</labelKey>
  <type>int</type>
  <failure>false</failure>
  <dataFormat class="linked-hash-map">
    <entry>
      <string>min</string>
      <string>0</string>
    </entry>
  </dataFormat>
  <category>standard</category>
  <criticity>0</criticity>
</SLVValueDescriptor>
<SLVValueDescriptor>
  <name>virtual.energy.consumption.sep</name>
  <label>Sept KWh</label>
  <labelKey>db.meaning.virtual.energy.consumption.sep.label</labelKey>
  <type>int</type>
  <failure>false</failure>
  <dataFormat class="linked-hash-map">
```

```
    <entry>
      <string>min</string>
      <string>0</string>
    </entry>
  </dataFormat>
  <category>standard</category>
  <criticity>0</criticity>
</SLVValueDescriptor>
<SLVValueDescriptor>
  <name>virtual.energy.consumption.oct</name>
  <label>Oct KWh</label>
  <labelKey>db.meaning.virtual.energy.consumption.oct.label</labelKey>
  <type>int</type>
  <failure>false</failure>
  <dataFormat class="linked-hash-map">
    <entry>
      <string>min</string>
      <string>0</string>
    </entry>
  </dataFormat>
  <category>standard</category>
  <criticity>0</criticity>
</SLVValueDescriptor>
<SLVValueDescriptor>
  <name>virtual.energy.consumption.nov</name>
  <label>Nov KWh</label>
  <labelKey>db.meaning.virtual.energy.consumption.nov.label</labelKey>
  <type>int</type>
  <failure>false</failure>
  <dataFormat class="linked-hash-map">
    <entry>
      <string>min</string>
      <string>0</string>
    </entry>
  </dataFormat>
  <category>standard</category>
  <criticity>0</criticity>
</SLVValueDescriptor>
<SLVValueDescriptor>
  <name>virtual.energy.consumption.dec</name>
```

```
    <label>Dec KWh</label>
    <labelKey>db.meaning.virtual.energy.consumption.dec.label</labelKey>
    <type>int</type>
    <failure>false</failure>
    <dataFormat class="linked-hash-map">
      <entry>
        <string>min</string>
        <string>0</string>
      </entry>
    </dataFormat>
    <category>standard</category>
    <criticity>0</criticity>
  </SLVValueDescriptor>
</SLVValueDescriptor-array>
```

Example with JSON format :

```
 [
  {
    "category" : "standard",
    "criticity" : 0,
    "dataFormat" :
      {
        "min" : "0"
      },
    "failure" : false,
    "help" : null,
    "label" : "Jan KWh",
    "labelKey" : "db.meaning.virtual.energy.consumption.jan.label",
    "name" : "virtual.energy.consumption.jan",
    "type" : "int",
    "unit" : null
  },
  {
    "category" : "standard",
    "criticity" : 0,
    "dataFormat" :
      {
        "min" : "0"
      },
      "failure" : false,
      "help" : null,
      "label" : "Feb KWh",
      "labelKey" : "db.meaning.virtual.energy.consumption.feb.label",
      "name" : "virtual.energy.consumption.feb",
      "type" : "int",
      "unit" : null
    },
    {
      "category" : "standard",
      "criticity" : 0,
      "dataFormat" :
        {
          "min" : "0"
        },
      "failure" : false,
      "help" : null,
      "label" : "March KWh",
      "labelKey" : "db.meaning.virtual.energy.consumption.mar.label",
      "name" : "virtual.energy.consumption.mar",
      "type" : "int",
      "unit" : null
    },
    {
      "category" : "standard",
      "criticity" : 0,
      "dataFormat" :
        {
          "min" : "0"
        },
      "failure" : false,
      "help" : null,
      "label" : "April KWh",
      "labelKey" : "db.meaning.virtual.energy.consumption.apr.label",
      "name" : "virtual.energy.consumption.apr",
      "type" : "int",
      "unit" : null
    },
    {
      "category" : "standard",
  "criticity" : 0,
  "dataFormat" :
    {
      "min" : "0"
    },
  "failure" : false,
  "help" : null,
  "label" : "May KWh",
  "labelKey" : "db.meaning.virtual.energy.consumption.may.label",
  "name" : "virtual.energy.consumption.may",
  "type" : "int",
  "unit" : null
},
{
  "category" : "standard",
  "criticity" : 0,
  "dataFormat" :
    {
      "min" : "0"
    },
  "failure" : false,
  "help" : null,
  "label" : "June KWHh",
  "labelKey" : "db.meaning.virtual.energy.consumption.jun.label",
  "name" : "virtual.energy.consumption.jun",
  "type" : "int",
  "unit" : null
},
{
  "category" : "standard",
  "criticity" : 0,
  "dataFormat" :
    {
      "min" : "0"
    },
  "failure" : false,
  "help" : null,
  "label" : "July KWHh",
  "labelKey" : "db.meaning.virtual.energy.consumption.jul.label",
  "name" : "virtual.energy.consumption.jul",
      "type" : "int",
      "unit" : null
    },
    {
      "category" : "standard",
      "criticity" : 0,
      "dataFormat" :
        {
          "min" : "0"
        },
      "failure" : false,
      "help" : null,
      "label" : "Aug KWh",
      "labelKey" : "db.meaning.virtual.energy.consumption.aug.label",
      "name" : "virtual.energy.consumption.aug",
      "type" : "int",
      "unit" : null
    },
    {
      "category" : "standard",
      "criticity" : 0,
      "dataFormat" :
        {
          "min" : "0"
        },
      "failure" : false,
      "help" : null,
      "label" : "Sept KWh",
      "labelKey" : "db.meaning.virtual.energy.consumption.sep.label",
      "name" : "virtual.energy.consumption.sep",
      "type" : "int",
      "unit" : null
    },
    {
      "category" : "standard",
      "criticity" : 0,
      "dataFormat" :
        {
          "min" : "0"
        },
    "failure" : false,
    "help" : null,
    "label" : "Oct KWh",
    "labelKey" : "db.meaning.virtual.energy.consumption.oct.label",
    "name" : "virtual.energy.consumption.oct",
    "type" : "int",
    "unit" : null
  },
  {
    "category" : "standard",
    "criticity" : 0,
    "dataFormat" :
      {
        "min" : "0"
      },
    "failure" : false,
    "help" : null,
    "label" : "Nov KWh",
    "labelKey" : "db.meaning.virtual.energy.consumption.nov.label",
    "name" : "virtual.energy.consumption.nov",
    "type" : "int",
    "unit" : null
  },
  {
    "category" : "standard",
    "criticity" : 0,
    "dataFormat" :
      {
        "min" : "0"
      },
    "failure" : false,
    "help" : null,
    "label" : "Dec KWh",
    "labelKey" : "db.meaning.virtual.energy.consumption.dec.label",
    "name" : "virtual.energy.consumption.dec",
    "type" : "int",
    "unit" : null
  }
]
```

5.31. GetVirtualDeviceValueDescriptors

Return an array of device value descriptors.

5.31.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getVirtualDeviceValueDescriptors

5.31.2. Supported Formats

XML, JSON

5.31.3. Supported request methods

GET

5.31.4. Parameters

propertyName	Array of property names	Required
propertyValue	Array of property values	Required
configFilePath	Config file name	Optional

5.31.5. Example requests

This request returns an array of SLVValueDescriptor objects.

5.31.5.1. Example with XML format :

```
 <SLVValueDescriptor-array>
  <SLVValueDescriptor>
    <name>controllerStrId</name>
    <label>controllerStrId</label>
    <labelKey>db.meaning.controllerstrid.label</labelKey>
    <type>select</type>
    <failure>false</failure>
    <dataFormat class="linked-hash-map">
      <entry>
        <string>mandatory</string>
        <string>true</string>
      </entry>
      <entry>
        <string>select.listingApiCall</string>
        <string>SLVAssetAPI!getAllControllers</string>
      </entry>
      <entry>
        <string>select.listingApiCall.labelPropertyName</string>
        <string>name</string>
      </entry>
      <entry>
        <string>select.listingApiCall.valuePropertyName</string>
        <string>strId</string>
      </entry>
      <entry>
        <string>mainGroup</string>
        <string>lightpoint.identity</string>
      </entry>
      <entry>
        <string>mainGroup.labelKey</string>
        <string>mainGroup.lightpoint.identity.label</string>
      </entry>
      <entry>
        <string>subGroup</string>
        <string>lightpoint.identity.group1</string>
      </entry>
      <entry>
        <string>subGroup.labelKey</string>
        <string>subGroup.lightpoint.identity.group1.label</string>
      </entry>
      <entry>
        <string>mainGroup.label</string>
        <string>Identity</string>
      </entry>
      <entry>
        <string>subGroup.label</string>
        <string>Device Identity</string>
      </entry>
    </dataFormat>
    <category>internal</category>
    <criticity>0</criticity>
  </SLVValueDescriptor>
  ...
</SLVValueDescriptor-array>
```

5.32. ImportDevicesFromCsvFileAsync

Import devices from a CSV file.

5.32.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingManagementAPI?methodName=importDevicesFromCsvFileAsync

5.32.2. Supported Formats

XML, JSON

5.32.3. Supported request methods

POST

5.32.4. Parameters

csvFile	Content of the CSV file	Required
separatorChar	Separator caracter of the file	Optional
quoteChar	Quote caracter in the file	Optional
encoding	Encoding of the file	Optional
doLog	True to log the import in the database. False by default.	Optional
syncMaxRowsCount	Row count imported synchronously, the rest will be imported asynchronously. 0 by default.	Optional

5.32.5. Example requests

This request returns a SLVBatchResult object.

5.32.5.1. Example with XML format :

```
 <SLVBatchResult>
  <status>OK</status>
  <errorCode>0</errorCode>
  <value class="com.dotv.streetlightserver.importer.StringsArrayImportResult">
    <running>false</running>
    <lineResults>
      <null/>
      <null/>
      <null/>
    </lineResults>
  </value>
  <batchRunning>true</batchRunning>
  <batchId>1329753349951</batchId>
  <batchProgressValue>0</batchProgressValue>
</SLVBatchResult>
```

Example with JSON format :

```
 {
   "batchId" : "1329753734694",
   "batchProgressMessage" : null,
   "batchProgressValue" : 0,
   "batchRunning" : true,
   "errorCode" : "0",
   "status" : "OK",
   "value" :
      {
         "running" : false
      }
}
```

5.33. ImportDevicesFromSdpFileAsync

Import devices from a SDP file.

5.33.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingManagementAPI?methodName=importDevicesFromSdpFileAsync

5.33.2. Supported Formats

XML, JSON

5.33.3. Supported request methods

POST

5.33.4. Parameters

sdpFile	Content of the SDP file	Required
encoding	Encoding of the file	Optional

5.33.5. Example requests

This request returns a SLVBatchResult object.

5.33.5.1. Example with XML format :

```
 <SLVBatchResult>
  <status>OK</status>
  <errorCode>0</errorCode>
  <value class="com.dotv.streetlightserver.importer.SDPImportResult">
    <running>true</running>
    <stepResults/>
    <login>admin</login>
  </value>
  <batchRunning>true</batchRunning>
  <batchId>1329751667425</batchId>
  <batchProgressValue>-1</batchProgressValue>
</SLVBatchResult>
```

Example with JSON format :

```
 {
    "batchId" : "1329751985479",
    "batchProgressMessage" : null,
    "batchProgressValue" : -1,
    "batchRunning" : true,
    "errorCode" : "0",
    "status" : "OK",
    "value" :
      {
        "login" : "admin",
        "project" : null,
        "running" : true
      }
}
```

5.34. MoveDevices

Move location of devices on the map.

5.34.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=moveDevices

5.34.2. Supported Formats

XML, JSON

5.34.3. Supported request methods

GET

5.34.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of device dentifiers on controller	Required
latitude	Array of latitude of devices	Required
longitude	Array of longitude of devices	Required

5.34.5. Example requests

This request returns a SLVResult object.

5.34.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <status>OK</status>
  <errorCode>0</errorCode>
  <value class="object-array">
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    ...
  </value>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : "OK",
   "value" :
      [
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         ...
      ]
```

```
}
```

5.35. RenameDevices

Rename devices.

5.35.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=renameDevices

5.35.2. Supported Formats

XML, JSON

5.35.3. Supported request methods

GET

5.35.4. Parameters

controllerStrId	Array of controller identifiers	Required
idOnController	Array of device identifiers on controller	Required
userName	Array of new device names	Required

5.35.5. Example requests

This request returns a SLVResult object.

5.35.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <status>OK</status>
  <errorCode>0</errorCode>
  <value class="object-array">
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    ...
  </value>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : "OK",
   "value" :
      [
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         ...
      ]
}
```

5.36. SearchDevices

Return all devices having their name containing the freeform parameter.

5.36.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=SearchDevices

5.36.2. Supported Formats

XML, JSON

5.36.3. Supported request methods

GET

5.36.4. Parameters

freeform	Text to be searched in the device name	Required

5.36.5. Example requests

This request returns an array of SLVDevice objects.

5.36.5.1. Example with XML format :

```
<SLVDevice-array>
  <SLVDevice>
    <id>6</id>
    <type>device</type>
    <name>SLV_Selc3000</name>
    <controllerStrId>SLV_Demo</controllerStrId>
    <idOnController>SLV_Selc3000</idOnController>
    <categoryStrId>streetlight</categoryStrId>
    <nodeTypeStrId>selc.3000</nodeTypeStrId>
    <technologyStrId>lonworks</technologyStrId>
    <modelName>SELC 3000</modelName>
    <functionId>Lamp</functionId>
    <geoZoneNamesPath>GeoZones/Streetlight.Vision Lab</geoZoneNamesPath>
    <lat>48.872166</lat>
    <lng>2.322051</lng>
    <properties>
      <SLVKeyValuePair>
        <key>dimmable</key>
        <value class="boolean">true</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>power</key>
        <value class="float">70.0</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
```

```
        <key>powerCorrection</key>
        <value class="float">14.0</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>lampTypeName</key>
        <value class="string">Master SON 70W Plus - dim 1-10V</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>lampLifeTime</key>
        <value class="int">21600</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.DimmingGroupName</key>
        <value class="string">Luminaires</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>userproperty.MacAddress</key>
        <value class="string">050139F23500</value>
      </SLVKeyValuePair>
    </properties>
  </SLVDevice>
  ...
</SLVDevice-array>
```

Example with JSON format :

```
 [
  {
    "address" : null,
    "categoryStrId" : "streetlight",
    "controllerStrId" : "SC_of_zone.3",
    "functionId" : "Lamp",
    "geoZoneNamesPath" : "Some European Cities/World-Wide Demo Centers/City of Lyon/Lyon 6ème Arrondissement/Rue de Sèze",
    "id" : 10,
    "idOnController" : "device",
    "lat" : 48.4583333333333,
    "lng" : 1.80416666666667,
    "modelName" : null,
    "name" : "device device on SC_of_zone.3",
    "nodeTypeStrId" : "philips.llc7020",
    "properties" :
      [
        {
          "key" : "power",
          "value" : null
        },
        {
          "key" : "powerCorrection",
          "value" : null
        },
        {
          "key" : "lampTypeName",
          "value" : "Brand 1"
        },
        {
          "key" : "lampLifeTime",
          "value" : null
        },
        {
          "key" : "userproperty.DimmingGroupName",
          "value" : null
        },
        {
          "key" : "userproperty.MacAddress",
          "value" : null
        }
      ],
    "technologyStrId" : null,
    "type" : "device"
  },
  ...
]
```

5.37. SearchGeoZones

Return an array of geozones.

5.37.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=searchGeoZones

5.37.2. Supported Formats

XML, JSON

5.37.3. Supported request methods

GET

5.37.4. Parameters

name	Name or partial name of geozone	Required
partialMatchParam	True to active the partial search. False by default.	Optional

5.37.5. Example requests

This request returns an array of SLVGeoZone objects.

5.37.5.1. Example with XML format :

```
 <SLVGeoZone-array>
   <SLVGeoZone>
       <id>35</id>
       <type>geozone</type>
       <name>Barcelona</name>
       <namesPath>World-Wide Demo Centers/Barcelona</namesPath>
       <idsPath>69/35</idsPath>
       <childrenCount>4</childrenCount>
       <devicesCount>0</devicesCount>
       <latMax>41.439665</latMax>
       <latMin>41.324814</latMin>
       <lngMax>2.251635</lngMax>
       <lngMin>2.056388</lngMin>
   </SLVGeoZone>
   ...
</SLVGeoZone-array>
```

Example with JSON format :

```
 [
  {
     "childrenCount" : 4,
     "devicesCount" : 0,
     "id" : 35,
     "idsPath" : "69/35",
     "latMax" : 41.439665,
     "latMin" : 41.324814,
     "lngMax" : 2.251635,
     "lngMin" : 2.056388,
     "name" : "Barcelona",
     "namesPath" : "World-Wide Demo Centers/Barcelona",
     "type" : "geozone"
  },
  ...
]
```

5.38. SetDevicesGeoZone

Move the given devices into the target geozone.

5.38.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=setDevicesGeoZone

5.38.2. Supported Formats

XML, JSON

5.38.3. Supported request methods

GET

5.38.4. Parameters

controllerStrId	Array of controller identifiers	Required

idOnController	Array of device identifiers on controller	Required
geoZoneId	Identifier of target geozone	Required

5.38.5. Example requests

This request returns a SLVResult object.

5.38.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <status>OK</status>
  <errorCode>0</errorCode>
  <value class="object-array">
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    <string>OK</string>
    ...
  </value>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : "OK",
   "value" :
      [
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
         "OK",
        ...
    ]
}
```

5.39. SetDevicesValues

Update the device values.

5.39.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingManagementAPI?methodName=setDevicesValues

5.39.2. Supported Formats

XML, JSON

5.39.3. Supported request methods

GET

5.39.4. Parameters

controllerStrId	Array of controller Identifiers	Required
idOnController	Array of device identifiers on controller	Required
valueName	Array of value names	Required
value	Array of values	Required
eventTime	Event time.	Optional
doLog	True to log the value changes in the database. False by default.	Optional
createDevice	True to create the device if it doesn't exist. False by default.	Optional

5.39.5. Example requests

This request returns a SLVResult object.

5.40. SetDeviceValue

Update the device value.

5.40.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingManagementAPI?methodName=setDeviceValue

5.40.2. Supported Formats

XML, JSON

5.40.3. Supported request methods

GET

5.40.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required
valueName	Name of value	Required
value	Value	Required
eventTime	Force the datetime of the event in the database. Current datetime of server by default.	Optional
doLog	True to log the value changes in the database. False by default.	Optional

5.40.5. Example requests

This request returns a SLVResult object.

5.41. SetDeviceValues

Update the device values.

5.41.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingManagementAPI?methodName=setDeviceValues

5.41.2. Supported Formats

XML, JSON

5.41.3. Supported request methods

GET

5.41.4. Parameters

controllerStrId	Identifier of controller	Required
idOnController	Identifier of device on controller	Required
valueName	Array of value names	Required
value	Array of values	Required
eventTime	Force the datetime of the event in the database. Current datetime of server by default.	Optional
doLog	True to log the value changes in the database. False by default.	Optional

5.41.5. Example requests

This request returns a SLVResult object.

5.42. UpdateBrand

Update a brand.

5.42.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=updateBrand

5.42.2. Supported Formats

XML, JSON

5.42.3. Supported request methods

GET

5.42.4. Parameters

id	Id of the brand to update	Optional*
name	Name of the brand to update	Optional*
newName	New name of the brand	Optional
newDescription	New description of the brand	Optional
newLifeTime	New lifetime of the brand	Optional

* one at least is required

5.42.5. Example requests

This request returns a SLVLabelledValue object.

5.42.5.1. Example with XML format :

```
 <SLVLabelledValue>
  <label>updatedBrand</label>
  <value class="int">106</value>
  <properties>
    <entry>
      <string>description</string>
      <string>updatedBrand</string>
    </entry>
    <entry>
      <string>name</string>
      <string>updatedBrand</string>
    </entry>
    <entry>
      <string>lifeTime</string>
      <string>100</string>
    </entry>
  </properties>
</SLVLabelledValue>
```

Example with JSON format :

```
{
  "label" : "updatedBrand",
  "properties" :
    {
      "description" : "updatedBrand",
      "name" : "updatedBrand",
      "lifeTime" : "100"
    },
  "value" : 106
}
```

5.43. UpdateGeoZone

Update the geozone properties.

5.43.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=updateGeoZone

5.43.2. Supported Formats

XML, JSON

5.43.3. Supported request methods

GET

5.43.4. Parameters

geoZoneId	Identifier of geozone to update	Required
newName	Name of geozone	Optional
parentId	Identifier of parent geozone	Optional

latMin	Minimum latitude of geozone	Optional
latMax	Maximum latitude of geozone	Optional
lngMin	Minimum longitude of geozone	Optional
lngMax	Maximum longitude of geozone	Optional

5.43.5. Example requests

This request returns a SLVGeoZone object.

5.43.5.1. Example with XML format :

```
 <SLVGeoZone>
    <id>101</id>
    <type>geozone</type>
    <name>MyGeoZone</name>
    <namesPath>MyParentGeoZone/MyGeoZone</namesPath>
    <idsPath>100/101</idsPath>
    <childrenCount>0</childrenCount>
    <devicesCount>0</devicesCount>
    <latMax>45.799207</latMax>
    <latMin>45.708545</latMin>
    <lngMax>4.914481</lngMax>
    <lngMin>4.775693</lngMin>
</SLVGeoZone>
```

Example with JSON format :

```
 {
   "childrenCount" : 0,
   "devicesCount" : 0,
   "id" : 101,
   "idsPath" : "100/101",
   "latMax" : 45.799207,
   "latMin" : 45.708545,
   "lngMax" : 4.914481,
   "lngMin" : 4.775693,
   "name" : "MyGeoZone",
   "namesPath" : "MyParentGeoZone/MyGeoZone",
   "type" : "geozone"
```

}

5.44. UpdateProvider

Update the given provider.

5.44.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetManagementAPI?methodName=updateProvider

5.44.2. Supported Formats

XML, JSON

5.44.3. Supported request methods

GET

5.44.4. Parameters

providerId	Identifier of the provider	Required
newName	New name of the provider	Optional
newParentId	New identifier of the parent provider	Optional
pollutionRate	Pollution rate	Optional

5.44.5. Example requests

This request returns the SLVProvider object of updated provider.

5.44.5.1. Example with XML format :

```
 <SLVProvider>
  <id>20</id>
  <type>provider</type>
  <name>updatedProvider</name>
  <pollution>0.179</pollution>
</SLVProvider>
```

Example with JSON format :

```
 {
   "id" : 20,
   "name" : "updatedProvider",
   "pollution" : 0.179,
   "type" : "provider"
}
```

6. Report

6.1. CreateScheduledReportTemplateDefinition

Create a new scheduled report definition.

6.1.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingManagementAPI?methodName=createScheduledReportTemplateDefinition

6.1.2. Supported Formats

XML, JSON

6.1.3. Supported request methods

GET

6.1.4. Parameters

id	Identifier of the scheduled report definition	Required
groupId	Identifier of the parent geozone	Required
templateImplClassName	Class name of the scheduled report template	Required

6.1.5. Example requests

This request returns a SLVScheduledReportDefinition object.

6.1.5.1. Example with XML format :

```
 <SLVScheduledReportDefinition>
  <id>demoTemplate</id>

<scheduledReportTemplateImplClassName>com.dotv.streetlightserver.plugin.scheduledreports.impl.failures.DailyFailuresReportTemplate</scheduledReportTemplateImplClassName>
  <geoZoneId>334</geoZoneId>
  <propertyValues>
    <entry>
      <string>minute</string>
      <int>0</int>
    </entry>
    <entry>
      <string>hourOfDay</string>
      <int>10</int>
    </entry>
    <entry>
      <string>ftpFileName</string>
      <string>failures</string>
    </entry>
  </propertyValues>
</SLVScheduledReportDefinition>
```

Example with JSON format :

```
 {
   "geoZoneId" : 391,
   "id" : "demoTemplate",
   "propertyValues" :
      {
         "minute" : 0,
         "downMeaningStrIds" :
            [
               "Default0",
               "Default13",
               "Default11",
               "Default5",
               "Default8",
               "DefaultLostNode",
               "Default10",
               "Default12",
               "Default3",
               "Default9"
            ],
         "unitName" : "LIGHTING COLUMN",
         "customerShortCode" : "SLV",
         "customerFullName" : "SLV",
         "serviceCode" : "SLVSymologyReport",
         "contactType" : "SLV",
         "ftpDirectory" : "",
         "webCustomerId" : "SLVWebPortal",
         "hourOfDay" : 13
      },
   "scheduledReportTemplateImplClassName" : "com.dotv.streetlightserver.plugin.scheduledreports.impl.symology.SymologyScheduledReportTemplate"
 }
```

6.2. DeleteScheduledReportDefinition

Delete the given scheduled report definition.

6.2.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingManagementAPI?methodName=deleteScheduledReportDefinition

6.2.2. Supported Formats

XML, JSON

6.2.3. Supported request methods

GET

6.2.4. Parameters

id	Identifier of the scheduled report definition	Required

6.2.5. Example requests

This request returns a SLVResult object.

6.2.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
  <status>OK</status>
  <errorCode>0</errorCode>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
   "errorCode" : "0",
   "status" : "OK",
   "value" : null
}
```

6.3. GetAllScheduledReportTemplateDefinitions

Return all the scheduled report template definitions.

6.3.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingManagementAPI?methodName=getAllScheduledReportTemplateDefinitions

6.3.2. Supported Formats

XML, JSON

6.3.3. Supported request methods

GET

6.3.4. Parameters

includeHidden	True to do recursive get. False by default.	Optional
propertyDescriptors	True to do recursive get. False by default.	Optional

6.3.5. Example requests

This request returns an array of SLVScheduledReportDefinition objects.

6.3.5.1. Example with XML format :

```
 <SLVScheduledReportDefinition-array>
  <SLVScheduledReportDefinition>
```

```
    <id>Conso hebdo</id>
<scheduledReportTemplateImplClassName>com.dotv.streetlightserver.plugin.scheduledreports.impl.energy.WeeklyEnergyReportTemplate</scheduledReportTemplateImplClassName>
    <geoZoneId>66</geoZoneId>
    <propertyValues>
      <entry>
        <string>minute</string>
        <int>0</int>
      </entry>
      <entry>
        <string>recipients</string>
        <string-array>
          <string>gwaroquier@streetlight-vision.com</string>
        </string-array>
      </entry>
      <entry>
        <string>description</string>
        <string></string>
      </entry>
      <entry>
        <string>periodStrId</string>
        <string>every.sunday</string>
      </entry>
      <entry>
        <string>htmlFormat</string>
        <boolean>true</boolean>
      </entry>
      <entry>
        <string>from</string>
        <string>gw_head_full@streetlight-vision.com</string>
      </entry>
      <entry>
        <string>hourOfDay</string>
        <int>0</int>
      </entry>
    </propertyValues>
  </SLVScheduledReportDefinition>
</SLVScheduledReportDefinition-array>
```

Example with JSON format :

```
 [
  {
    "label" : "Rapport UMSO",
    "properties" : null,
    "value" : "com.dotv.streetlightserver.plugin.scheduledreports.impl.umso.UmsoScheduledReportTemplate"
  },
  {
    "label" : "Rapport de consommation électrique hebdomadaire",
    "properties" : null,
    "value" : "com.dotv.streetlightserver.plugin.scheduledreports.impl.energy.WeeklyEnergyReportTemplate"
  },
  {
    "label" : "Rapport de panne quotidien",
    "properties" : null,
    "value" : "com.dotv.streetlightserver.plugin.scheduledreports.impl.failures.DailyFailuresReportTemplate"
  },
  {
    "label" : "Rapport Symology",
    "properties" : null,
    "value" : "com.dotv.streetlightserver.plugin.scheduledreports.impl.symology.SymologyScheduledReportTemplate"
  }
]
```

6.4. GetDevicesLastValues

Return an array of last values for the given devices. The last values are read on the controller in realtime.

6.4.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getDevicesLastValues

6.4.2. Supported Formats

XML, JSON

6.4.3. Supported request methods

GET

6.4.4. Parameters

deviceId	Array of device identifiers	Required
valueName	Array of value names	Required

6.4.5. Example requests

This request returns an array of SLVValue objects.

6.4.5.1. Example with XML format :

```
 <SLVValue-array>
  <SLVValue>
    <name>LowCurrent</name>
    <deviceId>76</deviceId>
    <controllerStrId>SC_of_zone.2</controllerStrId>
    <idOnController>device</idOnController>
    <info>null</info>
  </SLVValue>
  <SLVValue>
    <name>LowCurrent</name>
    <deviceId>1929</deviceId>
    <controllerStrId>SLV_Shanghai</controllerStrId>
    <idOnController>SLV_Comtec_v2.Lamp</idOnController>
    <info>null</info>
  </SLVValue>
  ...
<SLVValue-array>
```

Example with JSON format :

```
 [
   {
      "controllerStrId" : "SC_of_zone.2",
      "deviceId" : 76,
      "eventTime" : null,
      "idOnController" : "device",
      "info" : "null",
      "meaningLabel" : null,
      "name" : "LowVoltage",
      "updateTime" : null,
      "value" : null
   },
   {
      "controllerStrId" : "SLV_Shanghai",
      "deviceId" : 1929,
      "eventTime" : null,
      "idOnController" : "SLV_Comtec_v2.Lamp",
      "info" : "null",
      "meaningLabel" : null,
      "name" : "LowVoltage",
      "updateTime" : null,
      "value" : null
   },
   ...
 ]
```

6.5. GetDevicesLogValues

Return an array of log values for the given devices between date range. The log values are the last values stored in the Streetlight.Vision database.

6.5.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVLoggingAPI?methodName=getDevicesLogValues

6.5.2. Supported Formats

XML, JSON

6.5.3. Supported request methods

GET

6.5.4. Parameters

deviceId	Array of device identifiers	Required
name	Array of value names	Required
from	Date of oldest log value (dd/MM/yyyy HH:mm:ss)	Required
to	Date of newest log value (dd/MM/yyyy HH:mm:ss)	Required

6.5.5. Example requests

This request returns an array of SLVValue objects.

6.5.5.1. Example with XML format :

```
 <SLVValue-array>
  <SLVValue>
    <name>LampLevel</name>
    <deviceId>9399</deviceId>
    <controllerStrId>FR-ZacInnovia-A2</controllerStrId>
    <idOnController>A2-D3-12.Lamp1</idOnController>
    <eventTime class="sql-timestamp">2012-02-14 18:09:45.0</eventTime>
    <updateTime class="sql-timestamp">2012-02-15 05:16:58.0</updateTime>
    <value class="float">100.0</value>
    <info></info>
  </SLVValue>
  <SLVValue>
    <name>LampLevel</name>
    <deviceId>9399</deviceId>
    <controllerStrId>FR-ZacInnovia-A2</controllerStrId>
    <idOnController>A2-D3-12.Lamp1</idOnController>
    <eventTime class="sql-timestamp">2012-02-16 18:12:04.0</eventTime>
    <updateTime class="sql-timestamp">2012-02-17 05:01:55.0</updateTime>
    <value class="float">100.0</value>
    <info></info>
  </SLVValue>
</SLVValue-array>
```

Example with JSON format :

```
[
  {
    "controllerStrId" : "Cimetiere_ILON1",
    "deviceId" : 7219,
    "eventTime" : "2012-02-14T05:18:32.000",
    "idOnController" : "BO11S020.Lamp1",
    "info" : "",
    "meaningLabel" : null,
    "name" : "LampLevel",
    "updateTime" : "2012-02-14T18:24:25.000",
    "value" : 100.0
  },
  {
    "controllerStrId" : "Cimetiere_ILON1",
    "deviceId" : 7219,
    "eventTime" : "2012-02-14T22:52:27.000",
    "idOnController" : "BO11S020.Lamp1",
    "info" : "",
    "meaningLabel" : null,
    "name" : "LampLevel",
    "updateTime" : "2012-02-14T23:19:48.000",
    "value" : 75.0
  },
  {
    "controllerStrId" : "Cimetiere_ILON1",
    "deviceId" : 7219,
    "eventTime" : "2012-02-15T05:09:07.000",
    "idOnController" : "BO11S020.Lamp1",
    "info" : "",
    "meaningLabel" : null,
    "name" : "LampLevel",
    "updateTime" : "2012-02-15T18:26:35.000",
    "value" : 100.0
  },
  {
    "controllerStrId" : "Cimetiere_ILON1",
    "deviceId" : 7219,
    "eventTime" : "2012-02-15T22:52:34.000",
    "idOnController" : "BO11S020.Lamp1",
    "info" : "",
    "meaningLabel" : null,
    "name" : "LampLevel",
    "updateTime" : "2012-02-16T18:27:24.000",
    "value" : 75.0
  },
  {
    "controllerStrId" : "Cimetiere_ILON1",
    "deviceId" : 7219,
    "eventTime" : "2012-02-16T05:27:43.000",
    "idOnController" : "BO11S020.Lamp1",
    "info" : "",
    "meaningLabel" : null,
    "name" : "LampLevel",
    "updateTime" : "2012-02-16T18:27:28.000",
    "value" : 100.0
  },
  {
    "controllerStrId" : "Cimetiere_ILON1",
    "deviceId" : 7219,
    "eventTime" : "2012-02-16T23:03:53.000",
    "idOnController" : "BO11S020.Lamp1",
    "info" : "",
    "meaningLabel" : null,
    "name" : "LampLevel",
    "updateTime" : "2012-02-17T18:31:01.000",
    "value" : 75.0
  },
  {
    "controllerStrId" : "Cimetiere_ILON1",
    "deviceId" : 7219,
    "eventTime" : "2012-02-17T05:12:54.000",
    "idOnController" : "BO11S020.Lamp1",
    "info" : "",
    "meaningLabel" : null,
    "name" : "LampLevel",
    "updateTime" : "2012-02-17T18:31:02.000",
    "value" : 100.0
  },
  {
        "controllerStrId" : "Cimetiere_ILON1",
        "deviceId" : 7219,
        "eventTime" : "2012-02-17T22:33:46.000",
        "idOnController" : "BO11S020.Lamp1",
        "info" : "",
        "meaningLabel" : null,
        "name" : "LampLevel",
        "updateTime" : "2012-02-18T00:00:51.000",
        "value" : 75.0
    }
]
```

6.6. GetGroupEnergy

Return the energy consumption for the given geozone between date range.

6.6.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingAPI?methodName=getGroupEnergy

6.6.2. Supported Formats

XML, JSON

6.6.3. Supported request methods

GET

6.6.4. Parameters

groupId	Identifier of geozone	Required
from	Oldest datetime (dd/MM/yyyy HH:mm:ss)	Required
to	Newest datetime (dd/MM/yyyy HH:mm:ss)	Required

6.6.5. Example requests

This request returns a Float object.

6.7. GetGroupEnergySaved

Computes the energy consumed and the energy saved comparing to a theorical energy consumption for a given geozone between date range.

6.7.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingAPI?methodName=getGroupEnergySaved

6.7.2. Supported Formats

XML, JSON

6.7.3. Supported request methods

GET

6.7.4. Parameters

groupId	Identifier of geozone	Required
from	Oldest datetime (dd/MM/yyyy HH:mm:ss)	Required
to	Newest datetime (dd/MM/yyyy HH:mm:ss)	Required

6.7.5. Example requests

This request returns an array of Float objects.

6.8. GetRegisteredScheduledReportTemplateImplementations

Return the scheduled reports templates.

6.8.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingManagementAPI?methodName=getRegisteredScheduledReportTemplateImplementations

6.8.2. Supported Formats

XML, JSON

6.8.3. Supported request methods

GET

6.8.4. Example requests

This request returns an array of SLVLabelledValue objects.

6.8.4.1. Example with XML format :

```
 <list>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.scheduledreports.impl.umso.UmsoScheduledReportTemplate</value>
    <label>Rapport UMSO</label>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.scheduledreports.impl.energy.WeeklyEnergyReportTemplate</value>
    <label>Rapport de consommation électrique hebdomadaire</label>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.scheduledreports.impl.failures.DailyFailuresReportTemplate</value>
    <label>Rapport de panne quotidien</label>
  </SLVLabelledValue>
  <SLVLabelledValue>
    <value class="string">com.dotv.streetlightserver.plugin.scheduledreports.impl.symology.SymologyScheduledReportTemplate</value>
    <label>Rapport Symology</label>
  </SLVLabelledValue>
</list>
```

Example with JSON format :

```
 [
   {
      "label" : "Rapport UMSO",
      "properties" : null,
      "value"                                                                                                        :
"com.dotv.streetlightserver.plugin.scheduledreports.impl.umso.UmsoScheduledReportTemplate"
   },
   {
      "label" : "Rapport de consommation électrique hebdomadaire",
      "properties" : null,
      "value"                                                                                                        :
"com.dotv.streetlightserver.plugin.scheduledreports.impl.energy.WeeklyEnergyReportTemplate"
   },
   {
      "label" : "Rapport de panne quotidien",
      "properties" : null,
      "value"                                                                                                        :
"com.dotv.streetlightserver.plugin.scheduledreports.impl.failures.DailyFailuresReportTemplate"
   },
   {
      "label" : "Rapport Symology",
      "properties" : null,
      "value"                                                                                                        :
"com.dotv.streetlightserver.plugin.scheduledreports.impl.symology.SymologyScheduledReportTemplat
e"
   }
]
```

6.9. GetReport

Return the report for the given geozone and report type.

The list of available reports is the following :

FailuresApplicationReportBuilder

EnergyApplicationReportBuilder

BusinessIndicatorsApplicationReportBuilder

ForecastApplicationReport

6.9.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingAPI?methodName=getReport

6.9.2. Supported Formats

XML, JSON

6.9.3. Supported request methods

GET

6.9.4. Parameters

groupId	Identifier of geozone	Required
reportBuilderClassName	Name of report type	Required
reportBuilderPropertyName	Array of report property names	Required
reportBuilderPropertyValue	Array of report property values	Required

6.9.5. Example requests

This request returns a SLVReport object.

6.9.5.1. Example with XML format :

```
<SLVReport>
 <properties>
   <entry>
     <string>title</string>
     <string>Rapport des défauts</string>
   </entry>
   <entry>
     <string>bottomRow</string>
     <SLVLabelledValue>
       <label>GeoZones</label>
       <value class="object-array">
         <string>23</string>
```

```
      <string>0</string>
      <string>0.0</string>
      <string>0</string>
      <string>0.0</string>
      <string>0</string>
      <string>0.0</string>
    </value>
  </SLVLabelledValue>
</entry>
<entry>
  <string>columnNames</string>
  <string-array>
    <string>DevicesCountColumn</string>
    <string>FaultyDevicesCountColumn</string>
    <string>FaultyDeviceRatioColumn</string>
    <string>FaultyDevicesCountColumn</string>
    <string>FaultyDeviceRatioColumn</string>
    <string>CriticalRunningHoursDevicesCountColumn</string>
    <string>CriticalRunningHoursDeviceRatioColumn</string>
  </string-array>
</entry>
<entry>
  <string>rows</string>
  <SLVLabelledValue-array>
    <SLVLabelledValue>
      <label>TestBenchOthers402055</label>
      <value class="object-array">
        <string>9</string>
        <string>0</string>
        <string>0.0</string>
        <string>0</string>
        <string>0.0</string>
        <string>0</string>
        <string>0.0</string>
      </value>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <label>TestBenchPhilips402055</label>
      <value class="object-array">
        <string>9</string>
```

```
            <string>0</string>
            <string>0.0</string>
            <string>0</string>
            <string>0.0</string>
            <string>0</string>
            <string>0.0</string>
          </value>
        </SLVLabelledValue>
      </SLVLabelledValue-array>
    </entry>
    <entry>
      <string>columnLabels</string>
      <string-array>
        <string>Nombre</string>
        <string>Alerte</string>
        <string>Alerte %</string>
        <string>Pannes</string>
        <string>Pannes %</string>
        <string>Lampes à changer #</string>
        <string>Lampes à changer %</string>
      </string-array>
    </entry>
    <entry>
      <string>metadatas</string>
      <SLVLabelledValue-array>
        <SLVLabelledValue>
          <label>Zone :</label>
          <value class="string">GeoZones</value>
        </SLVLabelledValue>
        <SLVLabelledValue>
          <label>Date :</label>
          <value class="string">05/07/2011 14:17</value>
        </SLVLabelledValue>
      </SLVLabelledValue-array>
    </entry>
  </properties>
</SLVReport>
```

Example with JSON format :

```
{
  "properties" :
    {
      "title" : "Rapport des défauts",
      "bottomRow" :
        {
          "label" : "TOTAL",
          "properties" : null,
          "value" :
            [
              "0",
              "0",
              null,
              null,
              null,
              null,
              null,
              "?",
              null
            ]
        },
      "columnNames" :
        [
          "faulty",
          "criticalFaulty",
          "failures",
          "eventTime",
          "runningHours",
          "runningHoursLifeTimePercent",
          "name",
          "address",
          "geozone"
        ],
      "rows" :
        [
          {
            "label" : "AbelVigilonA.Lamp",
            "properties" : null,
            "value" :
              [
      null,
      null,
      null,
      null,
      null,
      null,
      "AbelVigilonA.Lamp",
      "?",
      "TestBenchOthers402055"
    ]
},
{
  "label" : "CityloneDALI.Lamp1",
  "properties" : null,
  "value" :
    [
      null,
      null,
      null,
      null,
      null,
      null,
      "CityloneDALI.Lamp1",
      "?",
      "TestBenchOthers402055"
    ]
},
{
  "label" : "CityloneDALI.Lamp2",
  "properties" : null,
  "value" :
    [
      null,
      null,
      null,
      null,
      null,
      null,
      "CityloneDALI.Lamp2",
      "?",
      "TestBenchOthers402055"
    ]
},
{
  "label" : "Selc2000Combi",
  "properties" : null,
  "value" :
    [
      null,
      null,
      null,
      null,
      null,
      null,
      "Selc2000Combi",
      "?",
      "TestBenchOthers402055"
    ]
},
{
  "label" : "Selc3000",
  "properties" : null,
  "value" :
    [
      null,
      null,
      null,
      null,
      null,
      null,
      "Selc3000",
      "?",
      "TestBenchOthers402055"
    ]
},
{
  "label" : "Philips7020",
  "properties" : null,
  "value" :
    [
      null,
      null,
      null,
      null,
      null,
      null,
      "Philips7020",
      "?",
      "TestBenchOthers402055"
    ]
},
{
  "label" : "Philips7030.OLC[1]",
  "properties" : null,
  "value" :
    [
      null,
      null,
      null,
      null,
      null,
      null,
      "Philips7030.OLC[1]",
      "?",
      "TestBenchOthers402055"
    ]
},
{
  "label" : "Philips7035_2.OLC[1]",
  "properties" : null,
  "value" :
    [
      null,
      null,
      null,
      null,
      null,
      null,
      "Philips7035_2.OLC[1]",
      "?",
          "TestBenchOthers402055"
        ]
    },
    {
      "label" : "Philips7045.OLC[0]",
      "properties" : null,
      "value" :
        [
          null,
          null,
          null,
          null,
          null,
          null,
          "Philips7045.OLC[0]",
          "?",
          "TestBenchOthers402055"
        ]
    }
  ],
"columnLabels" :
  [
    "Alerte",
    "Pannes",
    "Pannes",
    "Date",
    "Nombre d'heures d'allumage",
    "Pourcentage",
    "Nom",
    "Adresse",
    "Zone"
  ],
"metadatas" :
  [
    {
      "label" : "Zone :",
      "properties" : null,
      "value" : "TestBenchOthers402055"
    },
    {
          "label" : "Date :",
          "properties" : null,
          "value" : "06/07/2011 00:45"
        },
        {
          "label" : "Equipement :",
          "properties" : null,
          "value" : "9"
        }
      ]
    }
}
```

6.10. GetScheduledReportTemplateDefinition

Return a scheduled report template definition.

6.10.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingManagementAPI?methodName=getScheduledReportTemplateDefinition

6.10.2. Supported Formats

XML, JSON

6.10.3. Supported request methods

GET

6.10.4. Parameters

scheduledReportTemplateId	String identifier of the scheduled report	Required

6.10.5. Example requests

This request returns a SLVScheduledReportDefinition object.

6.10.5.1. Example with XML format :

```
 <SLVScheduledReportDefinition>
  <id>Conso hebdo</id>
<scheduledReportTemplateImplClassName>com.dotv.streetlightserver.plugin.scheduledreports.impl.energy.WeeklyEnergyReportTemplate</scheduledReportTemplateImplClassName>
  <geoZoneId>66</geoZoneId>
  <propertyValues>
    <entry>
      <string>minute</string>
      <int>0</int>
    </entry>
    <entry>
      <string>recipients</string>
      <string-array>
        <string>user@streetlight-vision.com</string>
      </string-array>
    </entry>
    <entry>
      <string>description</string>
      <string></string>
    </entry>
    <entry>
      <string>periodStrId</string>
      <string>every.sunday</string>
    </entry>
    <entry>
      <string>htmlFormat</string>
      <boolean>true</boolean>
    </entry>
    <entry>
      <string>from</string>
      <string>portal@streetlight-vision.com</string>
    </entry>
    <entry>
      <string>hourOfDay</string>
      <int>0</int>
    </entry>
  </propertyValues>
</SLVScheduledReportDefinition>
```

Example with JSON format :

```
{
  "geoZoneId" : 66,
  "id" : "Conso hebdo",
  "propertyValues" :
    {
      "minute" : 0,
      "recipients" :
        [
          "user@streetlight-vision.com"
        ],
      "description" : "",
      "periodStrId" : "every.sunday",
      "htmlFormat" : true,
      "from" : "portal@streetlight-vision.com",
      "hourOfDay" : 0
    },
  "scheduledReportTemplateImplClassName" : "com.dotv.streetlightserver.plugin.scheduledreports.impl.energy.WeeklyEnergyReportTemplate"
}
```

6.11. GetScheduledReportTemplateEditablePropertyDescriptors

Return a scheduled report template editable property descriptors.

6.11.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingManagementAPI?methodName=getScheduledReportTemplateEditablePropertyDescriptors

6.11.2. Supported Formats

XML, JSON

6.11.3. Supported request methods

GET

6.11.4. Parameters

scheduledReportTemplateId	String identifier of the scheduled report	Required

6.11.5. Example requests

This request returns an array of SLVPropertyDescriptor objects.

6.11.5.1. Example with XML format :

```
<SLVPropertyDescriptor-array>
 <SLVPropertyDescriptor>
   <name>description</name>
   <label>description</label>
   <type>string</type>
 </SLVPropertyDescriptor>
 <SLVPropertyDescriptor>
   <name>periodStrId</name>
   <label>Périodicité</label>
   <type>select</type>
   <enumValues class="SLVLabelledValue-array">
     <SLVLabelledValue>
       <value class="string">every.day</value>
       <label>Chaque jour</label>
     </SLVLabelledValue>
     <SLVLabelledValue>
       <value class="string">every.monday</value>
       <label>Chaque Lundi</label>
     </SLVLabelledValue>
     <SLVLabelledValue>
       <value class="string">every.tuesday</value>
       <label>Chaque Mardi</label>
     </SLVLabelledValue>
     <SLVLabelledValue>
       <value class="string">every.wednesday</value>
       <label>Chaque Mercredi</label>
     </SLVLabelledValue>
     <SLVLabelledValue>
       <value class="string">every.thursday</value>
```

```
      <label>Chaque Jeudi</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="string">every.friday</value>
      <label>Chaque Vendredi</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="string">every.saturday</value>
      <label>Chaque Samedi</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="string">every.sunday</value>
      <label>Chaque Dimanche</label>
    </SLVLabelledValue>
  </enumValues>
</SLVPropertyDescriptor>
<SLVPropertyDescriptor>
  <name>hourOfDay</name>
  <label>Heure</label>
  <type>select</type>
  <enumValues class="SLVLabelledValue-array">
    <SLVLabelledValue>
      <value class="int">0</value>
      <label>00</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="int">1</value>
      <label>01</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="int">2</value>
      <label>02</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="int">3</value>
      <label>03</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="int">4</value>
      <label>04</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">5</value>
        <label>05</label>
      </SLVLabelledValue>
      ...
      <SLVLabelledValue>
        <value class="int">23</value>
        <label>23</label>
      </SLVLabelledValue>
    </enumValues>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>minute</name>
    <label>Minute</label>
    <type>select</type>
    <enumValues class="SLVLabelledValue-array">
      <SLVLabelledValue>
        <value class="int">0</value>
        <label>00</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">1</value>
        <label>01</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">2</value>
        <label>02</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">3</value>
        <label>03</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">4</value>
        <label>04</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">5</value>
        <label>05</label>
      </SLVLabelledValue>
      ...
      <SLVLabelledValue>
        <value class="int">59</value>
        <label>59</label>
      </SLVLabelledValue>
    </enumValues>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>from</name>
    <label>De</label>
    <type>select</type>
    <enumValues class="SLVLabelledValue-array">
      <SLVLabelledValue>
        <value class="string">user@streetlight-vision.com</value>
        <label>User SLV &lt;user@streetlight-vision.com&gt;</label>
      </SLVLabelledValue>
    </enumValues>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>recipients</name>
    <label>Vers</label>
    <type>select</type>
    <enumValues class="SLVLabelledValue-array">
      <SLVLabelledValue>
        <value class="string">user@streetlight-vision.com</value>
        <label>User SLV &lt;user@streetlight-vision.com&gt;</label>
      </SLVLabelledValue>
    </enumValues>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>htmlFormat</name>
    <label>Format HTML</label>
    <type>boolean</type>
  </SLVPropertyDescriptor>
</SLVPropertyDescriptor-array>
```

Example with JSON format :

```
 [
{
  "enumValues" : null,
  "format" : null,
  "label" : "description",
  "labelKey" : null,
  "name" : "description",
  "type" : "string"
},
{
  "enumValues" :
    [
      {
        "label" : "Chaque jour",
        "properties" : null,
        "value" : "every.day"
      },
      {
        "label" : "Chaque Lundi",
        "properties" : null,
        "value" : "every.monday"
      },
      {
        "label" : "Chaque Mardi",
        "properties" : null,
        "value" : "every.tuesday"
      },
      {
        "label" : "Chaque Mercredi",
        "properties" : null,
        "value" : "every.wednesday"
      },
      {
        "label" : "Chaque Jeudi",
        "properties" : null,
        "value" : "every.thursday"
      },
      {
        "label" : "Chaque Vendredi",
        "properties" : null,
        "value" : "every.friday"
```

```
            },
            {
              "label" : "Chaque Samedi",
              "properties" : null,
              "value" : "every.saturday"
            },
            {
              "label" : "Chaque Dimanche",
              "properties" : null,
              "value" : "every.sunday"
            }
          ],
        "format" : null,
        "label" : "Périodicité",
        "labelKey" : null,
        "name" : "periodStrId",
        "type" : "select"
      },
      {
        "enumValues" :
          [
            {
              "label" : "00",
              "properties" : null,
              "value" : 0
            },
            {
              "label" : "01",
              "properties" : null,
              "value" : 1
            },
            {
              "label" : "02",
              "properties" : null,
              "value" : 2
            },
            {
              "label" : "03",
              "properties" : null,
              "value" : 3
          },
          {
            "label" : "04",
            "properties" : null,
            "value" : 4
          },
          {
            "label" : "05",
            "properties" : null,
            "value" : 5
          },
          ...
          {
            "label" : "23",
            "properties" : null,
            "value" : 23
          }
        ],
      "format" : null,
      "label" : "Heure",
      "labelKey" : null,
      "name" : "hourOfDay",
      "type" : "select"
    },
    {
      "enumValues" :
        [
          {
            "label" : "00",
            "properties" : null,
            "value" : 0
          },
          {
            "label" : "01",
            "properties" : null,
            "value" : 1
          },
          {
            "label" : "02",
            "properties" : null,
          "value" : 2
        },
        {
          "label" : "03",
          "properties" : null,
          "value" : 3
        },
        {
          "label" : "04",
          "properties" : null,
          "value" : 4
        },
        {
          "label" : "05",
          "properties" : null,
          "value" : 5
        },
        ...
        {
          "label" : "59",
          "properties" : null,
          "value" : 59
        }
      ],
    "format" : null,
    "label" : "Minute",
    "labelKey" : null,
    "name" : "minute",
    "type" : "select"
  },
  {
    "enumValues" :
      [
        {
          "label" : "User SLV <user@streetlight-vision.com>",
          "properties" : null,
          "value" : "user@streetlight-vision.com"
        }
      ],
    "format" : null,
    "label" : "De",
    "labelKey" : null,
    "name" : "from",
    "type" : "select"
  },
  {
    "enumValues" :
      [
        {
          "label" : "User SLV <user@streetlight-vision.com>",
          "properties" : null,
          "value" : "user@streetlight-vision.com"
        }
      ],
    "format" : null,
    "label" : "Vers",
    "labelKey" : null,
    "name" : "recipients",
    "type" : "select"
  },
  {
    "enumValues" : null,
    "format" : null,
    "label" : "Format HTML",
    "labelKey" : null,
    "name" : "htmlFormat",
    "type" : "boolean"
  }
]
```

6.12. GetVirtualScheduledReportTemplateEditablePropertyDescriptors

Return a virtual scheduled report template editable property descriptors.

6.12.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingManagementAPI?methodName=getVirtualScheduledReportTemplateEditablePropertyDescriptors

6.12.2. Supported Formats

XML, JSON

6.12.3. Supported request methods

GET

6.12.4. Parameters

groupId	Identifier of the geozone	Required
templateImplClassName	Class name of the scheduled report template	Required

6.12.5. Example requests

This request returns an array of SLVPropertyDescriptor objects.

6.12.5.1. Example with XML format :

```
<SLVPropertyDescriptor-array>
 <SLVPropertyDescriptor>
   <name>description</name>
   <label>description</label>
   <type>string</type>
 </SLVPropertyDescriptor>
 <SLVPropertyDescriptor>
   <name>periodStrId</name>
   <label>Périodicité</label>
   <type>select</type>
   <enumValues class="SLVLabelledValue-array">
     <SLVLabelledValue>
       <value class="string">every.day</value>
       <label>Every Day</label>
     </SLVLabelledValue>
     <SLVLabelledValue>
       <value class="string">every.monday</value>
       <label>Every Monday</label>
     </SLVLabelledValue>
```

```
      <SLVLabelledValue>
        <value class="string">every.tuesday</value>
        <label>Every Tuesday</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="string">every.wednesday</value>
        <label>Every Wednesday</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="string">every.thursday</value>
        <label>Every Thursday</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="string">every.friday</value>
        <label>Every Friday</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="string">every.saturday</value>
        <label>Every Saturday</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="string">every.sunday</value>
        <label>Every Sunday</label>
      </SLVLabelledValue>
    </enumValues>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>hourOfDay</name>
    <label>Heure</label>
    <type>select</type>
    <enumValues class="SLVLabelledValue-array">
      <SLVLabelledValue>
        <value class="int">0</value>
        <label>00</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">1</value>
        <label>01</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">2</value>
        <label>02</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">3</value>
        <label>03</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">4</value>
        <label>04</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">5</value>
        <label>05</label>
      </SLVLabelledValue>
      ...
      <SLVLabelledValue>
        <value class="int">23</value>
        <label>23</label>
      </SLVLabelledValue>
    </enumValues>
  </SLVPropertyDescriptor>
  <SLVPropertyDescriptor>
    <name>minute</name>
    <label>Minute</label>
    <type>select</type>
    <enumValues class="SLVLabelledValue-array">
      <SLVLabelledValue>
        <value class="int">0</value>
        <label>00</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">1</value>
        <label>01</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
        <value class="int">2</value>
        <label>02</label>
      </SLVLabelledValue>
      <SLVLabelledValue>
```

```
      <value class="int">3</value>
      <label>03</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="int">4</value>
      <label>04</label>
    </SLVLabelledValue>
    <SLVLabelledValue>
      <value class="int">5</value>
      <label>05</label>
    </SLVLabelledValue>
    ...
    <SLVLabelledValue>
      <value class="int">59</value>
      <label>59</label>
    </SLVLabelledValue>
  </enumValues>
</SLVPropertyDescriptor>
<SLVPropertyDescriptor>
  <name>from</name>
  <label>From</label>
  <type>select</type>
  <enumValues class="SLVLabelledValue-array">
    <SLVLabelledValue>
      <value class="string">user@streetlight-vision.com</value>
      <label>User SLV &lt;user@streetlight-vision.com&gt;</label>
    </SLVLabelledValue>
  </enumValues>
</SLVPropertyDescriptor>
<SLVPropertyDescriptor>
  <name>recipients</name>
  <label>To</label>
  <type>select</type>
  <enumValues class="SLVLabelledValue-array">
    <SLVLabelledValue>
      <value class="string">user@streetlight-vision.com</value>
      <label>User SLV &lt;user@streetlight-vision.com&gt;</label>
    </SLVLabelledValue>
  </enumValues>
</SLVPropertyDescriptor>
```

```
  <SLVPropertyDescriptor>
    <name>htmlFormat</name>
    <label>Format HTML</label>
    <type>boolean</type>
  </SLVPropertyDescriptor>
</SLVPropertyDescriptor-array>
```

Example with JSON format :

```
 [
   {
      "enumValues" : null,
      "format" : null,
      "label" : "description",
      "labelKey" : null,
      "name" : "description",
      "type" : "string"
   },
   {
      "enumValues" :
         [
            {
               "label" : "Chaque jour",
               "properties" : null,
               "value" : "every.day"
            },
            {
               "label" : "Chaque Lundi",
               "properties" : null,
               "value" : "every.monday"
            },
            {
               "label" : "Chaque Mardi",
               "properties" : null,
               "value" : "every.tuesday"
            },
            {
               "label" : "Chaque Mercredi",
               "properties" : null,
               "value" : "every.wednesday"
      },
      {
        "label" : "Chaque Jeudi",
        "properties" : null,
        "value" : "every.thursday"
      },
      {
        "label" : "Chaque Vendredi",
        "properties" : null,
        "value" : "every.friday"
      },
      {
        "label" : "Chaque Samedi",
        "properties" : null,
        "value" : "every.saturday"
      },
      {
        "label" : "Chaque Dimanche",
        "properties" : null,
        "value" : "every.sunday"
      }
    ],
  "format" : null,
  "label" : "Périodicité",
  "labelKey" : null,
  "name" : "periodStrId",
  "type" : "select"
},
{
  "enumValues" :
    [
      {
        "label" : "00",
        "properties" : null,
        "value" : 0
      },
      {
        "label" : "01",
        "properties" : null,
        "value" : 1
      },
      {
        "label" : "02",
        "properties" : null,
        "value" : 2
      },
      {
        "label" : "03",
        "properties" : null,
        "value" : 3
      },
      {
        "label" : "04",
        "properties" : null,
        "value" : 4
      },
      {
        "label" : "05",
        "properties" : null,
        "value" : 5
      },
      ...
      {
        "label" : "23",
        "properties" : null,
        "value" : 23
      }
    ],
  "format" : null,
  "label" : "Heure",
  "labelKey" : null,
  "name" : "hourOfDay",
  "type" : "select"
},
{
  "enumValues" :
    [
      {
        "label" : "00",
        "properties" : null,
          "value" : 0
        },
        {
          "label" : "01",
          "properties" : null,
          "value" : 1
        },
        {
          "label" : "02",
          "properties" : null,
          "value" : 2
        },
        {
          "label" : "03",
          "properties" : null,
          "value" : 3
        },
        {
          "label" : "04",
          "properties" : null,
          "value" : 4
        },
        {
          "label" : "05",
          "properties" : null,
          "value" : 5
        },
        ...
        {
          "label" : "59",
          "properties" : null,
          "value" : 59
        }
      ],
    "format" : null,
    "label" : "Minute",
    "labelKey" : null,
    "name" : "minute",
    "type" : "select"
  },
{
  "enumValues" : null,
  "format" : null,
  "label" : "Hôte FTP",
  "labelKey" : null,
  "name" : "ftpHost",
  "type" : "string"
},
{
  "enumValues" : null,
  "format" : null,
  "label" : "Utilisateur FTP",
  "labelKey" : null,
  "name" : "ftpUser",
  "type" : "string"
},
{
  "enumValues" : null,
  "format" : null,
  "label" : "Mot de passe FTP",
  "labelKey" : null,
  "name" : "ftpPassword",
  "type" : "string"
},
{
  "enumValues" : null,
  "format" : null,
  "label" : "Répertoire",
  "labelKey" : null,
  "name" : "ftpDirectory",
  "type" : "string"
},
{
  "enumValues" : null,
  "format" : null,
  "label" : "Temps d'écart minimum (minutes)",
  "labelKey" : null,
  "name" : "minDeltaTimeInMinutes",
  "type" : "int"
},
{
  "enumValues" : null,
  "format" : null,
  "label" : "Niveau d'écart minimum (%)",
  "labelKey" : null,
  "name" : "minDeltaLevel",
  "type" : "int"
},
{
  "enumValues" :
    [
      {
        "label" : "Ballast Communication",
        "properties" : null,
        "value" : "Default0"
      },
      {
        "label" : "Error",
        "properties" : null,
        "value" : "Default1"
      },
      {
        "label" : "Runhours Alarm",
        "properties" : null,
        "value" : "Default2"
      },
      ...
    ],
  "format" : null,
  "label" : "Défauts",
  "labelKey" : null,
  "name" : "downMeaningStrIds",
  "type" : "select"
},
{
  "enumValues" :
    [
      {
        "label" : "0",
        "properties" : null,
          "value" : 0
        },
        {
          "label" : "1",
          "properties" : null,
          "value" : 1
        },
        {
          "label" : "2",
          "properties" : null,
          "value" : 2
        },
        {
          "label" : "3",
          "properties" : null,
          "value" : 3
        },
        {
          "label" : "4",
          "properties" : null,
          "value" : 4
        },
        {
          "label" : "5",
          "properties" : null,
          "value" : 5
        },
        {
          "label" : "6",
          "properties" : null,
          "value" : 6
        },
        {
          "label" : "7",
          "properties" : null,
          "value" : 7
        }
      ],
    "format" : null,
    "label" : "Nb Tentatives",
      "labelKey" : null,
      "name" : "maxRetry",
      "type" : "select"
   }
]
```

6.13. UpdateScheduledReportTemplateDefinition

Update a scheduled report definition.

6.13.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVReportingManagementAPI?methodName=updateScheduledReportTemplateDefinition

6.13.2. Supported Formats

XML, JSON

6.13.3. Supported request methods

GET

6.13.4. Parameters

scheduledReportTemplateId	Identifier of the scheduled report definition	Required
propertyNames	Array of property names	Required
propertyValues	Array of property values	Required

6.13.5. Example requests

This request returns a SLVScheduledReportDefinition object.

6.13.5.1. Example with XML format :

```
 <SLVScheduledReportDefinition>
  <id>demoTemplate</id>
```

```
<scheduledReportTemplateImplClassName>com.dotv.streetlightserver.plugin.scheduledreports.impl.f
ailures.DailyFailuresReportTemplate</scheduledReportTemplateImplClassName>
  <geoZoneId>334</geoZoneId>
  <propertyValues>
    <entry>
      <string>minute</string>
      <int>0</int>
    </entry>
    <entry>
      <string>hourOfDay</string>
      <int>10</int>
    </entry>
    <entry>
      <string>ftpFileName</string>
      <string>failures</string>
    </entry>
  </propertyValues>
</SLVScheduledReportDefinition>
```

Example with JSON format :

```
 {
   "geoZoneId" : 391,
   "id" : "demoTemplate",
   "propertyValues" :
      {
        "minute" : 0,
        "downMeaningStrIds" :
          [
            "Default0",
            "Default13",
            "Default11",
            "Default5",
            "Default8",
            "DefaultLostNode",
            "Default10",
            "Default12",
            "Default3",
            "Default9"
          ],
        "unitName" : "LIGHTING COLUMN",
        "customerShortCode" : "SLV",
        "customerFullName" : "SLV",
        "serviceCode" : "SLVSymologyReport",
        "contactType" : "SLV",
        "ftpDirectory" : "",
        "webCustomerId" : "SLVWebPortal",
        "hourOfDay" : 13
    },
  "scheduledReportTemplateImplClassName" : "com.dotv.streetlightserver.plugin.scheduledreports.impl.symology.SymologyScheduledReportTemplate"
}
```

7. User

7.1. CreateProfil

Create a new profile.

7.1.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVUserProfilAPI?methodName=createProfil

7.1.2. Supported Formats

XML, JSON

7.1.3. Supported request methods

GET

7.1.4. Parameters

profilName	Name of profile	Required
geoZoneId	Identifier of root geozone	Required

7.1.5. Example requests

This request returns a SLVProfil object.

7.1.5.1. Example with XML format :

```
<SLVProfil>
  <name>newProfile</name>
  <properties>
    <SLVKeyValuePair>
      <key>img.topbanner</key>
      <value class="string">/resources/img/topbanner_left.gif</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>img.bottombanner</key>
      <value class="string">/resources/img/bottomBar.gif</value>
```

```
</SLVKeyValuePair>
<SLVKeyValuePair>
  <key>locale</key>
  <value class="string">en_US</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
  <key>desktop</key>
  <value class="string">desktopApplicationDefault.xml</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
  <key>hilitable</key>
  <value class="string">/resources/js/hilitableTable.js</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
  <key>providerRootId</key>
  <value class="string"></value>
</SLVKeyValuePair>
<SLVKeyValuePair>
  <key>img.toprightbanner</key>
  <value class="string">/resources/img/topbanner_right.gif</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
  <key>skin</key>
  <value class="string">streetlight</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
  <key>blockedActions</key>
  <value class="string">ShowDevicesApplicationAction</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
  <key>hide.bottombanner</key>
  <value class="string">false</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
  <key>maintenanceAccessFromDeviceCard</key>
  <value class="string">false</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
  <key>hide.topbanner</key>
  <value class="string">false</value>
```

```
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>img.topcenterbanner</key>
      <value class="string">/resources/img/topbanner_center.gif</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>customerRootId</key>
      <value class="string"></value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>css</key>
      <value class="string">/resources/css/default.css</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>geoZoneRootId</key>
      <value class="string">19</value>
    </SLVKeyValuePair>
  </properties>
</SLVProfil>
```

Example with JSON format :

```
{
  "name" : "newProfile",
  "properties" :
    [
      {
        "key" : "img.topbanner",
        "value" : "/resources/img/topbanner_left.gif"
      },
      {
        "key" : "img.bottombanner",
        "value" : "/resources/img/bottomBar.gif"
      },
      {
        "key" : "locale",
        "value" : "en_US"
      },
      {
        "key" : "desktop",
      "value" : "desktopApplicationDefault.xml"
    },
    {
      "key" : "hilitable",
      "value" : "/resources/js/hilitableTable.js"
    },
    {
      "key" : "providerRootId",
      "value" : ""
    },
    {
      "key" : "img.toprightbanner",
      "value" : "/resources/img/topbanner_right.gif"
    },
    {
      "key" : "skin",
      "value" : "streetlight"
    },
    {
      "key" : "blockedActions",
      "value" : ""
    },
    {
      "key" : "hide.bottombanner",
      "value" : "false"
    },
    {
      "key" : "maintenanceAccessFromDeviceCard",
      "value" : "false"
    },
    {
      "key" : "hide.topbanner",
      "value" : "false"
    },
    {
      "key" : "img.topcenterbanner",
      "value" : "/resources/img/topbanner_center.gif"
    },
    {
      "key" : "customerRootId",
        "value" : ""
      },
      {
        "key" : "css",
        "value" : "/resources/css/default.css"
      },
      {
        "key" : "geoZoneRootId",
        "value" : "19"
      }
    ]
}
```

7.2. CreateUser

Create a new user.

7.2.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVUserProfilAPI?methodName=createUser

7.2.2. Supported Formats

XML, JSON

7.2.3. Supported request methods

GET

7.2.4. Parameters

login	Login of user	Required
password	Password of user	Required
profilName	Name of a profile	Required
firstName	First name of user	Optional
lastName	Last name of user	Optional
contactAdresse	Address of user	Optional

contactPhone	Phone of user	Optional
contactPager	Pager of user	Optional
contactMail	Mail of user	Optional

7.2.5. Example requests

This request returns a SLVUser object.

7.2.5.1. Example with XML format :

```
 <SLVUser>
   <admin>true</admin>
   <login>slvUser</login>
   <password>slvPassword</password>
   <profilName>slvProfile</profilName>
   <contact>
      <id>1</id>
      <address>26 bvd MALESHERBES</address>
      <phone>+33 1 47 70 25 25</phone>
      <pager>+33 1 47 70 25 25</pager>
      <mail>support@streetlight-vision.com</mail>
   </contact>
</SLVUser>
```

Example with JSON format :

```
 {
   "admin" : true,
   "contact" :
     {
       "address" : "26 bvd MALESHERBES",
       "id" : 2,
       "mail" : "support@streetlight-vision.com",
       "name" : null,
       "pager" : "+33 1 47 70 25 25",
       "phone" : "+33 1 47 70 25 25"
     },
   "firstName" : null,
   "lastName" : null,
   "login" : "slvUser",
   "password" : "slvPassword",
   "profilName" : "slvProfile"
}
```

7.3. DeleteProfil

Delete a profile.

7.3.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVUserProfilAPI?methodName=deleteProfil

7.3.2. Supported Formats

XML, JSON

7.3.3. Supported request methods

GET

7.3.4. Parameters

profilName	Name of profile	Required

7.3.5. Example requests

This request returns a SLVResult object.

7.3.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
    <status>OK</status>
    <errorCode>0</errorCode>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
  "errorCode" : "0",
  "status" : "OK",
  "value" : null
}
```

7.4. DeleteUser

Delete an user.

7.4.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVUserProfilAPI?methodName=deleteUser

7.4.2. Supported Formats

XML, JSON

7.4.3. Supported request methods

GET

7.4.4. Parameters

login	Login of user	Required

7.4.5. Example requests

This request returns a SLVResult object.

7.4.5.1. Example with XML format :

```
 <com.dotv.streetlightserver.api.data.SLVResult>
   <status>OK</status>
   <errorCode>0</errorCode>
</com.dotv.streetlightserver.api.data.SLVResult>
```

Example with JSON format :

```
 {
  "errorCode" : "0",
  "status" : "OK",
  "value" : null
}
```

7.5. GetAllProfils

Return an array of all profiles.

7.5.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getAllProfils

7.5.2. Supported Formats

XML, JSON

7.5.3. Supported request methods

GET

7.5.4. Example requests

This request returns an array of SLVProfil objects.

7.5.4.1. Example with XML format :

```
<SLVProfil-array>
  <SLVProfil>
    <name>default</name>
    <properties>
      <SLVKeyValuePair>
        <key>img.topbanner</key>
        <value class="string">/resources/img/topbanner_left.gif</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>img.bottombanner</key>
        <value class="string">/resources/img/bottomBar.gif</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>locale</key>
        <value class="string">en_US</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>desktop</key>
        <value class="string">desktopApplicationDefault.xml</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>hilitable</key>
        <value class="string">/resources/js/hilitableTable.js</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>providerRootId</key>
        <value class="string"></value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>img.toprightbanner</key>
        <value class="string">/resources/img/topbanner_right.gif</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>skin</key>
        <value class="string">streetlight</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>blockedActions</key>
        <value class="string">ShowDevicesApplicationAction</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>hide.bottombanner</key>
        <value class="string">false</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>maintenanceAccessFromDeviceCard</key>
        <value class="string">false</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>hide.topbanner</key>
        <value class="string">false</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
```

```
        <key>img.topcenterbanner</key>
        <value class="string">/resources/img/topbanner_center.gif</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>customerRootId</key>
        <value class="string"></value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>css</key>
        <value class="string">/resources/css/default.css</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>geoZoneRootId</key>
        <value class="string"></value>
      </SLVKeyValuePair>
    </properties>
  </SLVProfil>
  ...
</SLVProfil-array>
```

Example with JSON format :

```
 [
  {
    "name" : "default",
    "properties" :
      [
        {
          "key" : "img.topbanner",
          "value" : "/resources/img/topbanner_left.gif"
        },
        {
          "key" : "img.bottombanner",
          "value" : "/resources/img/bottomBar.gif"
        },
        {
          "key" : "locale",
          "value" : "en_US"
        },
        {
```

```
      "key" : "desktop",
      "value" : "desktopApplicationDefault.xml"
    },
    {
      "key" : "hilitable",
      "value" : "/resources/js/hilitableTable.js"
    },
    {
      "key" : "providerRootId",
      "value" : ""
    },
    {
      "key" : "img.toprightbanner",
      "value" : "/resources/img/topbanner_right.gif"
    },
    {
      "key" : "skin",
      "value" : "streetlight"
    },
    {
      "key" : "blockedActions",
      "value" : "ShowDevicesApplicationAction"
    },
    {
      "key" : "hide.bottombanner",
      "value" : "false"
    },
    {
      "key" : "maintenanceAccessFromDeviceCard",
      "value" : "false"
    },
    {
      "key" : "hide.topbanner",
      "value" : "false"
    },
    {
      "key" : "img.topcenterbanner",
      "value" : "/resources/img/topbanner_center.gif"
    },
    {
        "key" : "customerRootId",
        "value" : ""
      },
      {
        "key" : "css",
        "value" : "/resources/css/default.css"
      },
      {
        "key" : "geoZoneRootId",
        "value" : ""
      }
    ]
  },
  ...
]
```

7.6. GetAllUsers

Return an array of all users.

7.6.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getAllUsers

7.6.2. Supported Formats

XML, JSON

7.6.3. Supported request methods

GET

7.6.4. Example requests

This request returns an array of SLVUser objects.

7.6.4.1. Example with XML format :

```
 <SLVUser-array>
```

```
    <SLVUser>
        <admin>true</admin>
        <login>user</login>
        <password>password</password>
        <profilName>default</profilName>
    </SLVUser>
    ...
</SLVUser-array>
```

Example with JSON format :

```
 [
  {
    "admin" : true,
    "contact" : null,
    "firstName" : null,
    "lastName" : null,
    "login" : "user",
    "password" : "password",
    "profilName" : "default"
  },
  ...
]
```

7.7. GetGeoZoneProfils

Return an array of profiles for the given geozone.

7.7.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVUserProfilAPI?methodName=getGeozoneProfils

7.7.2. Supported Formats

XML, JSON

7.7.3. Supported request methods

GET

7.7.4. Parameters

geoZoneId	Identifier of parent geozone	Required
recurse	True to do a recursive get. False by default.	Optional

7.7.5. Example requests

This request returns an array of SLVProfil objects.

7.7.5.1. Example with XML format :

```
<SLVProfil-array>
  <SLVProfil>
    <name>default</name>
    <properties>
      <SLVKeyValuePair>
        <key>img.topbanner</key>
        <value class="string">/resources/img/topbanner_left.gif</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>img.bottombanner</key>
        <value class="string">/resources/img/bottomBar.gif</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>locale</key>
        <value class="string">en_US</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>desktop</key>
        <value class="string">desktopApplicationDefault.xml</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>hilitable</key>
        <value class="string">/resources/js/hilitableTable.js</value>
      </SLVKeyValuePair>
      <SLVKeyValuePair>
        <key>providerRootId</key>
        <value class="string"></value>
      </SLVKeyValuePair>
```

```
<SLVKeyValuePair>
    <key>img.toprightbanner</key>
    <value class="string">/resources/img/topbanner_right.gif</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
    <key>skin</key>
    <value class="string">streetlight</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
    <key>blockedActions</key>
    <value class="string">ShowDevicesApplicationAction</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
    <key>hide.bottombanner</key>
    <value class="string">false</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
    <key>maintenanceAccessFromDeviceCard</key>
    <value class="string">false</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
    <key>hide.topbanner</key>
    <value class="string">false</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
    <key>img.topcenterbanner</key>
    <value class="string">/resources/img/topbanner_center.gif</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
    <key>customerRootId</key>
    <value class="string"></value>
</SLVKeyValuePair>
<SLVKeyValuePair>
    <key>css</key>
    <value class="string">/resources/css/default.css</value>
</SLVKeyValuePair>
<SLVKeyValuePair>
    <key>geoZoneRootId</key>
    <value class="string"></value>
</SLVKeyValuePair>
```

```
    </properties>
  </SLVProfil>
  ...
</SLVProfil-array>
```

Example with JSON format :

```
 [
  {
    "name" : "default",
    "properties" :
      [
        {
          "key" : "img.topbanner",
          "value" : "/resources/img/topbanner_left.gif"
        },
        {
          "key" : "img.bottombanner",
          "value" : "/resources/img/bottomBar.gif"
        },
        {
          "key" : "locale",
          "value" : "en_US"
        },
        {
          "key" : "desktop",
          "value" : "desktopApplicationDefault.xml"
        },
        {
          "key" : "hilitable",
          "value" : "/resources/js/hilitableTable.js"
        },
        {
          "key" : "providerRootId",
          "value" : ""
        },
        {
          "key" : "img.toprightbanner",
          "value" : "/resources/img/topbanner_right.gif"
        },
        {
          "key" : "skin",
          "value" : "streetlight"
        },
        {
          "key" : "blockedActions",
          "value" : "ShowDevicesApplicationAction"
        },
        {
          "key" : "hide.bottombanner",
          "value" : "false"
        },
        {
          "key" : "maintenanceAccessFromDeviceCard",
          "value" : "false"
        },
        {
          "key" : "hide.topbanner",
          "value" : "false"
        },
        {
          "key" : "img.topcenterbanner",
          "value" : "/resources/img/topbanner_center.gif"
        },
        {
          "key" : "customerRootId",
          "value" : ""
        },
        {
          "key" : "css",
          "value" : "/resources/css/default.css"
        },
        {
          "key" : "geoZoneRootId",
          "value" : ""
        }
      ]
  },
  ...
]
```

7.8. GetGeoZoneUsers

Return an array of users for the given geozone.

7.8.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVUserProfilAPI?methodName=getGeoZoneUsers

7.8.2. Supported Formats

XML, JSON

7.8.3. Supported request methods

GET

7.8.4. Parameters

geoZoneId	Identifier of parent geozone	Required
recurse	True to do a recursive get. False by default.	Optional

7.8.5. Example requests

This request returns an array of SLVUser objects.

7.8.5.1. Example with XML format :

```
 <SLVUser-array>
   <SLVUser>
       <admin>true</admin>
       <login>user</login>
       <password>password</password>
       <profilName>default</profilName>
   </SLVUser>
   ...
</SLVUser-array>
```

Example with JSON format :

```
 [
  {
    "admin" : true,
    "contact" : null,
    "firstName" : null,
    "lastName" : null,
    "login" : "user",
    "password" : "password",
    "profilName" : "default"
  },
  ...
]
```

7.9. GetProfil

Return the profile for the given profile name.

7.9.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVUserProfilAPI?methodName=getProfil

7.9.2. Supported Formats

XML, JSON

7.9.3. Supported request methods

GET

7.9.4. Parameters

profilName	Name of profile	Required

7.9.5. Example requests

This request returns a SLVProfil object.

7.9.5.1. Example with XML format :

```
<SLVProfil>
  <name>default</name>
  <properties>
    <SLVKeyValuePair>
      <key>img.topbanner</key>
      <value class="string">/resources/img/topbanner_left.gif</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>img.bottombanner</key>
      <value class="string">/resources/img/bottomBar.gif</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>locale</key>
      <value class="string">en_US</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>desktop</key>
      <value class="string">desktopApplicationDefault.xml</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>hilitable</key>
      <value class="string">/resources/js/hilitableTable.js</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>providerRootId</key>
      <value class="string"></value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>img.toprightbanner</key>
      <value class="string">/resources/img/topbanner_right.gif</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>skin</key>
      <value class="string">streetlight</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>blockedActions</key>
      <value class="string">ShowDevicesApplicationAction</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
```

```
      <key>hide.bottombanner</key>
      <value class="string">false</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>maintenanceAccessFromDeviceCard</key>
      <value class="string">false</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>hide.topbanner</key>
      <value class="string">false</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>img.topcenterbanner</key>
      <value class="string">/resources/img/topbanner_center.gif</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>customerRootId</key>
      <value class="string"></value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>css</key>
      <value class="string">/resources/css/default.css</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
      <key>geoZoneRootId</key>
      <value class="string"></value>
    </SLVKeyValuePair>
  </properties>
</SLVProfil>
```

Example with JSON format :

```
{
  "name" : "default",
  "properties" :
    [
      {
        "key" : "img.topbanner",
        "value" : "/resources/img/topbanner_left.gif"
      },
{
  "key" : "img.bottombanner",
  "value" : "/resources/img/bottomBar.gif"
},
{
  "key" : "locale",
  "value" : "en_US"
},
{
  "key" : "desktop",
  "value" : "desktopApplicationDefault.xml"
},
{
  "key" : "hilitable",
  "value" : "/resources/js/hilitableTable.js"
},
{
  "key" : "providerRootId",
  "value" : ""
},
{
  "key" : "img.toprightbanner",
  "value" : "/resources/img/topbanner_right.gif"
},
{
  "key" : "skin",
  "value" : "streetlight"
},
{
  "key" : "blockedActions",
  "value" : "ShowDevicesApplicationAction"
},
{
  "key" : "hide.bottombanner",
  "value" : "false"
},
{
  "key" : "maintenanceAccessFromDeviceCard",
  "value" : "false"
},
        {
          "key" : "hide.topbanner",
          "value" : "false"
        },
        {
          "key" : "img.topcenterbanner",
          "value" : "/resources/img/topbanner_center.gif"
        },
        {
          "key" : "customerRootId",
          "value" : ""
        },
        {
          "key" : "css",
          "value" : "/resources/css/default.css"
        },
        {
          "key" : "geoZoneRootId",
          "value" : ""
        }
      ]
}
```

7.10. GetProfilProperties

Return an array of profile properties for the current logged user.

7.10.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVAssetAPI?methodName=getProfilProperties

7.10.2. Supported Formats

XML, JSON

7.10.3. Supported request methods

GET

7.10.4. Example requests

This request returns an array of SLVKeyValuePair objects.

7.10.4.1. Example with XML format :

```
<SLVKeyValuePair-array>
  <SLVKeyValuePair>
    <key>img.topbanner</key>
    <value class="string">/resources/img/topbanner_left.gif</value>
  </SLVKeyValuePair>
  <SLVKeyValuePair>
    <key>img.bottombanner</key>
    <value class="string">/resources/img/bottomBar.gif</value>
  </SLVKeyValuePair>
  <SLVKeyValuePair>
    <key>locale</key>
    <value class="string">en_US</value>
  </SLVKeyValuePair>
  <SLVKeyValuePair>
    <key>desktop</key>
    <value class="string">desktopApplicationAdmin.xml</value>
  </SLVKeyValuePair>
  <SLVKeyValuePair>
    <key>hilitable</key>
    <value class="string">/resources/js/hilitableTable.js</value>
  </SLVKeyValuePair>
  <SLVKeyValuePair>
    <key>providerRootId</key>
    <value class="string"></value>
  </SLVKeyValuePair>
  <SLVKeyValuePair>
    <key>img.toprightbanner</key>
    <value class="string">/resources/img/topbanner_right.gif</value>
  </SLVKeyValuePair>
  <SLVKeyValuePair>
    <key>skin</key>
    <value class="string">streetlight</value>
  </SLVKeyValuePair>
```

```
    <SLVKeyValuePair>
        <key>blockedActions</key>
        <value class="string"></value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
        <key>hide.bottombanner</key>
        <value class="string">false</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
        <key>maintenanceAccessFromDeviceCard</key>
        <value class="string">false</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
        <key>hide.topbanner</key>
        <value class="string">false</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
        <key>img.topcenterbanner</key>
        <value class="string">/resources/img/topbanner_center.gif</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
        <key>customerRootId</key>
        <value class="string"></value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
        <key>css</key>
        <value class="string">/resources/css/default.css</value>
    </SLVKeyValuePair>
    <SLVKeyValuePair>
        <key>geoZoneRootId</key>
        <value class="string"></value>
    </SLVKeyValuePair>
</SLVKeyValuePair-array>
```

Example with JSON format :

```
 [
   {
      "key" : "img.topbanner",
      "value" : "/resources/img/topbanner_left.gif"
```

```
},
{
  "key" : "img.bottombanner",
  "value" : "/resources/img/bottomBar.gif"
},
{
  "key" : "locale",
  "value" : "en_US"
},
{
  "key" : "desktop",
  "value" : "desktopApplicationAdmin.xml"
},
{
  "key" : "hilitable",
  "value" : "/resources/js/hilitableTable.js"
},
{
  "key" : "providerRootId",
  "value" : ""
},
{
  "key" : "img.toprightbanner",
  "value" : "/resources/img/topbanner_right.gif"
},
{
  "key" : "skin",
  "value" : "streetlight"
},
{
  "key" : "blockedActions",
  "value" : ""
},
{
  "key" : "hide.bottombanner",
  "value" : "false"
},
{
  "key" : "maintenanceAccessFromDeviceCard",
  "value" : "false"
  },
  {
    "key" : "hide.topbanner",
    "value" : "false"
  },
  {
    "key" : "img.topcenterbanner",
    "value" : "/resources/img/topbanner_center.gif"
  },
  {
    "key" : "customerRootId",
    "value" : ""
  },
  {
    "key" : "css",
    "value" : "/resources/css/default.css"
  },
  {
    "key" : "geoZoneRootId",
    "value" : ""
  }
]
```

7.11. GetUser

Return the user for the given user login.

7.11.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVUserProfilAPI?methodName=getUser

7.11.2. Supported Formats

XML, JSON

7.11.3. Supported request methods

GET

7.11.4. Parameters

login	Login of user	Required

7.11.5. Example requests

This request returns a SLVUser object.

7.11.5.1. Example with XML format :

```
 <SLVUser>
    <admin>true</admin>
    <login>user</login>
    <password>password</password>
    <profilName>default</profilName>
</SLVUser>
```

Example with JSON format :

```
 {
   "admin" : true,
   "contact" : null,
   "firstName" : null,
   "lastName" : null,
   "login" : "user",
   "password" : "password",
   "profilName" : "default"
}
```

7.12. UpdateProfilProperties

Update the profile properties.

7.12.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVUserProfilAPI?methodName=updateProfilProperties

7.12.2. Supported Formats

XML, JSON

7.12.3. Supported request methods

GET

7.12.4. Parameters

profilName	Name of profile	Required
property.key	Array of properties where key is a profile property name.	Required

7.12.5. Example requests

This request returns OK status.

7.12.5.1. Example with XML format :

```
<string>OK</string>
```

Example with JSON format :

```
"OK"
```

7.13. UpdateUser

Update the user properties.

7.13.1. URL

http://{serverHost}:{serverPort}/reports/api/servlet/SLVUserProfilAPI?methodName=updateUser

7.13.2. Supported Formats

XML, JSON

7.13.3. Supported request methods

GET

7.13.4. Parameters

login	Login of user	Required
password	Password of user	Optional
profilName	Name of a profile	Optional
firstName	First name of user	Optional
lastName	Last name of user	Optional
contactAdresse	Address of user	Optional
contactPhone	Phone of user	Optional
contactPager	Pager of user	Optional
contactMail	Mail of user	Optional

7.13.5. Example requests

This request returns OK status.

7.13.5.1. Example with XML format :

```
<string>OK</string>
```

Example with JSON format :

```
"OK"
```