CA Unified Infrastructure Management RESTful
Web Services

Manual and Reference Guide (v2.17)

April 10, 2015
Version 2.17

© 2014 CA Inc. – All rights reserved
Contents
Introduction ........................................................................................................................................ 7
Changelog ......................................................................................................................................... 7
Known Issues and Planned Enhancements ..................................................................................... 9
Prerequisites .................................................................................................................................... 9
UI M Server ....................................................................................................................................... 9
Unified Monitoring Portal .............................................................................................................. 9
Installing the Application ................................................................................................................ 10
Deployment of the Package ............................................................................................................. 10
Testing a Successful Deployment ................................................................................................... 10
Call Reference ................................................................................................................................ 11
WADL Description .......................................................................................................................... 11
General Information on Paging ........................................................................................................ 11
Probe Related Calls ........................................................................................................................ 12
Get Probe Information .................................................................................................................. 12
Invoke Callback ............................................................................................................................. 13
Invoke Callback2 ............................................................................................................................ 15
Activate / Deactivate a Probe ......................................................................................................... 18
Get Probe Configuration ............................................................................................................... 18
Get Single Probe Configuration Value .......................................................................................... 19
Set single probe Configuration Value ........................................................................................... 19
Delete Single Probe Configuration Value ..................................................................................... 20
Set Multiple Probe Configuration Values ..................................................................................... 21
Alarm Related Calls ...................................................................................................................... 23
Create Alarm ................................................................................................................................. 23
Get Alarm Summary ...................................................................................................................... 24
Get Unfiltered List of Alarms ......................................................................................................... 25
Get Filtered List of Alarms .............................................................................................................. 27
Get State for all Alarms .................................................................................................................. 30
Get State for Filtered Alarms ......................................................................................................... 30
Accept an Alarm ............................................................................................................................ 32
Acknowledge an Alarm ................................................................................................................. 32
Assign an Alarm ............................................................................................................................. 32
Unassign an Alarm ........................................................................................................................ 33
Account Related Calls ................................................................................................................... 34
Get All Accounts ........................................................................................................................... 34
<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get an Account</td>
<td>35</td>
</tr>
<tr>
<td>Create a New Account</td>
<td>36</td>
</tr>
<tr>
<td>Update an Existing Account</td>
<td>38</td>
</tr>
<tr>
<td>Delete an Account</td>
<td>39</td>
</tr>
<tr>
<td>Get Account Contacts</td>
<td>39</td>
</tr>
<tr>
<td>Get Account Configuration Items</td>
<td>41</td>
</tr>
<tr>
<td>Add Account Configuration Items</td>
<td>41</td>
</tr>
<tr>
<td>Remove Account Configuration Items</td>
<td>42</td>
</tr>
<tr>
<td>Get Account Computer Systems</td>
<td>43</td>
</tr>
<tr>
<td>Add Account Computer Systems</td>
<td>43</td>
</tr>
<tr>
<td>Remove Account Computer Systems</td>
<td>44</td>
</tr>
<tr>
<td>Add Origin to an Account</td>
<td>45</td>
</tr>
<tr>
<td>Remove Origin from Account</td>
<td>46</td>
</tr>
<tr>
<td>Contact related calls</td>
<td>47</td>
</tr>
<tr>
<td>Get a Contact</td>
<td>47</td>
</tr>
<tr>
<td>Create a New Contact</td>
<td>48</td>
</tr>
<tr>
<td>Update an Existing Contact</td>
<td>49</td>
</tr>
<tr>
<td>Update an Existing Contacts Password</td>
<td>50</td>
</tr>
<tr>
<td>Delete an Existing Contact</td>
<td>51</td>
</tr>
<tr>
<td>UIM Infrastructure Related Calls</td>
<td>51</td>
</tr>
<tr>
<td>Get List of Hubs</td>
<td>51</td>
</tr>
<tr>
<td>Get List of Robots</td>
<td>53</td>
</tr>
<tr>
<td>Get Robot Details</td>
<td>54</td>
</tr>
<tr>
<td>Get List of Archive Packages</td>
<td>55</td>
</tr>
<tr>
<td>Dashboard Related Calls</td>
<td>57</td>
</tr>
<tr>
<td>Get Dashboards</td>
<td>57</td>
</tr>
<tr>
<td>Get Dashboard State</td>
<td>57</td>
</tr>
<tr>
<td>Configuration Item (CI) Data Retrieval Calls</td>
<td>59</td>
</tr>
<tr>
<td>Get CI Definitions</td>
<td>59</td>
</tr>
<tr>
<td>Get CI Definition</td>
<td>61</td>
</tr>
<tr>
<td>Get CI Metric Definitions</td>
<td>63</td>
</tr>
<tr>
<td>Get CI Metric Definition</td>
<td>64</td>
</tr>
<tr>
<td>Get CI Metric</td>
<td>65</td>
</tr>
<tr>
<td>QoS Data Retrieval Calls</td>
<td>67</td>
</tr>
<tr>
<td>Get QoS Definitions</td>
<td>67</td>
</tr>
<tr>
<td>Get a QoS Definition by Name</td>
<td>68</td>
</tr>
</tbody>
</table>
Get Sources for QoS-Name ........................................................................................................................................ 70
Get Targets for QoS-Name and Source ......................................................................................................................... 70
Get Targets for QoS-Name and Source for a Given Origin .......................................................................................... 71
Get Raw QoS Data ......................................................................................................................................................... 72
Get QoS Data Using MetricId .......................................................................................................................................... 74
Get QoS Data Using TableId .......................................................................................................................................... 75
Get Data Using QoS ConstraintId .................................................................................................................................. 77
Get Historical (aggregated) QoS Data ............................................................................................................................ 78
Get Historical QoS Data Using MetricId ......................................................................................................................... 80
Get Historical QoS Data Using TableId .......................................................................................................................... 81
Get Historical Data Using a QoS ConstraintId ................................................................................................................ 83
SLA Related Calls ......................................................................................................................................................... 85
Get All SLADefinitions ................................................................................................................................................... 85
Get SLA Definition.......................................................................................................................................................... 86
Get Configured Calculations .......................................................................................................................................... 87
Get Calculation Jobs for a SLA ......................................................................................................................................... 87
Get Compliance for All SLAs ......................................................................................................................................... 89
Get SLA Compliance....................................................................................................................................................... 90
Get SLO Definitions for a SLA........................................................................................................................................ 92
Get Compliance of SLOs of a SLA ................................................................................................................................. 93
SLO Related Calls .......................................................................................................................................................... 96
Get SLO Definition ........................................................................................................................................................ 96
Get QOS Constraints for a SLO ..................................................................................................................................... 96
Get QOS Constraint ....................................................................................................................................................... 99
Get Compliance Values for All QoS Constraints of a SLO ............................................................................................... 101
Get Compliance for a QoS Constraint by IDs ................................................................................................................ 102
Get QoS Constraint IDs for an SLO ................................................................................................................................ 103
ACL Related Calls ......................................................................................................................................................... 105
The ACL Data Structure ................................................................................................................................................ 105
Get a List of all ACLs ...................................................................................................................................................... 105
Get ACL of Current User ............................................................................................................................................... 106
Get a ACL by Name ....................................................................................................................................................... 107
Create a New ACL ......................................................................................................................................................... 108
Create a New ACL by Copying an Existing ACL ........................................................................................................... 109
Modify an Existing ACL ............................................................................................................................................... 109
Delete an ACL ............................................................................................................................................................. 110
### Introduction

CA Unified Infrastructure Management (UIM) has responded to rising customer demand for a RESTful (REST = Representational State Transfer) web service interface for CA UIM. This interface offers customers the functionality to access their UIM installation using REST-based web service calls.

### Changelog

This section outlines important changes introduced with each new release of the interface.

<table>
<thead>
<tr>
<th>Version</th>
<th>Changes</th>
</tr>
</thead>
</table>
| V2.17   | - Added note of clarification for origin behavior when updating an existing account.  
- Added information about origin behavior when creating an account immediately after UIM installation.  
- Added requirement for origin when creating an account.  
- Corrected the JSON and xml sample request for the createAlarm call. Removed `userTag1` and `userTag2` fields. These fields cannot be created using the `createAlarms` call.  
- Added support to get, create, update, and delete Liferay users in addition to Account Contact users when `webservices_rest` is installed with UMP version 8.2 and later.  
- Fixed update account function of REST API does not add origins.  
- Fixed REST API loses precision on large QoS data sample values. |
| V2.16   | - Fixed null pointer error in get alarms call for MySQL and MS SQL Server databases. |
| V 2.15  | - Fixed query to get filtered list of alarms.  
- Fixed query to add alarm attributes that were missing in v2.14. `timeReceived` has been replaced by `timeLast`.  
- Adding an origin displays in CA Unified Management Portal (UMP). |
| V 2.14  | - Fixed an issue in which the RESTful API would return different list of alarms from the probe utility and nas GUI  
- Fixed an issue in which REST calls would not return invisible alarms.  
- Fixed an issue in which using the probe-info call could result in a server error if the robot was running a probe that did not have a package name. This was typically seen on hub probes when they were initially installed, but not upgraded. This fix requires the wasp 8.0 probe shipped with UMP 8.0.  
- Corrected the Alarm Filter section of this guide. The `Id` field is now `id`. |
| V 2.13  | - Added calls for maintenance mode.  
- Added Get QoS Constraint IDs for an SLO.  
- Added calls for custom properties. |
| V 2.12  | - Corrected return codes for ACLs and Accounts (on Deletes and Updates).  
- Add and Remove Origins on accounts added.  
- Defect fixes. |
<table>
<thead>
<tr>
<th>Version</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>V 2.11</td>
<td>- Add support for posted content in JSON format.</td>
</tr>
<tr>
<td>V 2.10</td>
<td>- Added calls for CI (ConfigurationItem) data retrieval.</td>
</tr>
<tr>
<td>V 2.01</td>
<td>- Added call2 (invokeMethod2).</td>
</tr>
<tr>
<td>V 2.0</td>
<td>- Merged SOAP and RESTful service into one functional base.</td>
</tr>
<tr>
<td>V 1.7</td>
<td>- Corrected a problem where ACL calls would fail if the wasp is not running on a hub machine.</td>
</tr>
<tr>
<td>V 1.6</td>
<td>- Corrected a problem using the forward-slash in URLs (&quot;/&quot;).</td>
</tr>
<tr>
<td>V 1.5</td>
<td>- Corrected a problem in the usage of the database pools.</td>
</tr>
<tr>
<td>V 1.4</td>
<td>- Added calls to create, modify and delete ACLs</td>
</tr>
<tr>
<td>V 1.3</td>
<td>- The dates for started/restarted date in probe information requests are now correctly calculated.</td>
</tr>
<tr>
<td></td>
<td>- The contact password reset now works as documented i.e. that the contact id is not required in the supplied data structure but only in the url.</td>
</tr>
<tr>
<td></td>
<td>- The alarmfilter supplied to alarm calls can now also be filtered by alarm id.</td>
</tr>
<tr>
<td></td>
<td>- A call with the DELETE-method is now available to remove probe configuration keys.</td>
</tr>
<tr>
<td></td>
<td>- Calls to fetch all origins and the mapping between origins and accounts have been introduced.</td>
</tr>
<tr>
<td></td>
<td>- Calls to retrieve information about SLAs, their definitions, past compliances, current calculation jobs, their SLOs and SLO-compliances have been introduced. They are available under the resource /sla.</td>
</tr>
<tr>
<td></td>
<td>- Calls to retrieve information about SLOs, their definitions, compliance, the underlying QOS constraints and the constraint compliances have been introduced. They are available under the resource /slo.</td>
</tr>
<tr>
<td></td>
<td>- The call to retrieve hubs and to retrieve robots have been modified to support the optional GET query parameters maxrows and offset to support paging.</td>
</tr>
<tr>
<td>V 1.2</td>
<td>- The Account and Contact transfer structures were modified. The Models AccountDocument, AccountDocumentList, ContactDocument and ContactDocumentList were discontinued and replaced by Account, AccountList, Contact and ContactList.</td>
</tr>
<tr>
<td></td>
<td>- A problem where an account could be created with the same name as an existing Account was fixed.</td>
</tr>
<tr>
<td></td>
<td>- A problem where the list of QoS Sources contained all available sources not limited to the supplied QoS Name was fixed.</td>
</tr>
</tbody>
</table>
Known Issues and Planned Enhancements

- Versions of UMP prior to 8.2 do not display origins that the UIM system is unaware of. If webservices_rest 2.17 is installed with an UMP version prior to 8.2, an account can be created with an origin that does not exist elsewhere in UIM. UMP will display this account as if it does not have any origins. To view the origins that UMP is unaware of, use the REST API call for Get All Accounts. The REST API call Remove Origin from Account can be used to remove account-created origins if they will not be defined in UIM.

- The call to get alarms has been fixed to return the correct set of alarms. The new implementation can slow the performance down to some extent.

- When invoking a callback, communication errors can occur. This is because the REST call is not performing a retry if a communication error occurs in the background, for example, when a tunnel connection is unstable. This will be modified in the future.

- In rare situations, consuming the alarm list from a Java client can result in DeMarshallingExceptions due to non-UTF characters being present in the datastream. Other programming languages are not affected.

- Due to a defect in a subroutine of the QoS-Constraint call in the SLO resource, an account contact user could see qos values from other origins than the ones associated with its own account.

- Paging functionality is planned for all list structures being returned by the REST API.

Prerequisites

**UIM Server**

- Nimsoft Server v5.1 or later
- Infrastructure Manager v3.84.2 or later is required, and must be downloaded and installed before installing.

**Unified Monitoring Portal**

- Unified Monitoring Portal v2.5.1 GA or later.
Installing the Application

Deployment of the Package
To deploy the package, drag and drop the `webservices_rest` package from your local archive to a robot running the WASP probe.

The default configuration of the Tomcat server that serves as the platform for UMP does not allow encoded forward-slash characters ("/") in URLs. To avoid problems (for example, when requesting QoS metrics that contain a forward slash in the name), add the following key to the java startup arguments of the wasp probe:

```
-D org.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_SLASH=true
```

Once the deployment process has completed, restart the wasp probe where you deployed the package.

Testing a Successful Deployment
Browse to the following address:

http://<your_UMP_address>/rest/version-info

If the web services are running correctly, you should see version information for the web services:

```
Nimsoft RESTful web services interface version x.x
```

Note: Some browsers (such as Google Chrome) do not directly display this text, but instead show an error message such as Empty Document. To see the text, right-click the page and select Show Page Source.
Call Reference

Note: All URLs mentioned are relative to the REST web services base URL which is:

http(s)://<your_UMP_address>:<your_UMP_port>/rest

In the sample HTTP-Requests and HTTP-Responses, certain header information (such as authentication headers) is omitted. This means that the listed requests will not work if sent directly to the interface. The purpose of those samples is to illustrate the data structure that used when invoking the calls.

WADL Description

The RESTful web services interface offers an automatically generated web application description file that specifies all exposed service calls and structures.

This file is available on your UMP at:

http://<your_UMP_address>/rest/application.wadl

General Information on Paging

Many calls to retrieve list information support paging. On those that do, it is indicated in the URL of the call within the optional parameter setting: ?maxrows=X&offset=Y

The parameter maxrows defines how many rows the result should contain in maximum (maxrows=5 returns 5 entries maximum). If the list contains a lesser number of rows, all rows are returned.

The second parameter offset can only be used if maxrows is also set. Offset defines from where to fetch the number of entries defined in maxrows. For example, if the list contains 20 entries in total, then

?maxrows=5&offset=10

returns entries numbered 10 through 15 in the list.
# Probe Related Calls

## Get Probe Information

This call returns information about a running probe.

<table>
<thead>
<tr>
<th>URL</th>
<th>/probe/ {domain}/(hub)/{robot}/(probe)/info</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; XML/JSON ProbelInfo Structure</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Basic Management</td>
</tr>
</tbody>
</table>

**Sample Request**

```plaintext
get /rest/probe/chris-dev/primaryhub/nb-1538/cdm/info HTTP/1.1

Accept: application/xml
```

**Sample Reply (XML)**

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<probeinfo>
  <company>Nimsoft Corporation</company>
  <connections>1</connections>
  <libDate>Sep 17 2010</libDate>
  <libVersion>5.12 (64bit)</libVersion>
  <messages>1</messages>
  <name>cdm</name>
  <restarted>1970-01-01T01:00:00+01:00</restarted>
  <started>1969-12-21T19:54:36.832+01:00</started>
  <version>4.41</version>
</probeinfo>
```

**Sample Reply (JSON)**

```json
{
  "company": "Nimsoft Corporation",
  "connections": "51",
  "libDate": "Sep 17 2010",
  "libVersion": "5.12 (64bit)",
  "messages": "1",
  "name": "cdm",
  "restarted": "1970-01-01T01:00:00+01:00",
  "started": "1969-12-21T19:54:36.832+01:00",
  "version": "4.41"
}
```
### Invoke Callback

This method allows the invocation of any callback. This method does not support nested PDSes or various Table structures. Nested PDSes and Tables are supported in callback2 (below).

**Note:** The timeout parameter is denoted in milliseconds.

<table>
<thead>
<tr>
<th>URL</th>
<th>/probe/{domain}/{hub}/{robot}/{probe}/callback/{callback}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input</td>
<td>CallbackRequest structure</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; XML/JSON PdsDocument</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Basic Management</td>
</tr>
</tbody>
</table>

#### Sample Request (XML)

```xml
POST /rest/probe/chris-dev/primaryhub/nb-1538/controller/callback/get_info HTTP/1.1
Accept: application/xml
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<callbackrequest>
    <timeout>5000</timeout>
    <parameters>
        <name>detail</name>
        <type>int</type>
        <value>1</value>
    </parameters>
</callbackrequest>
```

#### Sample Reply (XML)

```xml
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<pds>
    <entry name="robotname" datatype="string">
        <value>nb-1538</value>
    </entry>
    <entry name="hubname" datatype="string">
        <value>primaryhub</value>
    </entry>
    <entry name="os_description" datatype="string">
        <value>Build 7600</value>
    </entry>
    <entry name="timezone_name" datatype="string">
        <value>Mitteleuropäische Zeit</value>
    </entry>
    <entry name="access_0" datatype="int">
        <value>0</value>
    </entry>
    <entry name="workdir" datatype="string">
        <value>C:\Program Files (x86)\Nimsoft</value>
    </entry>
    <entry name="access_1" datatype="int">
        <value>1</value>
    </entry>
</pds>
```
Sample Request (JSON)

POST /rest/probe/chris-dev/primaryhub/nb-1538/controller/callback/get_info HTTP/1.1
Accept: application/json
Content-Type: application/json

```
{
  "timeout":"5000",
  "parameters":{
    "name":"detail",
    "type":"int",
    "value":"1"
  }
}
```

Sample Reply (JSON)

HTTP/1.1 200 OK
Content-Type: application/json

```
{
  "entry":[
    {
      "@name":"robotname",
      "@datatype":"string",
      "value": {
        "@type": "xs:string",
        "$": "nb-1538"
      }
    },
    {
      "@name":"hubname",
      "@datatype":"string",
      "value": {
        "@type": "xs:string",
        "$": "primaryhub"
      }
    },
    {
      "@name":"os_description",
      "@datatype":"string",
      "value": {
        "@type": "xs:string",
        "$": "Service Pack 1 Build 7601"
      }
    },
    {
      "@name":"timezone_name",
      "@datatype":"string",
      "value": {
        "@type": "xs:string",
        "$": "Mitteleuropäische Zeit"
      }
    },
    {
      "@name":"access_0",
      "@datatype":"int",
      "value": {
        "@type": "xs:int",
        "$": "0"
      }
    }
  ]
}
```
### Invoke Callback2

This method allows the invocation of any callback. This method accepts and returns NimPds. The NimPds schema can be found at: [http://docs.nimsoft.com/prodhelp/en_US/Monitor/SDK/REST/nimpds.xsd](http://docs.nimsoft.com/prodhelp/en_US/Monitor/SDK/REST/nimpds.xsd).

<table>
<thead>
<tr>
<th>URL</th>
<th>/probe/{domain}/{hub}/{robot}/{probe}/callback2/{callback}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input</td>
<td>CallbackRequest structure</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; XML/JSON PdsDocument</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Basic Management</td>
</tr>
</tbody>
</table>

```xml
<nimPds>
  <nimInt key="newInt">1</nimInt>
  <nimInt key="newInt2">2</nimInt>
  <nimInt key="newInt3">3</nimInt>
  <nimString key="newString1">This is a string value 1</nimString>
  <nimString key="newString2">This is a string value 2</nimString>
  <nimIntTable key="IntTable1">
    <int index="0">1</int>
    <int index="1">2</int>
    <int index="2">3</int>
  </nimIntTable>
  <nimIntTable key="IntTable12">
    <int index="0">1</int>
    <int index="1">2</int>
    <int index="2">3</int>
  </nimIntTable>
  <nimStringTable key="StringTable1">
    <string index="0">String in a table 1</string>
    <string index="1">String in a table 2</string>
    <string index="2">String in a table 3</string>
  </nimStringTable>
  <nimPds>
    <nimInt key="newInt">1</nimInt>
    <nimInt key="newInt2">2</nimInt>
    <nimInt key="newInt3">3</nimInt>
    <nimString key="newString1">This is a string value 1</nimString>
    <nimString key="newString2">This is a string value 2</nimString>
    <nimIntTable key="IntTable1">
      <int index="0">1</int>
      <int index="1">2</int>
    </nimIntTable>
  </nimPds>
</nimPds>
```

(The following is a prototype that shows all the different nestings and organization possible. The nestings are not limited. Tables may only contain objects of the type of table that it is (no mixing). Also, no “key” is necessary.)
HTTP/1.1 200 OK
Content-Type: application/json
{
    "nimInt": [ 
        { 
            "@key": "ssl_mode",
            "$": "0"
        },
        { 
            "@key": "connections",
            "$": "2271"
        },
        { 
            "@key": "started",
            "$": "-16388"
        },
        { 
            "@key": "messages",
            "$": "10"
        },
        { 
            "@key": "restarted",
            "$": "0"
        }
    ],
    "nimString": [ 
        { 
            "@key": "libdate",
            "$": "Jun 30 2011"
        },
        { 
            "@key": "ssl_version",
            "$": "OpenSSL 1.0.0c 2 Dec 2010"
        },
        { 
            "@key": "company",
            "$": "Nimsoft Corp"
        },
        { 
            "@key": "name",
            "$": "hub"
        },
        { 
            "@key": "libversion",
            "$": "S.24 (64bit)"
        },
        { 
            "@key": "ssl_cipher",
            "$": "DEFAULT"
        },
        { 
            "@key": "version",
            "$": "S.60 [Jun 30 2011]"
        }
    ]
}
## Activate / Deactivate a Probe

| **URL** | `/probe/{domain}/{hub}/{robot}/{probe}/active|inactive` |
|---------|-------------------------------------------------|
| **Method** | POST |
| **Input** | . |
| **Returns** | 204 OK 401 Unauthorized 404 Not Found |
| **Valid Users** | UIM Users |
| **Required Permissions** | Web Service, Basic Management |
| **Sample Request** | POST /rest/probe/chris-dev/primaryhub/nb-1538/cdm/active HTTP/1.1 Accept: application/xml |
| **Sample Reply** | HTTP/1.1 204 No Content |

## Get Probe Configuration

<table>
<thead>
<tr>
<th><strong>URL</strong></th>
<th><code>/probe/{domain}/{hub}/{robot}/{probe}/config</code></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>GET</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>.</td>
</tr>
<tr>
<td><strong>Returns</strong></td>
<td>200 OK &amp; XML/JSON PdsDocument List of Configuration Keys 401 Unauthorized 404 Not Found</td>
</tr>
<tr>
<td><strong>Valid Users</strong></td>
<td>UIM Users</td>
</tr>
<tr>
<td><strong>Required Permissions</strong></td>
<td>Web Service, Basic Management</td>
</tr>
<tr>
<td><strong>Sample Request</strong></td>
<td>GET /rest/probe/chris-dev/primaryhub/nb-1538/cdm/config HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>
| **Sample Reply (XML)** | HTTP/1.1 200 OK  
```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<probe-configuration>
  <probeConfigKey>
    <key>/messages/PagefileWarning/text</key>
    <value>Average ($value_number samples) memory usage is now $value$unit, which is above the warning threshold ($value_limit$unit)</value>
  </probeConfigKey>
  <probeConfigKey>
    <key>/messages/PagefileWarning/level</key>
    <value>minor</value>
  </probeConfigKey>
</probe-configuration>
```
Sample Reply (JSON)
```
<probe-configuration>
HTTP/1.1 200 OK
Content-Type: application/json
{
    "probeConfigKey": [
        {
            "key": "/disk/\alarm/\fixed/Q:\Delta_error/\threshold",
            "value": "10"
        },
        {
            "key": "/memory/\interval",
            "value": "5 \text{ min}"
        },
        {
            "key": "/memory/qos_memory_paging",
            "value": "no"
        },
        {
            "key": "/memory/qos_memory_usage",
            "value": "no"
        },
        {
            "key": "/memory/samples","value": "5"
        },
        ...
    ]
}
```

Get Single Probe Configuration Value

<table>
<thead>
<tr>
<th>URL</th>
<th>/probe/{domain}/{hub}/{robot}/{probe}/config/{section+key}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; Configuration value as String</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Basic Management</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/probe/chris-dev/primaryhub/nb-1538/\cdm/setup/resttest HTTP/1.1</td>
</tr>
<tr>
<td></td>
<td>Accept: application/xml</td>
</tr>
<tr>
<td>Sample Reply</td>
<td>HTTP/1.1 200 OK &lt;value&gt; ignoreme&lt;/value&gt;</td>
</tr>
</tbody>
</table>

Set single probe Configuration Value

<table>
<thead>
<tr>
<th>URL</th>
<th>/probe/{domain}/{hub}/{robot}/{probe}/config</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Input</td>
<td>ProbeConfigKey structure</td>
</tr>
<tr>
<td>Returns</td>
<td>204 OK</td>
</tr>
<tr>
<td><strong>Delete Single Probe Configuration Value</strong></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td><strong>URL</strong></td>
<td><code>/probe/config/{domain}/{hub}/{robot}/{probe}/{section+key}</code></td>
</tr>
<tr>
<td><strong>Method</strong></td>
<td>DELETE</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>-</td>
</tr>
<tr>
<td><strong>Returns</strong></td>
<td>204 OK</td>
</tr>
<tr>
<td><strong>Valid Users</strong></td>
<td>UIM Users</td>
</tr>
<tr>
<td><strong>Required Permissions</strong></td>
<td>Web Service, Basic Management</td>
</tr>
</tbody>
</table>
| **Sample Request** | `DELETE /rest/probe/chris-dev/primaryhub/nb-1538/cdm/config/setup/resttest HTTP/1.1
Accept: application/xml` |
**Set Multiple Probe Configuration Values**

In order to delete configuration values, supply an empty value parameter. In the future, a DELETE call for this purpose will be in place.

<table>
<thead>
<tr>
<th>URL</th>
<th>/probe/config /{domain}/{hub}/{robot}/{probe}/config</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Basic Management</td>
</tr>
<tr>
<td>Input</td>
<td>ProbeConfigKeys structure</td>
</tr>
<tr>
<td>Returns</td>
<td>204 OK</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
</tbody>
</table>
| Sample Request (XML) | PUT /rest/probe/chris-dev/primaryhub/nb-1538/cdm/config HTTP/1.1
Accept: application/xml
Content-Type: application/xml

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<probeConfigKeys>
  <configkey>
    <key>/setup/resttest</key>
    <value>ignoreme</value>
  </configkey>
  <configkey>
    <key>/setup/resttest2</key>
    <value>ignoremetoo</value>
  </configkey>
</probeConfigKeys>
```

| Sample Reply (XML) | HTTP/1.1 204 No Content |
| Sample Request (JSON) | PUT /rest/probe/chris-dev/primaryhub/nb-1538/cdm/config HTTP/1.1
Accept: application/json
Content-Type: application/json |

```json
{"configkey": [{
  "key": "/setup/resttest",
  "value": "ignoreme"
}, {
  "key": "/setup/resttest2",
  "value": "ignoremetoo"
}]
```

| Sample Reply | HTTP/1.1 204 No Content |
# Alarm Related Calls

## Create Alarm

<table>
<thead>
<tr>
<th>URL</th>
<th>/alarms/createAlarm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; Alarm ID XML/JSON</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
</tbody>
</table>

### Valid Users
- UIM Users

### Required Permissions
- Web Service
- Alarm Management

### Sample Request (XML)

```xml
POST /rest/alarms/createAlarm HTTP/1.1
Accept: application/xml
Content-Type: application/xml

<alarm>
    <assignedBy/>
    <assignedTo/>
    <custom1/>
    <custom2/>
    <custom3/>
    <custom4/>
    <custom5/>
    <devId/>
    <domain>w2k8r2-x64-lc_domain</domain>
    <hostname>w2k8r2-x64-lc</hostname>
    <hub>w2k8r2-x64-lc_hub</hub>
    <level>1</level>
    <message>Testmessage</message>
    <metId/>
    <metId/>
    <nas>w2k8r2-x64-lc_hub</nas>
    <origin>w2k8r2-x64-lc_hub</origin>
    <prevLevel>0</prevLevel>
    <probe/>
    <probe/>
    <robot>w2k8r2-x64-lc</robot>
    <severity>Informational</severity>
    <source>w2k8r2-x64-lc</source>
    <subsystem>Nimsoft</subsystem>
    <subsystemId>1</subsystemId>
    <suppressionCount>0</suppressionCount>
    <suppressionKey/>
    <suppressionKey/>
```
Sample Reply (XML)

<table>
<thead>
<tr>
<th>HTTP/1.1 200 OK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content-Type: application/xml</td>
</tr>
<tr>
<td>&lt;?xml version=&quot;1.0&quot; encoding=&quot;UTF-8&quot; standalone=&quot;yes&quot;?&gt;</td>
</tr>
<tr>
<td>&lt;alarm&gt;</td>
</tr>
<tr>
<td>&lt;id&gt;OU50853950-04896&lt;/id&gt;</td>
</tr>
<tr>
<td>&lt;/alarm&gt;</td>
</tr>
</tbody>
</table>

Sample Request (JSON)

| POST /rest/alarms/createAlarm HTTP/1.1 |
| Accept: application/JSON |
| Content-Type: application/JSON |
| { |
| "assignedBy":"", |
| "assignedTo":"", |
| "custom1":"", |
| "devId":"", |
| "domain":"w2k8r2-x64-lc_domain", |
| "hostname":"w2k8r2-x64-lc", |
| "hub":"w2k8r2-x64-lc_hub", |
| "level":"1", |
| "message":"Testmessage", |
| "metid":"", |
| "nas":"w2k8r2-x64-lc_hub", |
| "origin":"w2k8r2-x64-lc_hub", |
| "prevLevel":"0", |
| "probe":"", |
| "robot":"w2k8r2-x64-lc", |
| "severity":"Informational", |
| "source":"w2k8r2-x64-lc", |
| "subsystem":"Nimsoft", |
| "subsystemid":"1", |
| "suppressionCount":"0", |
| "suppressionKey":"", |
| "visible":"true" |
| } |

Sample Reply (JSON)

| HTTP/1.1 200 OK |
| Content-Type: application/json |
| { |
| "id": "OU50853950-04945" |
| } |

**Get Alarm Summary**

Returns a structure containing alarm counts for the different alarm severities.

<p>| URL | /alarms/summary |
| Method | GET |
| Input | - |
| Returns | 200 OK &amp; AlarmSummary structure (XML/JSON) |
| | 401 Unauthorized |
| | 404 Not Found |</p>
<table>
<thead>
<tr>
<th>Valid Users</th>
<th>UIM Users, Account Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required Permissions</td>
<td>Web Service, Alarm Summary</td>
</tr>
<tr>
<td>Sample Request (XML)</td>
<td>GET /rest/alarms/summary HTTP/1.1</td>
</tr>
<tr>
<td></td>
<td>Accept: application/xml</td>
</tr>
<tr>
<td>Sample Reply (XML)</td>
<td>HTTP/1.1 200 OK</td>
</tr>
<tr>
<td></td>
<td>&lt;?xml version=&quot;1.0&quot; encoding=&quot;UTF-8&quot; standalone=&quot;yes&quot;?&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;alarmsummary&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;clear&gt;0&lt;/clear&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;critical&gt;12&lt;/critical&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;information&gt;47&lt;/information&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;major&gt;19&lt;/major&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;minor&gt;9&lt;/minor&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;warning&gt;13&lt;/warning&gt;</td>
</tr>
<tr>
<td>Sample Request (XML)</td>
<td>&lt;/alarmsummary&gt;</td>
</tr>
<tr>
<td>Sample Request (JSON)</td>
<td>GET /rest/alarms/summary HTTP/1.1</td>
</tr>
<tr>
<td></td>
<td>Accept: application/json</td>
</tr>
<tr>
<td>Sample Reply (JSON)</td>
<td>HTTP/1.1 200 OK</td>
</tr>
<tr>
<td></td>
<td>Content-Type: application/json</td>
</tr>
<tr>
<td></td>
<td>{</td>
</tr>
<tr>
<td>Get Unfiltered List of Alarms</td>
<td>&quot;clear&quot;:&quot;0&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;critical&quot;:&quot;7&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;information&quot;:&quot;24&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;major&quot;:&quot;9&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;minor&quot;:&quot;6&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;warning&quot;:&quot;6&quot;</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Alarm Details</td>
</tr>
<tr>
<td>Sample Request (XML)</td>
<td>GET /rest/alarms HTTP/1.1</td>
</tr>
<tr>
<td></td>
<td>Accept: application/xml</td>
</tr>
<tr>
<td>Sample Reply (XML)</td>
<td>HTTP/1.1 200 OK</td>
</tr>
<tr>
<td></td>
<td>Content-Type: application/xml</td>
</tr>
</tbody>
</table>
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<alarm-list>
  <alarm>
    <assignedBy></assignedBy>
    <assignedTo></assignedTo>
    <custom1></custom1>
    <custom2></custom2>
    <custom3></custom3>
    <custom4></custom4>
    <custom5></custom5>
    <domain>chris-dev</domain>
    <hostname>MFC7840W</hostname>
    <hub>primaryhub</hub>
    <id>DL31165458-84280</id>
    <level>5</level>
    <message>The SNMP agent on '192.168.0.119' is NOT responding.</message>
    <nas>primaryhub</nas>
    <origin>primaryhub</origin>
    <prevLevel>0</prevLevel>
    <probe>interface_traffic</probe>
    <robot>win-cxp6lp7V6g</robot>
    <severity>Critical</severity>
    <source>MFC7840W</source>
    <Subsystem>Network</Subsystem>
    <subsystemId>1.1.3</subsystemId>
    <suppressionCount>0</suppressionCount>
    <suppressionKey>NetTfc/state-192.168.0.119</suppressionKey>
    <timeArrival>2011-11-02T11:27:40+01:00</timeArrival>
    <timeOrigin>2011-11-02T11:27:39+01:00</timeOrigin>
    <timeLast>2011-11-02T11:27:40+01:00</timeLast>
    <userTag1></userTag1>
    <userTag2></userTag2>
    <visible>true</visible>
  </alarm>
  <alarm>
    <assignedBy></assignedBy>
    <assignedTo></assignedTo>
    <custom1></custom1>
  ...
</alarm>
</alarm-list>

Sample Request (JSON)
GET /rest/alarms HTTP/1.1
Accept: application/json

Sample Reply (JSON)
HTTP/1.1 200 OK
Content-Type: application/json
Transfer-Encoding: chunked

2000
"alarm": [
  
  "assignedBy": "",
  "assignedTo": "",
  "custom1": "",
  "custom2": "",
  "custom3": "",
  "custom4": "",
  "custom5": "",
  "domain": "chris-dev",
  "hostname": "win-cxp6lp7V6g",
  "hub": "primaryhub",
  "id": "RH99670515-00059",
]
Get Filtered List of Alarms

This call returns a filtered list of alarms that are visible to the user. For details on the Alarm Filter structure passed to this call, refer to “Message Structure Definition: Alarm Filter”.

<table>
<thead>
<tr>
<th>URL</th>
<th>/alarms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input</td>
<td>AlarmFilter structure</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; AlarmListDocument XML/JSON</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Alarm Details</td>
</tr>
</tbody>
</table>

Sample Request

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<alarmfilter>
  <level>2,3,4,5</level>
</alarmfilter>
```

Sample Reply

```
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
```
<alarm-list>
  <alarm>
    <assignedBy>
    <assignedTo>
    <custom1></custom1>
    <custom2></custom2>
    <custom3></custom3>
    <custom4></custom4>
    <custom5></custom5>
    <domain>chris-dev</domain>
    <hostname>MFC7840W</hostname>
    <hub>primaryhub</hub>
    <id>DL31165458-84280</id>
    <level>5</level>
    <message>The SNMP agent on '192.168.0.119' is NOT responding.</message>
    <nas>primaryhub</nas>
    <origin>primaryhub</origin>
    <prevLevel>0</prevLevel>
    <probe>interface_traffic</probe>
    <robot>win-cxp6lptV6g</robot>
    <severity>Critical</severity>
    <source>MFC7840W</source>
    <subsystem>Network</subsystem>
    <subsystemId>1.1.3</subsystemId>
    <suppressionCount>0</suppressionCount>
    <suppressionKey>NetTfc/state-192.168.0.119</suppressionKey>
    <timeArrival>2011-11-02T11:27:40+01:00</timeArrival>
    <timeOrigin>2011-11-02T11:27:39+01:00</timeOrigin>
    <timeLast>2011-11-02T11:27:40+01:00</timeLast>
    <userTag1></userTag1>
    <userTag2></userTag2>
    <visible>true</visible>
  </alarm>
  <alarm>
    <assignedBy>
    <assignedTo>
    <custom1></custom1>
    ...  
  </alarm>
</alarm-list>

**Sample Request (JSON)**

```
POST /rest/alarms HTTP/1.1
Accept: application/json
Content-Type: application/json

{
    "assigned_to":"userA",
    "custom1":"lorum",
    "custom2":"ipsum",
    "custom3":"dolor",
    "custom4":"sit",
    "custom5":"amet",
    "domain":"domainA",
    "hostname":"hostA",
    "hub":"hubB",
    "level":2,3,4,5,6,
    "message":"message text to search for",
    "message_count":">2",
    "origin":"customerA",
    "probe":"cdm",
    "robot":"robotC",
    "source":"maecenas",
    "subsystem":"cpu",
}```
HTTP/1.1 200 OK
Content-Type: application/json
Transfer-Encoding: chunked

2000
{"alarm": [
{
"assignedBy": ",",
"assignedTo": ",",
"custom1": ",",
"custom2": ",",
"custom3": ",",
"custom4": ",",
"custom5": ",",
"domain": "chris-dev",
"hostname": "win-cxp6lpt7v6g",
"hub": "primaryhub",
"id": "RH99670515-00059",
"level": "1",
"message": "USER32(1074 - None): The process C:\Windows\system32\winlogon.exe (WIN-CXP6LPT7V6G) has initiated the power off of computer WIN-CXP6LPT7V6G.",
"nas": "primaryhub",
"origin": "primaryhub",
"prevLevel": "0",
"probe": "ntevl",
"robot": "win-cxp6lpt7v6g",
"severity": "Informational",
"source": "192.168.211.129",
"subsystem": "System",
"subsystemId": "1.1.11.1.3",
"suppressionCount": "0",
"suppressionKey": "",
"timeArrival": "2011-11-11T12:54:35+01:00",
"timeOrigin": "2011-11-11T12:54:35+01:00",
"timeLast": "2011-11-11T12:54:35+01:00",
"userTag1": "utag1",
"userTag2": "utag2",
"visible": "true"
},
{
"assignedBy": ",",
"assignedTo": ",",
"visible": "true"
}]
}
### Get State for all Alarms
This call returns the severity of the most critical alarm currently visible to the user.

<table>
<thead>
<tr>
<th>URL</th>
<th>/alarms/state</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; AlarmSeverity structure XML/JSON</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Alarm Summary</td>
</tr>
<tr>
<td>Sample Request (XML)</td>
<td>GET /rest/alarms/state HTTP/1.1 Accept: application/xml</td>
</tr>
<tr>
<td>Sample Reply (XML)</td>
<td>&lt;?xml version=&quot;1.0&quot; encoding=&quot;UTF-8&quot; standalone=&quot;yes&quot;?&gt; &lt;alarm-severity&gt; &lt;level&gt;5&lt;/level&gt; &lt;severity&gt;Critical&lt;/severity&gt; &lt;/alarm-severity&gt;</td>
</tr>
<tr>
<td>Sample Request (JSON)</td>
<td>GET /rest/alarms/state HTTP/1.1 Accept: application/json</td>
</tr>
<tr>
<td>Sample Reply (JSON)</td>
<td>HTTP/1.1 200 OK Content-Type: application/json</td>
</tr>
<tr>
<td></td>
<td>{</td>
</tr>
<tr>
<td></td>
<td>&quot;level&quot;:&quot;5&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;severity&quot;:&quot;Critical&quot;</td>
</tr>
<tr>
<td></td>
<td>}</td>
</tr>
</tbody>
</table>

### Get State for Filtered Alarms
This call returns the severity of the most critical alarm currently open that matches the given alarm filter. For details on the Alarm Filter structure passed to this call, refer to “Message Structure Definition: Alarm Filter”.

<table>
<thead>
<tr>
<th>URL</th>
<th>/alarms/state</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input</td>
<td>AlarmFilter structure</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; AlarmSeverity structure XML/JSON</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Alarm Summary</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------</td>
</tr>
</tbody>
</table>
| **Sample Request (XML)** | POST /rest/alarms/state HTTP/1.1  
Accept: application/xml  
Content-Type: application/xml  

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<alarmfilter>
    <level>2,3,4,5,6</level>
</alarmfilter>
```

Filtering by assigned person:
```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<alarmfilter>
    <assigned_to>administrator</assigned_to>
</alarmfilter>
```

Filtering by alarm id:
```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<alarmfilter>
    <id>LC29347578-03846</id>
</alarmfilter>
```

<table>
<thead>
<tr>
<th>Sample Reply (XML)</th>
<th>Sample Request (JSON)</th>
</tr>
</thead>
</table>
| POST /rest/alarms/state HTTP/1.1  
Accept: application/json  
Content-Type: application/json  

```json
{
    "assigned_to":"userA",
    "custom1":"lorum",
    "custom2":"ipsum",
    "custom3":"dolor",
    "custom4":"sit",
    "custom5":"amet",
    "domain":"domainA",
    "hostname":"hostA",
    "hub":"hubB",
    "level":"2,3,4,5,6",
    "message":"message text to search for",
    "message_count">2",
    "origin":"customerA",
    "probe":"cdm",
    "robot":"robotC",
    "source":"maecenas",
    "subsystem":"cpu",
    "subsystem_id":"1.3.1",
    "timeArrival":"2011-11-22T12:54:35+01:00",
    "timeLast":"2011-11-22T12:54:35+01:00",
    "userTag1":"utag1",
    "userTag2":"utag2",
    "visible":"true"
}
```

<table>
<thead>
<tr>
<th>Sample Reply (JSON)</th>
<th>Sample Reply (JSON)</th>
</tr>
</thead>
</table>
| HTTP/1.1 200 OK  
Content-Type: application/json  

## Accept an Alarm
This call assigns the alarm to the current user. It lets the user assign the alarm to itself.

<table>
<thead>
<tr>
<th>URL</th>
<th>/alarms/{alarmid}/accept</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required</td>
<td>Web Service, Accept</td>
</tr>
<tr>
<td>Permissions</td>
<td></td>
</tr>
<tr>
<td>Sample Request</td>
<td>PUT /rest/alarms/DL31165458-85308/accept HTTP/1.1</td>
</tr>
<tr>
<td>Sample Reply</td>
<td>HTTP/1.1 204 No Content</td>
</tr>
</tbody>
</table>

## Acknowledge an Alarm
This call clears/closes the specified alarm.

<table>
<thead>
<tr>
<th>URL</th>
<th>/alarms/{alarmid}/ack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (= OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required</td>
<td>Web Service, Acknowledge</td>
</tr>
<tr>
<td>Permissions</td>
<td></td>
</tr>
<tr>
<td>Sample Request</td>
<td>PUT /rest/alarms/DL31165458-85308/ack HTTP/1.1</td>
</tr>
<tr>
<td>Sample Reply</td>
<td>HTTP/1.1 204 No Content</td>
</tr>
</tbody>
</table>

## Assign an Alarm

| URL            | /alarms/{alarmid}/assign/{assignToUsername} |
### Method
- PUT

### Input
- 

### Returns
- 204 No Content (=OK)
- 401 Unauthorized
- 404 Not Found

### Valid Users
- UIM User

### Require Permissions
- Web Service, Assign

### Sample Request
```
PUT /rest/alarms/DL31165458-85308/assign/operatorA HTTP/1.1
```

### Sample Reply
```
HTTP/1.1 204 No Content
```

### Unassign an Alarm

#### URL
- /alarms/{alarmid}/unassign

#### Method
- PUT

#### Input
- 

#### Returns
- 204 No Content (=OK)
- 401 Unauthorized
- 404 Not Found

#### Valid Users
- UIM User

#### Require Permissions
- Web Service, Unassign

#### Sample Request
```
PUT /rest/alarms/DL31165458-85308/unassign HTTP/1.1
```

#### Sample Reply
```
HTTP/1.1 204 No Content
```
## Account Related Calls

### Get All Accounts

<table>
<thead>
<tr>
<th>URL</th>
<th>/accounts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; AccountList structure XML/JSON</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/accounts HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>

**Sample Reply (XML)**

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<accountlist>
  <account>
    <address></address>
    <city></city>
    <country></country>
    <creationDate>2011-10-11T09:42:42+02:00</creationDate>
    <description></description>
    <fax></fax>
    <id>3</id>
    <name>companyA</name>
    <origin>primaryhub</origin>
    <origin>anotherOrigin</origin>
    <phone></phone>
    <postalCode></postalCode>
    <state></state>
    <website></website>
  </account>
</accountlist>
```

**Sample Reply (JSON)**

```json
HTTP/1.1 200 OK
Content-Type: application/json

{
   "account":{
       "accountId":"3",
       "address":"",
       "city":"",
       "country":"",
       "creationDate":"2011-10-11T09:42:42+02:00",
       "description":"",
       "fax":"",
       "name":"companyA",
       "origin":"primaryhub",
       "origin":"anotherOrigin",
       "phone":"",
       "postalCode":"",
       "state":"",
       "website":"
   }
}```
Get an Account

<table>
<thead>
<tr>
<th>URL</th>
<th>/accounts/{account_id}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; AccountDocument XML/JSON</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account User</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/accounts/9 HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>

Sample Reply (XML)

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<account>
  <address>Teststreet 42</address>
  <city>Testcity</city>
  <country>TestCountry</country>
  <creationDate>2011-11-02T19:15:32.513+01:00</creationDate>
  <description>Test Description</description>
  <fax>Test Fax no 4711</fax>
  <id>8</id>
  <name>REST Services Test Account</name>
  <origin>primaryhub</origin>
  <origin>anotherOrigin</origin>
  <phone>Test phone 0815</phone>
  <postalCode>123456</postalCode>
  <state>Teststate</state>
  <website>www.testrest.com</website>
</account>
```

Sample Reply

HTTP/1.1 200 OK
Content-Type: application/json
Create a New Account

The UIM system is not aware of origins immediately after installation. For this reason, and to allow accounts to be created immediately after installation, there is no check that an origin exists in the system. Users associated to an account with an origin that is not defined in the system can log into UMP and Admin Console. However, they will not see any infrastructure until the origin is defined in the system. Origins defined in the REST create account call but not defined elsewhere in the system, will be displayed in UMP for selection when creating accounts.

Multiple origins can be associated to the account by enclosing the origin in the <origin> tag for XML, or by using “origin”:“originName” for JSON. See examples.

<table>
<thead>
<tr>
<th>URL</th>
<th>/accounts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input</td>
<td>AccountDocument</td>
</tr>
<tr>
<td></td>
<td>200 OK &amp; AccountDocument XML/JSON</td>
</tr>
<tr>
<td>Returns</td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td></td>
<td>409 Resource Conflict (if the Account already exists)</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
<tr>
<td>Sample Request (XML)</td>
<td>POST /rest/accounts HTTP/1.1 Accept: application/xml Content-Type: application/xml</td>
</tr>
</tbody>
</table>

```xml
<account>
  <address>Teststreet 42</address>
  <city>Testcity</city>
  <country>TestCountry</country>
  <description>Test Description</description>
  <fax>Test Fax no 4711</fax>
  <name>REST Services Test Account</name>
</account>
```
Sample Reply (XML)

HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<account>
  <address>Teststreet 42</address>
  <city>Testcity</city>
  <country>TestCountry</country>
  <creationDate>2011-11-22T13:23:05.870+01:00</creationDate>
  <description>Test Description</description>
  <fax>Test Fax no 4711</fax>
  <name>REST Services Test Account</name>
  <origin>primaryhub</origin>
  <origin>anotherOrigin</origin>
  <phone>Test phone 0815</phone>
  <postalCode>123456</postalCode>
  <state>Teststate</state>
  <webSite>www.testrest.com</webSite>
</account>

Sample Request (JSON)

POST /rest/accounts HTTP/1.1
Accept: application/json
Content-Type: application/json

{
  "accountId": "0",
  "address": "Teststreet 42",
  "city": "Testcity",
  "country": "TestCountry",
  "description": "Test Description",
  "fax": "Test Fax no 4711",
  "name": "REST Services Test Account",
  "origin": "primaryhub",
  "origin": "anotherOrigin",
  "phone": "Test phone 0815",
  "postalCode": "123456",
  "state": "Teststate",
  "webSite": "www.testrest.com"
}

Sample Reply (JSON)

HTTP/1.1 200 OK
Content-Type: application/json

{
  "accountId": "36",
  "address": "Teststreet 42",
  "city": "Testcity",
  "country": "TestCountry",
  "creationDate": "2011-11-22T13:23:05.870+01:00",
  "description": "Test Description",
  "fax": "Test Fax no 4711",
  "name": "REST Services Test Account",
  "origin": "primaryhub",
  "origin": "anotherOrigin",
  "phone": "Test phone 0815",
  "postalCode": "123456",
}
## Update an Existing Account

Note: Specifying one or more origins when updating an Account does not replace the existing origins. It only adds additional origins. Use the “Remove Origin from Account” call to remove origins. Empty tags (<origin></origin>), tags with white space (<origin>  </origin>), and duplicate origins are ignored.

<table>
<thead>
<tr>
<th>URL</th>
<th>/accounts/{account_id}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Input</td>
<td>AccountDocument</td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
</tbody>
</table>

### Sample Request (XML)

PUT /rest/accounts/8 HTTP/1.1  
Content-Type: application/xml  

```xml  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<account>  
  <accountId>8</accountId>  
  <address>Teststreet 42</address>  
  <city>Testcity</city>  
  <country>TestCountry</country>  
  <description>Test Description</description>  
  <fax>Test Fax no 4711</fax>  
  <name>REST Services Test Account</name>  
  <origin>primaryhub</origin>  
  <phone>Test phone 0815</phone>  
  <postalCode>123456</postalCode>  
  <state>Teststate</state>  
  <website>www.testrest.com</website>  
</account>  
```

### Sample Reply

HTTP/1.1 204 No Content

### Sample Request (JSON)

PUT /rest/accounts/8 HTTP/1.1  
Content-Type: application/json  

```json  
{  
  "accountId": "8",  
  "address": "Teststreet 42",  
  "city": "Testcity",  
  "country": "TestCountry",  
  "description": "Test Description",  
  "fax": "Test Fax no 4711",  
  "name": "REST Services Test Account",  
  "origin": "primaryhub",  
  "phone": "Test phone 0815",  
  "postalCode": "123456",  
}  
```
Delete an Account

<table>
<thead>
<tr>
<th>URL</th>
<th>/accounts/{account_id}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
<tr>
<td>Sample Request</td>
<td>DELETE /rest/accounts/14 HTTP/1.1</td>
</tr>
<tr>
<td>Sample Reply</td>
<td>HTTP/1.1 204 No Content</td>
</tr>
</tbody>
</table>

Get Account Contacts

This call can be used to retrieve the contacts associated with the Account. In order to create/delete contacts, please see the Contact-related calls.

<table>
<thead>
<tr>
<th>URL</th>
<th>/accounts/{account_id}/contacts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – ContactList structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/accounts/3/contacts HTTP/1.1 Accept: application/xml</td>
</tr>
<tr>
<td>Sample Reply</td>
<td>HTTP/1.1 200 OK Content-Type: application/xml</td>
</tr>
</tbody>
</table>
XML

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<contactlist>
  <contact>
    <accountId>3</accountId>
    <acl>Guest</acl>
    <contactId>4</contactId>
    <creationDate>2011-10-28T16:45:18+02:00</creationDate>
    <department></department>
    <description></description>
    <email></email>
    <firstName></firstName>
    <lastName></lastName>
    <loginName>contact</loginName>
    <mobile></mobile>
    <password>a9Ki6Vu+O6S56vAZHXK95Q==</password>
    <phone></phone>
    <title></title>
  </contact>
  <contact>
    <accountId>3</accountId>
    <acl>Administrator</acl>
    <contactId>5</contactId>
    <creationDate>2011-10-28T16:45:18+02:00</creationDate>
    <department></department>
    <description></description>
    <email></email>
    <firstName></firstName>
    <lastName></lastName>
    <loginName>contactadmin</loginName>
    <mobile></mobile>
    <password>a9Ki6Vu+O6S56vAZHXK95Q==</password>
    <phone></phone>
    <title></title>
  </contact>
</contactlist>

Sample Reply
(JSON)

HTTP/1.1 200 OK
Content-Type: application/json

{
    "contact": [
        {
            "accountId": "3",
            "acl": "AccountOp",
            "contactId": "2",
            "creationDate": "2011-10-11T09:42:42+02:00",
            "department": ":",
            "description": ":",
            "email": "chris@lala.com",
            "firstName": ":",
            "lastName": ":",
            "loginName": "chris",
            "mobile": ":",
            "password": "osfPlF9dQcjkJ66HmL8pg==",
            "phone": ":",
            "title": ":"
        },
        {
            "accountId": "3",
            "acl": "AccountOp",
            "contactId": "3",
            "creationDate": "2011-10-11T09:51:38+02:00",
            "department": ":",
            "description": ":",
            "email": "chris@lala.com",
            "firstName": ":",
            "lastName": ":",
            "loginName": "chris",
            "mobile": ":",
            "password": "osfPlF9dQcjkJ66HmL8pg==",
            "phone": ":",
            "title": ":"
        }
    ]
}
Get Account Configuration Items

**URL**
/accounts/{account_id}/configuration_items

**Method**
GET

**Input**
-

**Returns**
200 OK – ConfigurationItemList structure
401 Unauthorized
404 Not Found

**Valid Users**
UIM Users

**Required Permissions**
Web Service, Account Administration

**Sample Request**
GET /rest/accounts/32/configuration_items HTTP/1.1
Accept: application/xml

**Sample Reply (XML)**
HTTP/1.1 200 OK
Content-Type: application/xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ci-list>
  <ci>C0869202F0CCE4AFD41EE074E9AAF4D25</ci>
</ci-list>

**Sample Reply (JSON)**
HTTP/1.1 200 OK
Content-Type: application/json
{
  "ci":"C0869202F0CCE4AFD41EE074E9AAF4D25"
}

Add Account Configuration Items

**Note:** If you get a foreign key constraint error on "FK_CM_CONFIGURATION_ITEM_OWNERSHIP_CI" it is likely that the configuration item id supplied is not a valid input. One way of finding a valid ci id is by using the "get CI Metric" call.

**URL**
/accounts/{account_id}/configuration_items

{...}
"description":"","email":"","firstName":"","lastName":"","loginName":"remko","mobile":"","password":"MNL1sYsagL8G9FJjKpsKw==","phone":"","title":""}
<table>
<thead>
<tr>
<th>Method</th>
<th>POST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>ConfigurationItemList structure containing ConfigurationItemIds</td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
</tbody>
</table>
| Sample Request (XML) | POST /rest/accounts/32/configuration_items HTTP/1.1  
Content-Type: application/xml  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<ci-list>  
<ci>C0869202F0CCE4AFD41EE074E9AAF4D25</ci>  
</ci-list> |
| Sample Reply (XML) | HTTP/1.1 204 No Content |
| Sample Request (JSON) | POST /rest/accounts/39/configuration_items HTTP/1.1  
Accept: application/json  
Content-Type: application/json  
{  
   "ci":"C0869202F0CCE4AFD41EE074E9AAF4D25"  
} |
| Sample Reply (JSON) | HTTP/1.1 204 No Content |

**Remove Account Configuration Items**

<table>
<thead>
<tr>
<th>URL</th>
<th>/accounts/{account_id}/configuration_items</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Input</td>
<td>ConfigurationItemList structure</td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
</tbody>
</table>
| Sample Request (XML) | DELETE /rest/accounts/32/configuration_items HTTP/1.1  
Content-Type: application/xml  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<ci-list>  
</ci-list> |
### Sample Reply (XML)

HTTP/1.1 204 No Content

### Sample Request (JSON)

DELETET /rest/accounts/39/configuration_items HTTP/1.1
Content-Type: application/json

```
{
  "ci":"C0869202F0CCE4AFD41EE074E9AAF4D25"
}
```

### Sample Reply (JSON)

HTTP/1.1 204 No Content

### Get Account Computer Systems

**URL**

/accounts/{account_id}/computer_systems

**Method**

GET

**Input**

-

**Returns**

200 OK – ComputerSystemList structure

401 Unauthorized

404 Not Found

**Valid Users**

UIM Users

**Required Permissions**

Web Service, Account Administration

**Sample Request**

GET /rest/accounts/32/computer_systems HTTP/1.1
Accept: application/xml

**Sample Reply (XML)**

HTTP/1.1 200 OK
Content-Type: application/xml

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<cs-list>
  <cs>9</cs>
  <cs>10</cs>
</cs-list>
```

**Sample Reply (JSON)**

HTTP/1.1 200 OK
Content-Type: application/json

```
"cs-list":
  {
    "cs":["9","10"]
  }
```

### Add Account Computer Systems

**URL**

/accounts/{account_id}/computer_systems

**Method**

POST
### Input
ComputerSystemList structure containing Computer System IDs

### Returns
- 204 No Content (=OK)
- 401 Unauthorized
- 404 Not Found

### Valid Users
UIM Users

### Required Permissions
Web Service, Account Administration

#### Sample Request (XML)
```
POST /rest/accounts/32/computer_systems HTTP/1.1
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<cs-list>
  <cs>9</cs>
  <cs>10</cs>
</cs-list>
```

#### Sample Reply (XML)
HTTP/1.1 204 No Content

#### Sample Request (JSON)
```
POST /rest/accounts/39/computer_systems HTTP/1.1
Accept: application/json
Content-Type: application/json

{  
  "cs": ["9","10"]  
}
```

#### Sample Reply (JSON)
HTTP/1.1 204 No Content

### Remove Account Computer Systems

#### URL
/accounts/{account_id}/computer_systems

#### Method
DELETE

#### Input
ComputerSystemList structure containing Computer System IDs

#### Returns
- 204 No Content (=OK)
- 401 Unauthorized
- 404 Not Found

#### Valid Users
UIM Users

#### Required Permissions
Web Service, Account Administration

#### Sample Request (XML)
```
DELETE /rest/accounts/32/computer_systems HTTP/1.1
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<cs-list>
  <cs>9</cs>
</cs-list>
```

#### Sample Reply (XML)
HTTP/1.1 204 No Content
## Add Origin to an Account

<table>
<thead>
<tr>
<th>URL</th>
<th>/accounts/{account_id}/origins</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input</td>
<td>Origins to be added</td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
</tbody>
</table>

### Sample Request (XML)

```
POST /rest/accounts/54/origins HTTP/1.1
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<origins>
    <origin>Origin1</origin>
    <origin>Origin2</origin>
</origins>
```

### Sample Reply (XML)

HTTP/1.1 204 No Content

### Sample Request (JSON)

```
POST /rest/accounts/54/origins HTTP/1.1
Content-Type: application/json

{  
    "origin": ["Origin1", "Origin2"]
}
```

### Sample Reply (JSON)

HTTP/1.1 204 No Content
**Remove Origin from Account**

Accounts must always have an origin. The origin cannot be removed if it is the last origin associated with the account.

<table>
<thead>
<tr>
<th>URL</th>
<th>/accounts/{account_id}/origin/{origin name}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Input</td>
<td>None</td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
<tr>
<td>Sample Request (XML)</td>
<td>DELETE /rest/accounts/54/origin/Origin1 HTTP/1.1</td>
</tr>
<tr>
<td>Sample Reply (XML)</td>
<td>HTTP/1.1 204 No Content</td>
</tr>
<tr>
<td>Sample Request (JSON)</td>
<td>POST /rest/accounts/54/origin/Origin1 HTTP/1.1</td>
</tr>
<tr>
<td>Sample Reply (JSON)</td>
<td>HTTP/1.1 204 No Content</td>
</tr>
</tbody>
</table>
Contact related calls

If UMP 8.2 and later is installed, this API supports updating Liferay users as well as Account Contact users. Language can now be set for Liferay users. Setting language will have no impact for Account Contact users. In addition, email is required in order to create, update, or delete a Liferay user that uses email authentication. Email is not required for Account Contact users. If UMP is not installed or versions of UMP older than 8.2 are installed, the Liferay user support is not applicable. See Appendix C for valid values for setting the Liferay User language.

Get a Contact

<table>
<thead>
<tr>
<th>URL</th>
<th>/contacts/{contact_id}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – Contact structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/contacts/5 HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>

Sample Reply (XML)

HTTP/1.1 200 OK
Content-Type: application/xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<contact>
    <accountId>3</accountId>
    <acl>Administrator</acl>
    <contactId>5</contactId>
    <creationDate>2011-10-28T16:45:18+02:00</creationDate>
    <department></department>
    <description></description>
    <email></email>
    <firstName></firstName>
    <language></language>
    <lastName></lastName>
    <loginName>contactadmin</loginName>
    <mobile></mobile>
    <password>a9Ki6Vu+O6S56vAZHtXK95Q==</password>
    <phone></phone>
    <title></title>
</contact>

Sample Reply (JSON)

HTTP/1.1 200 OK
Content-Type: application/json
{

Note: The language field is only included if the user’s language is set.
Create a New Contact

Please note that the accountId specified in the request xml/JSON must be the id of a valid account. If it is not, you will get a foreign key constraint error.

<table>
<thead>
<tr>
<th>URL</th>
<th>/contacts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input</td>
<td>Contact structure</td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
</tbody>
</table>

Sample Request (XML)

```xml
POST /rest/contacts HTTP/1.1
Accept: application/xml
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<contact>
  <accountId>21</accountId>
  <email>chris@lala.com</email>
  <firstName>Testfirstname</firstName>
  <language>English</language>
  <lastName>Testlastname</lastName>
  <loginName>testcontact</loginName>
  <password>lalala</password>
</contact>
```

Sample Reply (XML)

```
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<contact>
  <accountId>21</accountId>
  <contactId>8</contactId>
  <creationDate>2011-11-03T15:47:29.703+01:00</creationDate>
</contact>
```
### Update an Existing Contact

**URL**
/contacts/{contact_id}

**Method**
PUT

**Input**
ContactDocument

**Returns**
204 No Content (=OK)
401 Unauthorized
404 Not Found

**Valid Users**
UIM Users

**Required Permissions**
Web Service, Account Administration

**Sample Request (XML)**

```xml
<contact>
  <!-- Contact details here -->
</contact>
```
### Update an Existing Contacts Password

<table>
<thead>
<tr>
<th><strong>URL</strong></th>
<th>/contacts/{contact_id}/password</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>PUT</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>ContactDocument with new clear text password</td>
</tr>
<tr>
<td><strong>Returns</strong></td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td><strong>Valid Users</strong></td>
<td>UIM Users, Account User</td>
</tr>
<tr>
<td><strong>Required Permissions</strong></td>
<td>Web Service, [Account Administration only required for UIM Users and Account users not equal to the modified contact]</td>
</tr>
</tbody>
</table>

**Sample Request**
```
PUT /rest/contacts/13/password HTTP/1.1
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<contact>
  <password>newpassword</password>
</contact>
```

**Sample Reply**
HTTP/1.1 204 No Content

---

Sample Request (JSON)
```
PUT /rest/contacts/18 HTTP/1.1
Content-Type: application/json

{
  "accountId":"41",
  "contactId":"18",
  "creationDate":"2011-11-22T13:51:41.840+01:00",
  "email":chris@lala.com,
  "firstName":"Testfirstname",
  "language":"English",
  "lastName":"Testmodifiedlastname",
  "loginName":"testcontact",
  "password":"osPIf9dQcjkJ66Hml8pg=="
}
```

Sample Reply (JSON)
HTTP/1.1 204 No Content

---

Sample Reply (XML)
HTTP/1.1 204 No Content

---

Sample Request
```
PUT /rest/contacts/13/password HTTP/1.1
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<contact>
  <password>newpassword</password>
</contact>
```

**Sample Reply**
HTTP/1.1 204 No Content
### Delete an Existing Contact

<table>
<thead>
<tr>
<th><strong>URL</strong></th>
<th>/contacts/{contact_id}</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>DELETE</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>-</td>
</tr>
<tr>
<td><strong>Returns</strong></td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td><strong>Valid Users</strong></td>
<td>UIM Users</td>
</tr>
<tr>
<td><strong>Required Permissions</strong></td>
<td>Web Service, Account Administration</td>
</tr>
<tr>
<td><strong>Sample Request</strong></td>
<td>DELETE /rest/contacts/8 HTTP/1.1</td>
</tr>
<tr>
<td><strong>Sample Reply</strong></td>
<td>HTTP/1.1 204 No Content</td>
</tr>
</tbody>
</table>

### UIM Infrastructure Related Calls

#### Get List of Hubs

<table>
<thead>
<tr>
<th><strong>URL</strong></th>
<th>/hubs (optional query parameters: ?maxrows=2&amp;offset=5)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>GET</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>-</td>
</tr>
<tr>
<td><strong>Returns</strong></td>
<td>200 OK – HubList structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td><strong>Valid Users</strong></td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td><strong>Required Permissions</strong></td>
<td>Web Service</td>
</tr>
<tr>
<td><strong>Sample Request</strong></td>
<td>GET /rest/hubs HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>
| **Sample Reply** (XML) | HTTP/1.1 200 OK Content-Type: application/xml
<hublist>
  <hub>
    <address>chris-dev/primaryhub/win-cxp6lpt7v6g</address>
    <ip>192.168.211.129</ip>
    <name>primaryhub</name>
  </hub>
</hublist>
<table>
<thead>
<tr>
<th>Sample Reply (JSON)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP/1.1 200 OK</td>
</tr>
<tr>
<td>Content-Type: application/json</td>
</tr>
<tr>
<td>{</td>
</tr>
<tr>
<td>&quot;hub&quot;:{</td>
</tr>
<tr>
<td>&quot;address&quot;:/chris-dev/primaryhub/win-cxp6lpt7v6g&quot;,</td>
</tr>
<tr>
<td>&quot;ip&quot;:&quot;192.168.211.129&quot;,</td>
</tr>
<tr>
<td>&quot;name&quot;:&quot;primaryhub&quot;,</td>
</tr>
<tr>
<td>&quot;port&quot;:&quot;48002&quot;,</td>
</tr>
<tr>
<td>&quot;robotName&quot;:&quot;win-cxp6lpt7v6g&quot;,</td>
</tr>
<tr>
<td>&quot;status&quot;:&quot;0&quot;,</td>
</tr>
<tr>
<td>&quot;version&quot;:&quot;5.44 [Apr 26 2011]&quot;</td>
</tr>
<tr>
<td>}</td>
</tr>
<tr>
<td>}</td>
</tr>
</tbody>
</table>
## Get List of Robots

<table>
<thead>
<tr>
<th>URL</th>
<th>/hubs/{domain}/{hubname}/robots (optional query parameters: ?maxrows=2&amp;offset=5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – RobotList structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/hubs/chris-dev/primaryhub/robots HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>

**Sample Reply (XML)**

```
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<robotlist>
  <robot>
    <address>/chris-dev/primaryhub/nb-1538</address>
    <ip>192.168.211.1</ip>
    <name>nb-1538</name>
    <status>0</status>
    <version>5.32 Mar 21 2011</version>
  </robot>
  <robot>
    <address>/chris-dev/primaryhub/win-cxp6lpt7v6g</address>
    <ip>192.168.211.129</ip>
    <name>win-cxp6lpt7v6g</name>
    <status>0</status>
    <version>5.32 Apr 26 2011</version>
  </robot>
</robotlist>
```

**Sample Reply (JSON)**

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "robot": [
    {
      "address":"/chris-dev/primaryhub/nb-1538",
      "ip":"192.168.211.1",
      "name":"nb-1538",
      "status":"0",
      "version":"5.32 Mar 21 2011"
    },
    {
      "address":"/chris-dev/primaryhub/win-cxp6lpt7v6g",
      "ip":"192.168.211.129",
      "name":"win-cxp6lpt7v6g",
      "status":"0",
      "version":"5.32 Apr 26 2011"
    }
  ]
}
```
Get Robot Details

URL  /hubs/{domain}/{hub}/{robot}
Method  GET
Input  -
Returns  200 OK – Robot structure  
          401 Unauthorized  
          404 Not Found
Valid Users  UIM Users, Account Users
Required Permissions  Web Service
Sample Request  

Sample Reply (XML)

GET /rest/hubs/chris-dev/primaryhub/win-cxp6lpt7v6g  HTTP/1.1
Accept: application/xml

HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<robot>
  <address>/chris-dev/primaryhub/win-cxp6lpt7v6g</address>
  <ip>192.168.211.129</ip>
  <name>win-cxp6lpt7v6g</name>
  <status>0</status>
  <version>5.32 Apr 26 2011</version>
  <probes>
    <active>true</active>
    <address>/chris-dev/primaryhub/win-cxp6lpt7v6g/controller</address>
    <command>controller.exe</command>
    <config>robot.cfg</config>
    <description>Robot process and port controller</description>
    <group>Infrastructure</group>
    <ip>192.168.211.129</ip>
    <logfile>controller.log</logfile>
    <name>controller</name>
    <packageName>robot_update</packageName>
    <packageVersion>5.32</packageVersion>
    <pid>1620</pid>
    <port>48000</port>
    <timesActivated>0</timesActivated>
    <timesStarted>1</timesStarted>
    <timespec></timespec>
    <type>0</type>
    <workdir>robot</workdir>
  </probes>
</robot>
Sample Reply (JSON)

HTTP/1.1 200 OK
Content-Type: application/json

```json
{
    "address": "/chris-dev/primaryhub/win-cxp6lpt7v6g",
    "ip": "192.168.211.129",
    "name": "win-cxp6lpt7v6g",
    "probes": [
        {
            "active": "true",
            "address": "/chris-dev/primaryhub/win-cxp6lpt7v6g/controller",
            "command": "controller.exe",
            "config": "robot.cfg",
            "description": "Robot process and port controller",
            "group": "Infrastructure",
            "ip": "192.168.211.129",
            "logfile": "controller.log",
            "name": "controller",
            "packageName": "robot_update",
            "packageVersion": "5.32",
            "pid": "1680",
            "port": "48000",
            "timesActivated": "0",
            "timesStarted": "1",
            "timespec": "",
            "type": "0",
            "workdir": "robot"
        },
        {
            "active": "true",
            "address": "/chris-dev/primaryhub/win-cxp6lpt7v6g/hub",
            "arguments": "",
            ...
        }
    ],
    "status": "0",
    "version": "5.32 Apr 26 2011"
}
```

Get List of Archive Packages

<table>
<thead>
<tr>
<th>URL</th>
<th>/archive/list</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – ArchiveList structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required</td>
<td>Web Service, Archive Management</td>
</tr>
</tbody>
</table>
| Sample Request | GET /rest/archive/list  HTTP/1.1  
|               | Accept: application/xml |
| Sample Reply (XML) | HTTP/1.1 200 OK  
|               | Content-Type: application/xml  
|               | <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
|               | <archive-items>  
|               |   <archive-item>  
|               |     <build>1</build>  
|               |     <contains>0</contains>  
|               |     <date>26.04.2011</date>  
|               |     <description>Automatic Configuration Engine</description>  
|               |     <fileName>ace.zip</fileName>  
|               |     <group>Service</group>  
|               |     <name>ace</name>  
|               |     <version>2.18</version>  
|               |   </archive-item>  
|               |   <archive-item>  
|               |     <build>14</build>  
|               |     <contains>0</contains>  
|               |     <date>13.10.2010</date>  
|               |     <description>Alarm augmentation and routing component</description>  
|               |     <fileName>alarm_augmentation_3.11.zip</fileName>  
|               |     <group>Application</group>  
|               |     <name>alarm_augmentation</name>  
|               |     <version>3.11</version>  
|               |   </archive-item>  
|               | [...  
|               | </archive-items>  
| Sample Reply (JSON) | HTTP/1.1 200 OK  
|               | Content-Type: application/json  
|               | {  
|               |     "archive-item":  
|               |       {  
|               |         "build":"1",  
|               |         "contains":"0",  
|               |         "date":"26.04.2011",  
|               |         "description":"Automatic Configuration Engine",  
|               |         "fileName":"ace.zip",  
|               |         "group":"Service",  
|               |         "name":"ace",  
|               |         "version":"2.18"  
|               |       },  
|               |       {  
|               |         "build":"14",  
|               |         "contains":"0",  
|               |         "date":"13.10.2010",  
|               |         "description":"Alarm augmentation and routing component",  
|               |         "fileName":"alarm_augmentation_3.11.zip",  
|               |         "group":"Application",  
|               |         "name":"alarm_augmentation",  
|               |         "version":"3.11"  
|               |       },  
|               |       ...  
|               |     }  
|               | }  
|               | }  

Dashboard Related Calls

Note: These calls allow you to retrieve the deprecated Enterprise Console-type dashboards and do not cover the Custom Dashboards from UMP.

### Get Dashboards

<table>
<thead>
<tr>
<th>URL</th>
<th>/dashboards</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – DashboardList structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

**Sample Request**

GET /dashboards HTTP/1.1
Accept: application/xml

**Sample Reply (XML)**

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dashboards>
  <dashboard>Example</dashboard>
  <dashboard>MSP-ISP</dashboard>
</dashboards>
```

**Sample Reply (JSON)**

```json
HTTP/1.1 200 OK
Content-Type: application/json

{  
  "dashboard": [  
    "Example",  
    "MSP-ISP"  
  ]
}
```

### Get Dashboard State

<table>
<thead>
<tr>
<th>URL</th>
<th>/dashboards/{dashboard-name}/state</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – AlarmSeverity structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/dashboards/Example/state HTTP/1.1 Accept: application/xml</td>
</tr>
<tr>
<td>Sample Reply</td>
<td>HTTP/1.1 200 OK Content-Type: application/xml 2</td>
</tr>
</tbody>
</table>
ConfigurationItem (CI) Data Retrieval Calls

Get CI Definitions

This call returns a list of all CI definitions. This call is not limited to definitions delivered by components belonging to the account of an account user.

<table>
<thead>
<tr>
<th>URL</th>
<th>/ci/definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – CIDefinitionList structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/ci/definitions HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>

Sample Reply (XML)

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ci_definition_list>
  <ci_definition>
    <type>1</type>
    <description>System</description>
    <parent xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/>
    <children_list>
      <ci_definition>
        <type>1.1</type>
        <description>System.Disk</description>
        <parent>1</parent>
        <children_list>
          <ci_definition>
            <type>1.1.1</type>
            <description>System.Disk.Filesystem</description>
            <parent>1.1</parent>
            <children_list/>
          </ci_definition>
          <ci_definition>
            <type>1.1.2</type>
            <description>System.Disk.Partition</description>
            <parent>1.1</parent>
            <children_list/>
          </ci_definition>
        </children_list>
      </ci_definition>
      <ci_definition>
        <type>1.10</type>
        <description>System.File</description>
        <parent>1</parent>
      </ci_definition>
    </children_list>
  </ci_definition>
</ci_definition_list>
```
<children_list>
  <ci_definition>
    <type>1.10.1</type>
    <description>System.File.XML</description>
    <parent>1.10</parent>
    <children_list />
  </ci_definition>
  ... (more exists, but not shown)
</ci_definition>
</children_list>

HTTP/1.1 200 OK
Content-Type: application/json
{
  "ci_definition": [
    {
      "type": "1",
      "description": "System",
      "parent": {
        "@nil": "true"
      },
      "children_list": [
        {
          "type": "1.1",
          "description": "System.Disk",
          "parent": "1",
          "children_list": [
            {
              "type": "1.1.1",
              "description": "System.Disk.Filesystem",
              "parent": "1.1",
              "children_list": null
            },
            {
              "type": "1.1.2",
              "description": "System.Disk.Partition",
              "parent": "1.1",
              "children_list": null
            }
          ]
        },
        {
          "type": "1.10",
          "description": "System.File",
          "parent": "1",
          "children_list": [
            {
              "type": "1.10.1",
              "description": "System.File.XML",
              "parent": "1.10",
              "children_list": null
            }
          ]
        }
      ]
    }
  ]
}
**Get CI Definition**

This call returns an individual CI definition for a given ci_type. This call is not limited to definitions delivered by components belonging to the account of an account user. The ci_type can be determined by retrieving all definitions and looking for the correct match.

<table>
<thead>
<tr>
<th>URL</th>
<th>/ci/definition/{ci_type}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – CI Definition structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/ci/definitions/3.24 HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>
| Sample Reply (XML) | HTTP/1.1 200 OK Content-Type: application/xml  
  <?xml version="1.0" encoding="utf-8" standalone="yes"?>  
  <ci_definition>  
    <type>3.24</type>  
    <description>Application.OCS</description>  
    <parent>3</parent>  
    <children_list>  
      <ci_definition>  
        <type>3.24.1</type>  
        <description>Application.OCS.EventLogs</description>  
        <parent>3.24</parent>  
        <children_list />  
      </ci_definition>  
      <ci_definition>  
        <type>3.24.2</type>  
        <description>Application.OCS.Files</description>  
        <parent>3.24</parent>  
        <children_list />  
      </ci_definition>  
      <ci_definition>  
        <type>3.24.3</type>  
        <description>Application.OCS.Filesystem</description>  
        <parent>3.24</parent>  
        <children_list />  
      </ci_definition>  
      <ci_definition>  
        <type>3.24.4</type>  
        <description>Application.OCS.PerformanceCounters</description>  
        <parent>3.24</parent>  
        <children_list>  
          <ci_definition>  
            <type>3.24.4.1</type>  
            <description>Application.OCS.PerformanceCounters.DynamicCounters</description>  
          </ci_definition>  
        </children_list>  
      </ci_definition>  
    </children_list>  
  </ci_definition>  

### Sample Reply (JSON)

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "type": "3.24",
  "description": "Application.OCS",
  "parent": "3",
  "children_list": {
    "ci_definition": [
      {
        "type": "3.24.1",
        "description": "Application.OCS.EventLogs",
        "parent": "3.24",
        "children_list": null
      }
    ]
  }
}
```

---

```xml
<parent>3.24.5</parent>
<description>Application.OCS.Processes</description>
<parent>3.24</parent>
<children_list />
<ci_definition>
  <type>3.24.6</type>
  <description>Application.OCS.Services</description>
  <parent>3.24</parent>
  <children_list />
</ci_definition>
<ci_definition>
  <type>3.24.7</type>
  <description>Application.OCS.WMI</description>
  <parent>3.24</parent>
  <children_list>
    <ci_definition>
      <type>3.24.7.1</type>
      <description>Application.OCS.WMI.DynamicCounters</description>
      <parent>3.24.7</parent>
      <children_list />
    </ci_definition>
  </children_list>
</ci_definition>
```
Get CI Metric Definitions
This call returns all CI metric definitions. This call is not limited to definitions delivered by components belonging to the account of an account user.

<table>
<thead>
<tr>
<th>URL</th>
<th>/ci/metricdefinitions/</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – CIMetricDefinitionList structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>
Sample Request
GET /rest/ci/metricdefinitions/ HTTP/1.1
Accept: application/xml

Sample Reply (XML)
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<ci_metric_definition_list>
  <ci_metric_definition>
    <metric_type>1.1.2:1</metric_type>
    <description>% CPU Time</description>
    <unit_type>User Defined</unit_type>
    <ci_type>1.1.2</ci_type>
  </ci_metric_definition>
  <ci_metric_definition>
    <metric_type>1.1.2:10</metric_type>
    <description>Total Mbytes</description>
    <unit_type>User Defined</unit_type>
    <ci_type>1.1.2</ci_type>
  </ci_metric_definition>
  ...
</ci_metric_definition_list>

Sample Reply (JSON)
HTTP/1.1 200 OK
Content-Type: application/json

{ "ci_metric_definition": [ { "metric_type":"1.1.2:1", "description":"% CPU Time", "unit_type":"User Defined", "ci_type":"1.1.2" }, { "metric_type":"1.1.2:10", "description":"Total Mbytes", "unit_type":"User Defined", "ci_type":"1.1.2" }, ...
] } (more definitions exist)

Get CI Metric Definition
This call returns an individual CI metric definition for a given metric_type. This call is not limited to definitions delivered by components belonging to the account of an account user. A metric_type can be determined by listing all metric definitions and searching for the one of interest.

URL /ci/metricdefinitions/{metric_type}
Method GET
Input -
Returns 200 OK – CIMetricDefinition structure
401 Unauthorized
404 Not Found
Valid Users UIM Users, Account Users
### Sample Request

GET /rest/ci/metricdefinitions/1.10:21 HTTP/1.1
Accept: application/xml

### Sample Reply (XML)

HTTP/1.1 200 OK
Content-Type: application/xml

```xml
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<ci_metric_definition>
  <metric_type>1.10:21</metric_type>
  <description>File Size In GB</description>
  <unit_type>GB</unit_type>
  <ci_type>1.10</ci_type>
</ci_metric_definition>
```

### Sample Reply (JSON)

HTTP/1.1 200 OK
Content-Type: application/json

```json
{
  "metric_type":"1.10:21",
  "description":"File Size In GB",
  "unit_type":"GB",
  "ci_type":"1.10"
}
```

### Get CI Metric

This call returns an individual CI metric for a given metric_id. This call is not limited to definitions delivered by components belonging to the account of an account user.

<table>
<thead>
<tr>
<th>URL</th>
<th>/ci/metrics/{metric_id}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – CIMetric structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

### Sample Request

GET /rest/ci/metrics/M031B035E62A457DA2F674DD723D926F3 HTTP/1.1
Accept: application/xml

### Sample Reply (XML)

HTTP/1.1 200 OK
Content-Type: application/xml

```xml
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<ci_metric>
  <metric_id>M031B035E62A457DA2F674DD723D926F3</metric_id>
  <ci_id>C4FDD4CE43A2F891815F7129E195ACBC1</ci_id>
  <metric_type>1.5:1</metric_type>
</ci_metric>
```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "metric_id":"M031B035E62A4S7DA2F674DD723D926F3",
  "ci_id":"C4FDD4CE3A2F891815F7129E195ACBC1",
  "metric_type":"1.5:1"
}
## QoS Data Retrieval Calls

### Get QoS Definitions

This call returns a list of all QoS definitions. This call is not limited to definitions delivered by components belonging to the account of an account user.

<table>
<thead>
<tr>
<th>URL</th>
<th>/qos/definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – QoSDefinitionList structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

#### Sample Request

```
GET /rest/qos/definitions HTTP/1.1
Accept: application/xml
```

#### Sample Reply (XML)

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<qos-definitions>
  <qos-definition>
    <bool>false</bool>
    <description>CPU Usage</description>
    <hasMax>true</hasMax>
    <name>QOS_CPU_USAGE</name>
    <qosDefId>14</qosDefId>
    <qosGroup>QOS_MACHINE</qosGroup>
    <type>0</type>
    <unit>Percent</unit>
    <unitShort>%</unitShort>
  </qos-definition>
  <qos-definition>
    <bool>false</bool>
    <description>Disk Usage</description>
    <hasMax>true</hasMax>
    <name>QOS_DISK_USAGE</name>
    <qosDefId>12</qosDefId>
    <qosGroup>QOS_MACHINE</qosGroup>
    <type>0</type>
    <unit>Megabytes</unit>
    <unitShort>MB</unitShort>
  </qos-definition>
</qos-definitions>
```

#### Sample Reply (JSON)

```json
HTTP/1.1 200 OK
Content-Type: application/json
{
  "qos-definition": [
    {
      // ...
    },
    {
      // ...
    }
  ]
}
```
Get a QoS Definition by Name

This call returns a QoS definition for a given QoS name. With the QoS-Definition, all source/target combinations are returned. For Account users, only source/target combinations of the matching origin are displayed.

<table>
<thead>
<tr>
<th>URL</th>
<th>/qos/definitions/{qos-name}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – QoSDefinition structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/qos/definitions/QOS_CPU_USAGE HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>
| Sample Reply (XML)| HTTP/1.1 200 OK
|                   | Content-Type: application/xml |
|                   | <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
|                   | <qos-definition>
|                   |   <bool>false</bool>
|                   |   <description>CPU Usage</description>
|                   |   <hasMax>true</hasMax>
|                   |   <name>QOS_CPU_USAGE</name>
|                   |   <qosDefId>14</qosDefId>
<qosGroup>QOS_MACHINE</qosGroup>
<type>0</type>
<unit>Percent</unit>
<unitShort>%</unitShort>
<sourceTargets>
    <origin>primaryhub</origin>
    <source>WIN-CXP6LPT7V6G</source>
    <target>WIN-CXP6LPT7V6G</target>
</sourceTargets>
<sourceTargets>
    <origin>primaryhub</origin>
    <source>WIN-CXP6LPT7V6G</source>
    <target>User</target>
</sourceTargets>
<sourceTargets>
    <origin>primaryhub</origin>
    <source>WIN-CXP6LPT7V6G</source>
    <target>System</target>
</sourceTargets>
<sourceTargets>
    <origin>primaryhub</origin>
    <source>WIN-CXP6LPT7V6G</source>
    <target>Wait</target>
</sourceTargets>
</qos-definition>

HTTP/1.1 200 OK
Content-Type: application/json

{
    "bool":"false",
    "description":"CPU Usage",
    "hasMax":"true",
    "name":"QOS_CPU_USAGE",
    "qosDefId":"14",
    "qosGroup":"QOS_MACHINE",
    "type":"0",
    "unit":"Percent",
    "unitShort":"%",
    "sourceTargets":[
        {
            "origin":"primaryhub",
            "source":"WIN-CXP6LPT7V6G",
            "target":"WIN-CXP6LPT7V6G"
        },
        {
            "origin":"primaryhub",
            "source":"WIN-CXP6LPT7V6G",
            "target":"User"
        },
        {
            "origin":"primaryhub",
            "source":"WIN-CXP6LPT7V6G",
            "target":"System"
        },
        {
            "origin":"primaryhub",
            "source":"WIN-CXP6LPT7V6G",
            "target":"Wait"
        },
        {
            "origin":"primaryhub",
            "source":"WIN-CXP6LPT7V6G",
            "target":"Idle"
        }
    ]
}
### Get Sources for QoS-Name

<table>
<thead>
<tr>
<th><strong>URL</strong></th>
<th>/qos/sources/{qos-name}</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>GET</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>-</td>
</tr>
<tr>
<td><strong>Returns</strong></td>
<td>200 OK – QoS Source List</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td><strong>Valid Users</strong></td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td><strong>Required Permissions</strong></td>
<td>Web Service</td>
</tr>
<tr>
<td><strong>Sample Request</strong></td>
<td>GET /rest/qos/sources/QOS_CPU_USAGE HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>
| **Sample Reply (XML)** | HTTP/1.1 200 OK Content-Type: application/xml  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<qos-sources>  
  <qos-source>  
    <origin>primaryhub</origin>  
    <source>192.168.0.1</source>  
  </qos-source>  
  <qos-source>  
    <origin>primaryhub</origin>  
    <source>192.168.0.120</source>  
  </qos-source>  
  <qos-source>  
    <origin>primaryhub</origin>  
    <source>WIN-CXP6LPT7V6G</source>  
  </qos-source>  
</qos-sources> |
| **Sample Reply (JSON)** | HTTP/1.1 200 OK Content-Type: application/json  
{
  "qos-source":{
    "origin":"primaryhub",
    "source":"WIN-CXP6LPT7V6G"
  }
} |

### Get Targets for QoS-Name and Source

Returns a list of all targets for this combination of QoS-Name and Source.

<table>
<thead>
<tr>
<th><strong>URL</strong></th>
<th>/qos/targets/{qos-name}/{source-name}</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>GET</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>-</td>
</tr>
<tr>
<td><strong>Returns</strong></td>
<td>200 OK – QoS Target List</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/qos/targets/QOS_CPU_USAGE/WIN-CXP6LPT7V6G/primaryhub HTTP/1.1 Accept: application/xml</td>
</tr>
<tr>
<td>Sample Reply (XML)</td>
<td>HTTP/1.1 200 OK</td>
</tr>
<tr>
<td></td>
<td>&lt;?xml version=&quot;1.0&quot; encoding=&quot;UTF-8&quot; standalone=&quot;yes&quot;?&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;targets&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;target&gt;Idle&lt;/target&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;target&gt;System&lt;/target&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;target&gt;User&lt;/target&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;target&gt;Wait&lt;/target&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;target&gt;WIN-CXP6LPT7V6G&lt;/target&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;/targets&gt;</td>
</tr>
<tr>
<td>Sample Reply (JSON)</td>
<td>HTTP/1.1 200 OK</td>
</tr>
<tr>
<td></td>
<td>Content-Type: application/json</td>
</tr>
<tr>
<td></td>
<td>{</td>
</tr>
<tr>
<td></td>
<td>&quot;target&quot;:[</td>
</tr>
<tr>
<td></td>
<td>&quot;Idle&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;System&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;User&quot;,</td>
</tr>
<tr>
<td></td>
<td>&quot;Wait&quot;,</td>
</tr>
</tbody>
</table>
|                             |     "WIN-CXP6LPT7V6G"
|                             | ]                     |
| Get Targets for QoS-Name and Source for a Given Origin |
| URL                         | /qos/targets/{qos-name}/{source-name}/{origin} |
| Method                      | GET                     |
| Input                       | -                       |
| Returns                     | 200 OK – QoS Target List |
|                             | 401 Unauthorized        |
|                             | 404 Not Found           |
| Valid Users                 | UIM Users, Account Users |
| Required Permissions        | Web Service             |
| Sample Request              | GET /rest/qos/targets/QOS_CPU_USAGE/WIN-CXP6LPT7V6G HTTP/1.1 Accept: application/xml |
| Sample Reply (XML)          | HTTP/1.1 200 OK         |
|                             | <?xml version="1.0" encoding="UTF-8" standalone="yes"?> |
|                             | <targets>               |
Get Raw QoS Data

This call returns “raw” QoS data. In UIM, “raw” data will return all data points that are stored for that combination of QoS Name, source and target. The alternative is “historical” = aggregated data. To retrieve aggregated data, see the call “Get historical QoS Data” (or simply append a “/historical” to this call).

URL

/qos/data /name/{qosname}/{source}/{target}/{from}/{to}/{maxrows}

Parameter explanation:

- “qosname” is the name of a QoS definition, e.g. “QOS_CPU_USAGE”.
- “source” is the name of a valid QoS source for that QoS definition.
- “target” is a valid target-value for the combination of QoS Name and Source.
- “from” should be a date in the format yyyyddMMHHmm (e.g. 201107131200) or one of the following keywords: lasthour, lastday, lastweek, lastmonth.
- “to” should be a date in the format yyyyddMMHHmm (e.g. 201111100938) or the keyword “now” which will be resolved to the current date and time on the server.
- “maxrows” indicates the maximum number of datapoints to return (0 = unlimited).

Method

GET

Input

-

Returns

200 OK – QoS Data List

401 Unauthorized

404 Not Found

Valid Users

UIM Users, Account Users

Required Permissions

Web Service

Sample Request

GET /rest/qos/data/name/QOS_CPU_USAGE/WIN-CXP6LPT7V6G/WIN-CXP6LPT7V6G/201101010000/20111111257/20 HTTP/1.1
Accept: application/xml

Sample Reply (XML)

HTTP/1.1 200 OK
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

Sample Reply (JSON)

HTTP/1.1 200 OK
Content-Type: application/json

{  
  "target": [ 
    "Idle",
    "System",
    "User",
    "Wait",
    "WIN-CXP6LPT7V6G"
  ]
}"
<qos-data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>30.09</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>62.2</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>71.04</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>53.72</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
</qos-data>

HTTP/1.1 200 OK
Content-Type: application/json

{
  "data": [
    {
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "54.65",
      "tableid": "1",
      "tz_offset": "-3600"
    },
    {
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "75.13",
      "tableid": "1",
      "tz_offset": "-3600"
    },
    {
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "92.49",
      "tableid": "1",
      "tz_offset": "-3600"
    },
    {
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "85.23",
      "tableid": "1",
      "tz_offset": "-3600"
    }
  ]
}
### Get QoS Data Using MetricId

URL

```
/qos/data /metricid/{ci_metric_id}/[from]/[to]/{maxrows}
```

Parameter explanation:
- **“ci_metric_id”** – a valid Configuration Item Metric Id. This ID can be retrieved from alarms.
  - “from” should be a date in the format yyyymmddHHmm (e.g. 201107131200) or one of the following keywords: lasthour, lastday, lastweek, lastmonth.
  - “to” should be a date in the format yyyymmddHHmm (e.g. 201111090938) or the keyword “now” which will be resolved to the current date and time on the server.
  - “maxrows” indicates the maximum number of datapoints to return (0 = unlimited).

**Method**

| GET |

**Input**

- |

**Returns**

<table>
<thead>
<tr>
<th>200 OK – QoS Data List</th>
</tr>
</thead>
<tbody>
<tr>
<td>401 Unauthorized</td>
</tr>
<tr>
<td>404 Not Found</td>
</tr>
</tbody>
</table>

**Valid Users**

UIM Users, Account Users

**Required Permissions**

Web Service

**Sample Request**

```
GET /rest/qos/data/metricid/MDF2DD98996F2EB3FCF3C60B4AC9A5FES/201101010000/201111111257/20 HTTP/1.1
Accept: application/xml
```

**Sample Reply (XML)**

```
<qos-data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>30.09</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>62.2</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>71.04</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>53.72</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
</qos-data>
```
Sample Reply (JSON)

HTTP/1.1 200 OK
Content-Type: application/json

```json
{
    "data": [
        {
            "sampletime": "2011-11-17T00:00:00+01:00",
            "samplevalue": "54.65",
            "tableid": "1",
            "tz_offset": "-3600"
        },
        {
            "sampletime": "2011-11-17T00:00:00+01:00",
            "samplevalue": "75.13",
            "tableid": "1",
            "tz_offset": "-3600"
        },
        {
            "sampletime": "2011-11-17T00:00:00+01:00",
            "samplevalue": "92.49",
            "tableid": "1",
            "tz_offset": "-3600"
        },
        {
            "sampletime": "2011-11-17T00:00:00+01:00",
            "samplevalue": "85.23",
            "tableid": "1",
            "tz_offset": "-3600"
        }
    ]
}
```

Get QoS Data Using TableId

URL

/qos/data /tableid/{table_id}/(from)/(to)/(maxrows)

Parameter explanation:

- “table_id” – The table_id identifying this QoS data series.
- “from” should be a date in the format yyyyddMMHHmm (e.g. 201107131200) or one of the following keywords: lasthour, lastday, lastweek, lastmonth.
- “to” should be a date in the format yyyyddMMHHmm (e.g. 2011111100938) or the keyword “now” which will be resolved to the current date and time on the server.
- “maxrows” indicates the maximum number of datapoints to return (0 = unlimited).

Method

GET

Input

-

Returns

200 OK – QoS Data List
401 Unauthorized
404 Not Found

Valid Users

UIM Users, Account Users

Required Permissions

Web Service

Sample Request

GET /rest/qos/data/tableid/1/201101010000/201111111257/20 HTTP/1.1
Accept: application/xml

Sample Reply

HTTP/1.1 200 OK
**XML**

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<qos-data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>30.09</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>62.2</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>71.04</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>53.72</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
</qos-data>
```

**Sample Reply (JSON)**

```
HTTP/1.1 200 OK
Content-Type: application/json

{
   "data": [
      {
         "sampletime": "2011-11-17T00:00:00+01:00",
         "samplevalue": "54.65",
         "tableid": "1",
         "tz_offset": "-3600"
      },
      {
         "sampletime": "2011-11-17T00:00:00+01:00",
         "samplevalue": "75.13",
         "tableid": "1",
         "tz_offset": "-3600"
      },
      {
         "sampletime": "2011-11-17T00:00:00+01:00",
         "samplevalue": "92.49",
         "tableid": "1",
         "tz_offset": "-3600"
      },
      {
         "sampletime": "2011-11-17T00:00:00+01:00",
         "samplevalue": "85.23",
         "tableid": "1",
         "tz_offset": "-3600"
      }
   ]
}
```
## Get Data Using QoS ConstraintId

<table>
<thead>
<tr>
<th>URL</th>
<th>/qos/data/constraint/{constraint_id}/{from}/{to}/{maxrows}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter explanation:</td>
<td></td>
</tr>
<tr>
<td>• “constraint_id” – A valid QoS constraint ID. Can be retrieved via the SLA/SLO calls.</td>
<td></td>
</tr>
<tr>
<td>• “from” should be a date in the format yyyyddMMHHmm (e.g. 201107131200) or one of the following keywords: lasthour, lastday, lastweek, lastmonth.</td>
<td></td>
</tr>
<tr>
<td>• “to” should be a date in the format yyyyddMMHHmm (e.g. 201111100938) or the keyword “now” which will be resolved to the current date and time on the server.</td>
<td></td>
</tr>
<tr>
<td>• “maxrows” indicates the maximum number of datapoints to return (0 = unlimited).</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – QoS Data List</td>
</tr>
<tr>
<td>401 Unauthorized</td>
<td></td>
</tr>
<tr>
<td>404 Not Found</td>
<td></td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

**Sample Request**

```
GET /rest/qos/data/constraint/14/201101010000/201111111257/20 HTTP/1.1
Accept: application/xml
```

**Sample Reply (XML)**

```
HTTP/1.1 200 OK

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
&qos-data>
<data>
  <sampletime>2011-11-11T00:00:00+01:00</sampletime>
  <samplevalue>30.09</samplevalue>
  <tableid>1</tableid>
  <tz_offset>-3600</tz_offset>
</data>
<data>
  <sampletime>2011-11-11T00:00:00+01:00</sampletime>
  <samplevalue>62.2</samplevalue>
  <tableid>1</tableid>
  <tz_offset>-3600</tz_offset>
</data>
<data>
  <sampletime>2011-11-11T00:00:00+01:00</sampletime>
  <samplevalue>71.04</samplevalue>
  <tableid>1</tableid>
  <tz_offset>-3600</tz_offset>
</data>
<data>
  <sampletime>2011-11-11T00:00:00+01:00</sampletime>
  <samplevalue>53.72</samplevalue>
  <tableid>1</tableid>
  <tz_offset>-3600</tz_offset>
</data>
</qos-data>
```

**Sample Reply**

HTTP/1.1 200 OK
### Get Historical (aggregated) QoS Data

This call returns “historical” QoS data. “Historical” data is aggregated data while “raw” data will return all data points that are stored for that combination of QoS Name, source and target.

**URL**

```
/qos/data /name/{qosname}/{source}/{target}/from/{to}/maxrows/historical
```

**Parameter explanation:**
- “qosname” is the name of a QoS definition, e.g. “QOS_CPU_USAGE”.
- “source” is the name of a valid QoS source for that QoS definition.
- “target” is a valid target-value for the combination of QoS Name and Source.
- “from” should be a date in the format yyyyddMMHHmm (e.g. 201107131200) or one of the following keywords: lasthour, lastday, lastweek, lastmonth.
- “to” should be a date in the format yyyyddMMHHmm (e.g. 201111100938) or the keyword “now” which will be resolved to the current date and time on the server.
- “maxrows” indicates the maximum number of datapoints to return (0 = unlimited).

**Method**

GET

**Input**

- 

**Returns**

- 200 OK – QoS Data List
- 401 Unauthorized
- 404 Not Found

**Valid Users**

UIM Users, Account Users

**Required Permissions**

Web Service
### Sample Request

```
GET /rest/qos/data/name/QOS_CPU_USAGE/WIN-CXP6LPT7V6G/WIN-CXP6LPT7V6G/201101010000/20111111257/20/historical HTTP/1.1
Accept: application/xml
```

### Sample Reply (XML)

```xml
HTTP/1.1 200 OK
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<qos-data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>30.09</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>62.2</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>71.04</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>53.72</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
</qos-data>
```

### Sample Reply (JSON)

```json
HTTP/1.1 200 OK
Content-Type: application/json
{
  "data": [
    {
      "sampletime":"2011-11-17T00:00:00+01:00",
      "samplevalue":"54.65",
      "tableid":"1",
      "tz_offset":"-3600"
    },
    {
      "sampletime":"2011-11-17T00:00:00+01:00",
      "samplevalue":"75.13",
      "tableid":"1",
      "tz_offset":"-3600"
    },
    {
      "sampletime":"2011-11-17T00:00:00+01:00",
      "samplevalue":"92.49",
      "tableid":"1",
      "tz_offset":"-3600"
    },
    {
      "sampletime":"2011-11-17T00:00:00+01:00",
      "samplevalue":"85.23",
      "tableid":"1",
      "tz_offset":"-3600"
    }
  ]
}
```
Get Historical QoS Data Using MetricId

URL

/\qos/data /metricid/[ci\_metric\_id]/[/from]/[/to]/[/maxrows]/historical

Parameter explanation:
- “ci\_metric\_id” – a valid Configuration Item Metric Id. This ID can be retrieved from alarms.
- “from” should be a date in the format yyyydddMMHHmm (e.g. 201107131200) or one of the following keywords: lasthour, lastday, lastweek, lastmonth.
- “to” should be a date in the format yyyyydddMMHHmm (e.g. 201111090909) or the keyword “now” which will be resolved to the current date and time on the server.
- “maxrows” indicates the maximum number of datapoints to return (0 = unlimited).

Method
GET

Input
-

Returns
200 OK – QoS Data List
401 Unauthorized
404 Not Found

Valid Users
UIM Users, Account Users

Required Permissions
Web Service

Sample Request
GET /rest/qos/data/metricid/MDF2DD98996F2EB3FCF3C60B4AC9A5FES\201101010000\201111111257\20/historical
Accept: application/xml

Sample Reply (XML)
HTTP/1.1 200 OK
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<qos-data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>30.09</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>62.2</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>71.04</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>53.72</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
</qos-data>
Get Historical QoS Data Using TableId

URL
/qos/data /tableid/\{table_id\}/\{from\}/\{to\}/\{maxrows\}\historical

Parameter explanation:
- “table_id” – The table_id identifying this QoS data series.
  “from” should be a date in the format yyyyddMMHHmm (e.g. 201107131200) or one of the following keywords: lasthour, lastday, lastweek, lastmonth.
- “to” should be a date in the format yyyyddMMHHmm (e.g. 201111100938) or the keyword “now” which will be resolved to the current date and time on the server.
- “maxrows” indicates the maximum number of datapoints to return (0 = unlimited).

Method
GET

Input
-

Returns
200 OK – QoS Data List
401 Unauthorized
404 Not Found

Valid Users
UIM Users, Account Users

Required Permissions
Web Service

Sample Request
GET /rest/qos/data/tableid/1/201101010000/201111111257/20/historical HTTP/1.1
Accept: application/xml

Sample Reply
HTTP/1.1 200 OK

Sample Reply (JSON)
HTTP/1.1 200 OK
Content-Type: application/json

```json
{
    "data": [  
        {
            "sampletime": "2011-11-17T00:00:00+01:00",
            "samplevalue": "54.65",
            "tableid": "1",
            "tz_offset": "-3600"
        },
        {
            "sampletime": "2011-11-17T00:00:00+01:00",
            "samplevalue": "75.13",
            "tableid": "1",
            "tz_offset": "-3600"
        },
        {
            "sampletime": "2011-11-17T00:00:00+01:00",
            "samplevalue": "92.49",
            "tableid": "1",
            "tz_offset": "-3600"
        },
        {
            "sampletime": "2011-11-17T00:00:00+01:00",
            "samplevalue": "85.23",
            "tableid": "1",
            "tz_offset": "-3600"
        }
    ]
}
```
<xml version="1.0" encoding="UTF-8" standalone="yes">
<qos-data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>30.09</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>62.2</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>71.04</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  <data>
    <sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>53.72</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
</qos-data>

HTTP/1.1 200 OK
Content-Type: application/json

{  
  "data": [  
    {  
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "54.65",
      "tableid": "1",
      "tz_offset": "-3600"
    },  
    {  
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "75.13",
      "tableid": "1",
      "tz_offset": "-3600"
    },  
    {  
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "92.49",
      "tableid": "1",
      "tz_offset": "-3600"
    },  
    {  
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "85.23",
      "tableid": "1",
      "tz_offset": "-3600"
    }
  ]
}
### Get Historical Data Using a QoS Constraint ID

- **URL**
  
  /qos/data/constraintid/{constraint_id}/(from)/(to)/(maxrows)/historical

  Parameter explanation:
  
  - "constraint_id" – A valid QoS constraint ID. Can be retrieved via the SLA/SLO calls.
  - "from" should be a date in the format yyyyddMMHHmm (e.g. 201107131200) or one of the following keywords: lasthour, lastday, lastweek, lastmonth.
  - "to" should be a date in the format yyyyddMMHHmm (e.g. 201111110000) or the keyword “now” which will be resolved to the current date and time on the server.
  - "maxrows" indicates the maximum number of datapoints to return (0 = unlimited).

- **Method**
  
  GET

- **Input**
  
  -

- **Returns**
  
  200 OK – QoS Data List
  401 Unauthorized
  404 Not Found

- **Valid Users**
  
  UIM Users, Account Users

- **Required Permissions**
  
  Web Service

- **Sample Request**
  
  GET /rest/qos/data/constraintid/14/201101010000/201111111257/20/historical HTTP/1.1
  
  Accept: application/xml

- **Sample Reply (XML)**
  
  HTTP/1.1 200 OK

  ```xml
  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <qos-data>
    <data><sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>30.09</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
    <data><sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>62.2</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
    <data><sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>71.04</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
    <data><sampletime>2011-11-11T00:00:00+01:00</sampletime>
    <samplevalue>53.72</samplevalue>
    <tableid>1</tableid>
    <tz_offset>-3600</tz_offset>
  </data>
  </qos-data>
  ```

- **Sample Reply**
  
  HTTP/1.1 200 OK
Content-Type: application/json

```json
{
  "data": [
    {
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "54.65",
      "tableid": "1",
      "tz_offset": "-3600"
    },
    {
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "75.13",
      "tableid": "1",
      "tz_offset": "-3600"
    },
    {
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "92.49",
      "tableid": "1",
      "tz_offset": "-3600"
    },
    {
      "sampletime": "2011-11-17T00:00:00+01:00",
      "samplevalue": "85.23",
      "tableid": "1",
      "tz_offset": "-3600"
    }
  ]
}
```
### SLA Related Calls

#### Get All SLADefinitions

This call returns a list of all defined SLAs visible to the user invoking the call.

<table>
<thead>
<tr>
<th>URL</th>
<th>/sla/definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – SLA List</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/sla/definitions HTTP/1.1</td>
</tr>
<tr>
<td>Sample Reply</td>
<td>HTTP/1.1 200 OK</td>
</tr>
</tbody>
</table>
|            | <xml version="1.0" encoding="UTF-8" standalone="yes"?>
|            | <slas>
|            |     <sla>
|            |         <complianceAlarm>false</complianceAlarm>
|            |         <compliancePercentage>100</compliancePercentage>
|            |         <complianceWarning>false</complianceWarning>
|            |         <complianceWarningLevel>100</complianceWarningLevel>
|            |         <description></description>
|            |         <name>TestSLA</name>
|            |         <slaId>1</slaId>
|            |         <periodBegin>2011-12-01T00:00:00+01:00</periodBegin>
|            |         <periodCode>m</periodCode>
|            |         <periodEnd>2012-01-01T00:00:00+01:00</periodEnd>
|            |         <periodNumber>1</periodNumber>
|            |         <periodStart>2011-11-01T00:00:00+02:00</periodStart>
|            |     </sla>
|            |     <sla>
|            |         <complianceAlarm>false</complianceAlarm>
|            |         <compliancePercentage>100</compliancePercentage>
|            |         <complianceWarning>false</complianceWarning>
|            |         <complianceWarningLevel>100</complianceWarningLevel>
|            |         <description></description>
|            |         <name>AccountSLA</name>
|            |         <slaId>2</slaId>
|            |         <periodBegin>2011-12-01T00:00:00+01:00</periodBegin>
|            |         <periodCode>m</periodCode>
|            |         <periodEnd>2012-01-01T00:00:00+01:00</periodEnd>
|            |         <periodNumber>1</periodNumber>
|            |         <periodStart>2011-11-01T00:00:00+01:00</periodStart>
|            |     </sla>
|            | </slas>                       |
| Sample Reply | HTTP/1.1 200 OK                |
### Get SLA Definition

**URL**
/sla/{sla-id}/definition

**Method**
GET

**Input**
-

**Returns**
200 OK – SLA
401 Unauthorized
404 Not Found

**Valid Users**
UIM Users, Account Users

**Required Permissions**
Web Service

**Sample Request**
GET /rest/sla/1/definition HTTP/1.1
Accept: application/xml

**Sample Reply (XML)**
HTTP/1.1 200 OK
Content-Type: application/xml

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<sla>
  <complianceAlarm>false</complianceAlarm>
  <compliancePercentage>100</compliancePercentage>
  <complianceWarning>false</complianceWarning>
  <complianceWarningLevel>100</complianceWarningLevel>
  <description/>
  <name>TestSLA</name>
  <slaId>1</slaId>
  <periodBegin>2011-12-01T00:00:00+01:00</periodBegin>
  <periodCode>m</periodCode>
  <periodEnd>2012-01-01T00:00:00+01:00</periodEnd>
  <periodNumber>1</periodNumber>
  <periodStart>2011-09-01T00:00:00+02:00</periodStart>
</sla>
```
Get Configured Calculations

This call returns a list of all existing calculation methods. Custom calculation methods can be configured in the Service Level Manager.

<table>
<thead>
<tr>
<th>URL</th>
<th>/sla/calculations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – Calculation List</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/sla/calculations HTTP/1.1 Accept: application/xml</td>
</tr>
</tbody>
</table>
| Sample Reply (XML) | HTTP/1.1 200 OK Content-Type: application/xml

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<calculations>
  <calculation>
  </calculation>
</calculations>
```
<description>Average</description>
<query>SELECT AVG(percentage) AS pct FROM D_SLO_COMPLIANCE WHERE job_id=@job_id AND sla_id=@sla_id</query>
?type>0</type>
</calculation>
<calculation>
<calcId>2</calcId>
<description>Weight</description>
<query>SELECT SUM((percentage * slo_weight) /100) AS pct FROM D_SLO_COMPLIANCE WHERE job_id=@job_id AND sla_id=@sla_id</query>
?type>1</type>
</calculation>
<calculation>
<calcId>3</calcId>
<description>Worst</description>
<query>SELECT MIN(percentage) AS pct FROM D_SLO_COMPLIANCE WHERE job_id=@job_id AND sla_id=@sla_id</query>
?type>0</type>
</calculation>
<calculation>
<calcId>4</calcId>
<description>Best</description>
<query>SELECT MAX(percentage) AS pct FROM D_SLO_COMPLIANCE WHERE job_id=@job_id AND sla_id=@sla_id</query>
?type>0</type>
</calculation>
<calculation>
<calcId>5</calcId>
<description>Sequential</description>
<query>SELECT 100 - CASE WHEN SUM(100 - percentage) >= 100 THEN 100 ELSE SUM(100 - percentage) END AS pct FROM D_SLO_COMPLIANCE WHERE job_id=@job_id AND sla_id=@sla_id</query>
?type>0</type>
</calculation>
</calculations>

Sample Reply
(JSON)

HTTP/1.1 200 OK
Content-Type: application/json

{"calculation":[
{
"calcId":"1",
"description":"Average",
"query":"SELECT AVG(percentage) AS pct FROM D_SLO_COMPLIANCE WHERE job_id=@job_id AND sla_id=@sla_id",
?type":"0"
},
{
"calcId":"2",
"description":"Weight",
"query":"SELECT SUM((percentage * slo_weight) /100) AS pct FROM D_SLO_COMPLIANCE WHERE job_id=@job_id AND sla_id=@sla_id",
?type":"1"
},
{
"calcId":"3",
"description":"Worst",
"query":"SELECT MIN(percentage) AS pct FROM D_SLO_COMPLIANCE WHERE job_id=@job_id AND sla_id=@sla_id",
?type":"0"
},
{
"calcId":"4",
"description":"Best",
"query":"SELECT MAX(percentage) AS pct FROM D_SLO_COMPLIANCE WHERE job_id=@job_id AND sla_id=@sla_id",
?type":"0"
},
{...}]}
Get Calculation Jobs for a SLA

<table>
<thead>
<tr>
<th>URL</th>
<th>/sla/{sla-id}/jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – QoS Source List</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

Sample Request

GET /rest/sla/1/jobs HTTP/1.1
Accept: application/xml

Sample Reply (XML)

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<sla-jobs>
  <sla-job>
    <createDate>2011-12-12T11:17:00+01:00</createDate>
    <description>Automatic</description>
    <executeDate>2011-12-12T11:17:00+01:00</executeDate>
    <expireDate>2011-12-15T11:18:00+01:00</expireDate>
    <history>false</history>
    <jobId>42</jobId>
    <jobState>2</jobState>
    <owner>SYSTEM</owner>
    <periodBegin>2011-12-01T00:00:00+01:00</periodBegin>
    <periodEnd>2012-01-01T00:00:00+01:00</periodEnd>
    <report>false</report>
  </sla-job>
  <sla-job>
    <createDate>2011-12-14T00:05:00+01:00</createDate>
    <description>Automatic</description>
    <executeDate>2011-12-14T00:05:00+01:00</executeDate>
    <expireDate>2011-12-15T00:05:00+01:00</expireDate>
    <history>false</history>
    <jobId>48</jobId>
    <jobState>2</jobState>
    <owner>SYSTEM</owner>
    <periodBegin>2011-12-01T00:00:00+01:00</periodBegin>
    <periodEnd>2012-01-01T00:00:00+01:00</periodEnd>
    <report>false</report>
  </sla-job>
</sla-jobs>
```
### Sample Reply (JSON)

<table>
<thead>
<tr>
<th>HTTP/1.1 200 OK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content-Type: application/json</td>
</tr>
</tbody>
</table>

```json
{ "sla-job": [  
    {  
      "createDate": "2011-12-12T11:17:00+01:00",  
      "description": "Automatic",  
      "executeDate": "2011-12-12T11:17:00+01:00",  
      "expireDate": "2011-12-15T11:18:00+01:00",  
      "history": false,  
      "jobId": "42",  
      "jobState": "2",  
      "owner": "SYSTEM",  
      "periodBegin": "2011-12-01T00:00:00+01:00",  
      "periodEnd": "2012-01-01T00:00:00+01:00",  
      "report": false
    },  
    {  
      "createDate": "2011-12-14T00:05:00+01:00",  
      "description": "Automatic",  
      "executeDate": "2011-12-14T00:05:00+01:00",  
      "expireDate": "2011-12-15T00:05:00+01:00",  
      "history": false,  
      "jobId": "48",  
      "jobState": "2",  
      "owner": "SYSTEM",  
      "periodBegin": "2011-12-01T00:00:00+01:00",  
      "periodEnd": "2012-01-01T00:00:00+01:00",  
      "report": false
    }  
]  
}
```

### Get Compliance for All SLAs

<table>
<thead>
<tr>
<th>URL</th>
<th>/sla/compliances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – SLACompliance List</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

**Sample Request**

GET /rest/sla/compliances HTTP/1.1
Accept: application/xml

**Sample Reply (XML)**

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<sla-compliances>
  <sla-compliance>
  ...
  </sla-compliance>
</sla-compliances>
```
Sample Reply (JSON)

HTTP/1.1 200 OK
Content-Type: application/json

{"sla-compliance":[
{
    "created":"2011-12-12T11:17:00+01:00",
    "jobId":"42",
    "percentage":99.04,
    "periodBegin":"2011-12-01T00:00:00+01:00",
    "periodEnd":"2012-01-01T00:00:00+01:00",
    "slaId":"1",
    "breachDate":"2011-12-01T00:00:00+01:00",
    "breachValue":100.0,
    "compliancePercentage":100
},
{
    "created":"2011-12-12T11:18:00+01:00",
    "jobId":"43",
    "percentage":81.11,
    "periodBegin":"2011-12-01T00:00:00+01:00",
    "periodEnd":"2012-01-01T00:00:00+01:00",
    "slaId":"2",
    "breachDate":"2011-12-01T00:00:00+01:00",
    "breachValue":100.0,
    "compliancePercentage":100
},
{
    "created":"2011-12-14T00:05:00+01:00",
    "jobId":"48",
    "percentage":99.13,
    "periodBegin":"2011-12-01T00:00:00+01:00",
    "periodEnd":"2012-01-01T00:00:00+01:00",
    "slaId":"1",
    "breachDate":"2011-12-01T00:00:00+01:00",
    "breachValue":100.0,
    "compliancePercentage":100
},
{
    "created":"2011-12-14T00:05:00+01:00",
    "jobId":"49",
    "percentage":78.09,
    "periodBegin":"2011-12-01T00:00:00+01:00",
    "periodEnd":"2012-01-01T00:00:00+01:00",
    "slaId":"2",
    "breachDate":"2011-12-01T00:00:00+01:00",
    "breachValue":100.0,
    "compliancePercentage":100
}]}
Get SLA Compliance

<table>
<thead>
<tr>
<th>URL</th>
<th>/sla/{sla-id}/compliance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – SLACompliance</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

**Sample Request**

GET /rest/sla/1/compliance HTTP/1.1
Accept: application/xml

**Sample Reply (XML)**

```
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<sla-compliance>
  <created>2011-12-12T00:00:00+01:00</created>
  <jobId>48</jobId><percentage>99.13</percentage>
  <periodBegin>2011-12-01T00:00:00+01:00</periodBegin>
  <periodEnd>2012-01-01T00:00:00+01:00</periodEnd>
  <slaId>1</slaId>
  <breachDate>2011-12-01T00:00:00+01:00</breachDate>
  <breachValue>100.0</breachValue>
  <compliancePercentage>100</compliancePercentage>
</sla-compliance>
```

**Sample Reply (JSON)**

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "created":"2011-12-12T11:17:00+01:00",
  "jobId":"42",
  "percentage":"99.04",
  "periodBegin":"2011-12-01T00:00:00+01:00",
  "periodEnd":"2012-01-01T00:00:00+01:00",
  "slaId":"1",
  "breachDate":"2011-12-01T00:00:00+01:00",
  "breachValue":"100.0",
  "compliancePercentage":"100"
}
```
### Get SLO Definitions for a SLA

**URL**  
/sla/{sla-id}/slo

**Method**  
GET

**Input**  
-

**Returns**  
200 OK – SLO List  
401 Unauthorized  
404 Not Found

**Valid Users**  
UIM Users, Account Users

**Required Permissions**  
Web Service

**Sample Request**  
GET /rest/sla/1/slo HTTP/1.1  
Accept: application/xml

**Sample Reply (XML)**  
HTTP/1.1 200 OK  
Content-Type: application/xml  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<slos>  
  <slo>  
    <complianceAlarm>false</complianceAlarm>  
    <compliancePercentage>100</compliancePercentage>  
    <complianceWarning>false</complianceWarning>  
    <complianceWarningLevel>100</complianceWarningLevel>  
    <description></description>  
    <name>SLO1</name>  
    <slaId>1</slaId>  
    <sloId>1</sloId>  
    <weight>0</weight>  
  </slo>  
</slos>

**Sample Reply (JSON)**  
HTTP/1.1 200 OK  
Content-Type: application/json  
{"slo":  
  {  
    "complianceAlarm":"false",  
    "compliancePercentage":"100",  
    "complianceWarning":"false",  
    "complianceWarningLevel":"100",  
    "description":"",  
    "name":"SLO1",  
    "slaId":"1",  
    "sloId":"1",  
    "weight":0"  
  }
}

### Get Compliance of SLOs of a SLA

**URL**  
/sla/{sla-id}/slo-compliance
<table>
<thead>
<tr>
<th>Method</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>-</td>
</tr>
</tbody>
</table>
| Returns | 200 OK – SLOCompliance List  
401 Unauthorized  
404 Not Found |
| Valid Users | UIM Users, Account Users |
| Required Permissions | Web Service |

**Sample Request**

GET /rest/sla/1/slo-compliance HTTP/1.1  
Accept: application/xml

**Sample Reply (XML)**

```
HTTP/1.1 200 OK  
Content-Type: application/xml  

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<slo-compliances>  
  <slo-compliance>  
    <created>2011-12-12T11:00:00+01:00</created>  
    <jobId>42</jobId>  
    <percentage>99.04</percentage>  
    <periodBegin>2011-12-01T00:00:00+01:00</periodBegin>  
    <periodEnd>2012-01-01T00:00:00+01:00</periodEnd>  
    <slaId>1</slaId>  
    <compliancePercentage>0</compliancePercentage>  
    <slaPercentage>99</slaPercentage>  
    <sloId>1</sloId>  
    <sloWeight>0</sloWeight>  
  </slo-compliance>  
  <slo-compliance>  
    <created>2011-12-14T00:05:00+01:00</created>  
    <jobId>48</jobId>  
    <percentage>99.13</percentage>  
    <periodBegin>2011-12-01T00:00:00+01:00</periodBegin>  
    <periodEnd>2012-01-01T00:00:00+01:00</periodEnd>  
    <slaId>1</slaId>  
    <compliancePercentage>0</compliancePercentage>  
    <slaPercentage>99</slaPercentage>  
    <sloId>1</sloId>  
    <sloWeight>0</sloWeight>  
  </slo-compliance>  
</slo-compliances>
```

**Sample Reply (JSON)**

```
HTTP/1.1 200 OK  
Content-Type: application/json  

{"slo-compliance":[  
  {  
    "created":"2011-12-12T11:00:00+01:00",  
    "jobId":"42",  
    "percentage":"99.04",  
    "periodBegin":"2011-12-01T00:00:00+01:00",  
    "periodEnd":"2012-01-01T00:00:00+01:00",  
    "slaId":1,  
    "compliancePercentage":"0",  
    "slaPercentage":99,  
    "sloId":1,  
    "sloWeight":0  
  },  
  {  
    "created":"2011-12-14T00:05:00+01:00",  
    "jobId":"48",  
    "percentage":"99.13",  
    "periodBegin":"2011-12-01T00:00:00+01:00",  
    "periodEnd":"2012-01-01T00:00:00+01:00",  
    "slaId":1,  
    "compliancePercentage":0,  
    "slaPercentage":99,  
    "sloId":1,  
    "sloWeight":0  
  }  
]
```
## SLO Related Calls

### Get SLO Definition

| URL | /slo/{sloid}/definition |
| Method | GET |
| Input | - |
| Returns | 200 OK – SLODefinition  
401 Unauthorized  
404 Not Found |
| Valid Users | UIM Users, Account Users |
| Required Permissions | Web Service |

**Sample Request**

GET /rest/slo/1/definition HTTP/1.1  
Accept: application/json

**Sample Reply (XML)**

HTTP/1.1 200 OK  
Content-Type: application/xml  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<slo>  
<complianceAlarm>false</complianceAlarm>  
<compliancePercentage>100</compliancePercentage>  
<complianceWarning>false</complianceWarning>  
<complianceWarningLevel>100</complianceWarningLevel>  
<description></description>  
<name>SLO1</name>  
<slaId>1</slaId>  
<sloId>1</sloId>  
<weight>0</weight>  
</slo>

**Sample Reply (JSON)**

HTTP/1.1 200 OK  
Content-Type: application/json  
{
  "complianceAlarm":"false",  
  "compliancePercentage":"100",  
  "complianceWarning":"false",  
  "complianceWarningLevel":"100",  
  "description":",",  
  "name":"SLO1",  
  "slaId":"1",  
  "sloId":"1",  
  "weight":"0"
}

### Get QOS Constraints for a SLO

Returns a list of all qos-constraints that this SLO is based on.

| URL | /slo/{sloid}/constraints |
Method
GET

Input
-

Returns
200 OK – QoSConstraint List
401 Unauthorized
404 Not Found

Valid Users
UIM Users, Account Users

Required Permissions
Web Service

Sample Request
GET /rest/slo/1/constraints HTTP/1.1
Accept: application/json

Sample Reply (XML)
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<qos-constraints>
  <qos-constraint>
    <calcId>0</calcId>
    <dirty>true</dirty>
    <expectedAccuracy>0.00</expectedAccuracy>
    <operId>0</operId>
    <operator>&lt;=</operator>
    <qosConstId>1</qosConstId>
    <qosDefinition>
      <bool>false</bool>
      <description>CPU Usage</description>
      <hasMax>true</hasMax>
      <name>QOS_CPU_USAGE</name>
      <qosDefId>14</qosDefId>
      <qosGroup>QOS_MACHINE</qosGroup>
      <sourceTargets>
        <origin>primaryhub</origin>
        <source>WIN-CXP6LPT7V6G</source>
        <target>WIN-CXP6LPT7V6G</target>
      </sourceTargets>
      <sourceTargets>
        <origin>primaryhub</origin>
        <source>WIN-CXP6LPT7V6G</source>
        <target>User</target>
      </sourceTargets>
      <sourceTargets>
        <origin>primaryhub</origin>
        <source>WIN-CXP6LPT7V6G</source>
        <target>System</target>
      </sourceTargets>
      <sourceTargets>
        <origin>primaryhub</origin>
        <source>WIN-CXP6LPT7V6G</source>
        <target>Wait</target>
      </sourceTargets>
      <sourceTargets>
        <origin>primaryhub</origin>
        <source>WIN-CXP6LPT7V6G</source>
        <target>Idle</target>
      </sourceTargets>
    </qosDefinition>
  </qos-constraint>
</qos-constraints>
```xml
<qosDefinition>
  <qosConstraint calcId="0" dirty="true" expectedAccuracy="0.00" operId="0" operator="\"<\" qosConstId="1" qosDefinition="false" description="\"CPU Usage\", hasMax="true", name="\"QOS_CPU_USAGE\", qosDefId="14", qosGroup="\"QOS_MACHINE\", sourceTargets=
    [
      { origin="primaryhub", source="WIN-CXP6LPT7V6G", target="WIN-CXP6LPT7V6G", unit="pct", unitShort="%" },
      { origin="primaryhub", source="WIN-CXP6LPT7V6G", target="User", unit="pct", unitShort="%" },
      { origin="primaryhub", source="WIN-CXP6LPT7V6G", target="System", unit="pct", unitShort="%" },
      { origin="primaryhub", source="WIN-CXP6LPT7V6G", target="Wait", unit="pct", unitShort="%" },
      { origin="primaryhub", source="WIN-CXP6LPT7V6G", target="Idle", unit="pct", unitShort="%" }
    ]
  </qosConstraint>
</qosDefinition>
```

### Sample Reply (JSON)

HTTP/1.1 200 OK
Content-Type: application/json

```
{"qos-constraint":
  {
    "calcId":0,
    "dirty":true,
    "expectedAccuracy":0.00,
    "operId":0,
    "operator":<, "qosConstId":1,
    "qosDefinition":{
      "bool":false,
      "description":\"CPU Usage\", "hasMax":true,
      "name":\"QOS_CPU_USAGE\", "qosDefId":14,
      "qosGroup":\"QOS_MACHINE\", "sourceTargets":[
          unit=pct, unitShort=%
        },
        { origin:primaryhub, source:WIN-CXP6LPT7V6G, target:User,
          unit=pct, unitShort=%
        },
        { origin:primaryhub, source:WIN-CXP6LPT7V6G, target:System,
          unit=pct, unitShort=%
        },
        { origin:primaryhub, source:WIN-CXP6LPT7V6G, target:Wait,
          unit=pct, unitShort=%
        },
        { origin:primaryhub, source:WIN-CXP6LPT7V6G, target:Idle,
          unit=pct, unitShort=%
        }
      ]
    }
  }
```
Get QOS Constraint

Returns one qos-constraint of a SLO by QoS-Constraint-Id. Fetch the ID by querying the constraints for the SLO.

**URL**
/slo/constraint/{qosConstId}

**Method**
GET

**Input**
-

**Returns**
200 OK – QOSConstraint
401 Unauthorized
404 Not Found

**Valid Users**
UIM Users, Account Users

**Required Permissions**
Web Service

**Sample Request**
GET /rest/slo/constraint/1 HTTP/1.1
Accept: application/json

**Sample Reply (XML)**
HTTP/1.1 200 OK
Content-Type: application/xml

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<qos-constraint>
  <calcId>0</calcId>
  <dirty>true</dirty>
  <expectedAccuracy>0.00</expectedAccuracy>
  <operId>0</operId>
  <operator>&lt;=</operator>
  <qosConstId>1</qosConstId>
  <qosDefinition>
    <bool>false</bool>
    <description>CPU Usage</description>
    <hasMax>true</hasMax>
    <name>QOS_CPU_USAGE</name>
    <qosDefId>14</qosDefId>
    <qosGroup>QOS_MACHINE</qosGroup>
    <sourceTargets>
      <origin>primaryhub</origin>
      <source>WIN-CXP6LPT7V6G</source>
      <target>WIN-CXP6LPT7V6G</target>
    </sourceTargets>
    <sourceTargets>
      <origin>primaryhub</origin>
      <source>WIN-CXP6LPT7V6G</source>
      <target>User</target>
    </sourceTargets>
  </qosDefinition>
</qos-constraint>
```
HTTP/1.1 200 OK
Content-Type: application/json

{
    "calcId":"0",
    "dirty":"true",
    "expectedAccuracy":"0.00",
    "operId":"0",
    "operator":"<=",
    "qosConstId":"1",
    "qosDefinition":{
        "bool":"false",
        "description":"CPU Usage",
        "hasMax":"true",
        "name":"QOS_CPU_USAGE",
        "qosDefId":"14",
        "qosGroup":"QOS_MACHINE",
        "sourceTargets":[
            {
                "origin":"primaryhub",
                "source":"WIN-CXP6LPT7V6G",
                "target":"WIN-CXP6LPT7V6G"
            },
            {
                "origin":"primaryhub",
                "source":"WIN-CXP6LPT7V6G",
                "target":"User"
            },
            {
                "origin":"primaryhub",
                "source":"WIN-CXP6LPT7V6G",
                "target":"System"
            },
            {
                "origin":"primaryhub",
                "source":"WIN-CXP6LPT7V6G",
                "target":"Wait"
            }
        ]
    }
}
Get Compliance Values for All QoS Constraints of a SLO

This call returns the compliance of all qos-constraints for the given SLO. This is helpful for drilling down from the SLO to find the root of the service degradation.

URL  /slo/{sloid}/qos-compliance
Method  GET
Input  -
Returns  200 OK – QoSCompliance List
101 Unauthorized
404 Not Found
Valid Users  UIM Users, Account Users
Required Permissions  Web Service
Sample Request  GET /rest/slo/qos-compliance/1 HTTP/1.1
Accept: application/json
Sample Reply (XML)  HTTP/1.1 200 OK
Content-Type: application/xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<qos-compliances>
  <qos-compliance>
    <created>2011-12-12T11:17:00+01:00</created>
    <jobId>42</jobId>
    <percentage>99.04</percentage>
    <periodBegin>2011-12-01T00:00:00+01:00</periodBegin>
    <periodEnd>2012-01-01T00:00:00+01:00</periodEnd>
    <slaId>1</slaId>
    <accuracy>22.69</accuracy>
    <ok>3712</ok>
    <qosConstId>1</qosConstId>
    <slaid>1</slaid>
    <threshold>80</threshold>
    <total>3748</total>
  </qos-compliance>
</qos-compliances>
Get Compliance for a QoS Constraint by IDs

**URL**

/slo/qos-compliance/{qosconstid}

**Method**

GET

**Input**

-
<table>
<thead>
<tr>
<th>Returns</th>
<th>200 OK – QoSCompliance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Account Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

**Sample Request**

```plaintext
GET /rest/slo/qos-compliance/1 HTTP/1.1
Accept: application/json
```

**Sample Reply (XML)**

```xml
<?xml version=“1.0” encoding=“UTF-8” standalone=“yes”?>
<qos-compliance>
  <created>2011-12-12T11:17:00+01:00</created>
  <jobId>42</jobId>
  <percentage>99.04</percentage>
  <periodBegin>2011-12-01T00:00:00+01:00</periodBegin>
  <periodEnd>2012-01-01T00:00:00+01:00</periodEnd>
  <slaId>1</slaId>
  <accuracy>22.69</accuracy>
  <ok>3712</ok>
  <qosConstId>1</qosConstId>
  <sloId>1</sloId>
  <threshold>80</threshold>
  <total>3748</total>
  <weight>0</weight>
</qos-compliance>
```

**Sample Reply (JSON)**

```json
{
  "created": "2011-12-12T11:17:00+01:00",
  "jobId": "42",
  "percentage": "99.04",
  "periodBegin": "2011-12-01T00:00:00+01:00",
  "periodEnd": "2012-01-01T00:00:00+01:00",
  "slaId": "1",
  "accuracy": "22.69",
  "ok": "3712",
  "qosConstId": "1",
  "sloId": "1",
  "threshold": "80",
  "total": "3748",
  "weight": "0"
}
```

**Get QoS Constraint IDs for an SLO**

Returns a list of IDs for all qos-constraints that this SLO is based on.

**URL**

```
/slo/{sloid}/constraint_ids
```

**Method**

```
GET
```

**Input**

```
- 
```
## Returns
- **200 OK** – QoSConstraint ID List
- 401 Unauthorized
- 404 Not Found

## Valid Users
- UIM Users, Account Users

## Required Permissions
- Web Service

## Sample Request
```
GET /rest/slo/1/constraint_ids HTTP/1.1
Accept: application/json
```

## Sample Reply (XML)
```
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<qos_constraint_ids>
  <qos_constraint_id>1</qos_constraint_id>
  <qos_constraint_id>2</qos_constraint_id>
  <qos_constraint_id>3</qos_constraint_id>
</qos_constraint_ids>
```

## Sample Reply (JSON)
```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "qos_constraint_id":["1","2"]
}
```
**ACL Related Calls**

**The ACL Data Structure**

- **Name** – the name of the ACL
- **ldapGroupName (optional)** – The name of the LDAP group to link this ACL to. Only if using LDAP authentication.
- **Permission** – String-array of Permissions to assign to the ACL. The necessary Strings can be found in Infrastructure manager.
- **Permission_level (read-only)** – The permission level assigned to the ACL.

**Get a List of all ACLs**

<table>
<thead>
<tr>
<th>URL</th>
<th>/acls (optional query parameters: ?maxrows=&lt;int&gt;&amp;offset=&lt;int&gt;)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK – ACLList structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

**Sample Request**

GET /rest/acls HTTP/1.1
Accept: application/xml

**Sample Reply (XML)**

```
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<acls>
  <acl>
    <name>AccountOp</name>
  </acl>
  <acl>
    <name>Operator</name>
  </acl>
  <acl>
    <name>Guest</name>
  </acl>
  <acl>
    <name>Superuser</name>
  </acl>
  <acl>
    <name>Administrator</name>
  </acl>
  <acl>
    <name>Dashboard Designer</name>
  </acl>
</acls>
```

**Sample Reply (JSON)**

```
HTTP/1.1 200 OK
Content-Type: application/json

{
    "acl":[
        
        {"name": "Operator"},
        
        {"name": "AccountOp"},
        
        {"name": "Guest"},
        
        {"name": "Superuser"},
        
        {"name": "Administrator"},
        
        {"name": "Dashboard Designer"}
    ]
}
```
### Get ACL of Current User

Returns the ACL of the user invoking the callback.

<table>
<thead>
<tr>
<th>URL</th>
<th>/acls/current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK - ACL structure</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users, Contacts</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
</tr>
</tbody>
</table>

**Sample Request**

GET /rest/acls/current HTTP/1.1
Accept: application/xml

**Sample Reply (XML)**

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<acl>
    <name>Superuser</name>
    <accountLink>23</accountLink>
    <permission>Accept</permission>
    <permission>Account Administration</permission>
    <permission>Acknowledge</permission>
    <permission>Alarm Details</permission>
    <permission>Alarm History</permission>
    <permission>Alarm Management</permission>
    <permission>Alarm Summary</permission>
    <permission>Archive Management</permission>
    <permission>Assign</permission>
    <permission>Basic Management</permission>
    <permission>Change Password</permission>
    <permission>Custom Dashboards</permission>
    <permission>Custom Reports</permission>
    <permission>Dashboard Design</permission>
    <permission>Dashboard Designer</permission>
    <permission>Dashboard Download</permission>
    <permission>Dashboard Publish</permission>
    <permission>Dashboard Upload</permission>
    <permission>Default Customization</permission>
    <permission>Discovery</permission>
    <permission>Discovery Management</permission>
    <permission>Discovery Pie</permission>
    <permission>Distribution</permission>
    <permission>Dynamic Views</permission>
    <permission>Dynamic Views Dashboards</permission>
    <permission>Dynamic Views Reports</permission>
    <permission>Dynamic Views States</permission>
    <permission>Execution Level 1</permission>
    <permission>Execution Level 2</permission>
</acl>
```
<permission>Execution Level 3</permission>
<permission>Extended Security</permission>
<permission>Invisible Alarms</permission>
<permission>License Management</permission>
<permission>Manage ACL</permission>
<permission>Manage Profiles</permission>
<permission>Management Tools</permission>
<permission>Modify Profiles</permission>
<permission>Program Options</permission>
<permission>Reassign</permission>
<permission>Report Designer</permission>
<permission>SDP</permission>
<permission>SLM Admin</permission>
<permission>SLM View</permission>
<permission>Unassign</permission>
<permission>Unified Reports</permission>
<permission>User Administration</permission>
<permission>User Customization</permission>
<permission>User Monitoring</permission>
<permission>Web Publish</permission>
<permission>Web Service</permission>

Sample Reply (JSON)

HTTP/1.1 200 OK
Content-Type: application/json
{
    "name":"Superuser",
}

Get a ACL by Name

URL /acls/{acl-name}
Method GET
Input Acl-name: the name of the ACL.
Returns 200 OK - ACL structure
401 Unauthorized
404 Not Found
Valid Users UIM Users
Required Permissions Web Service, Manage ACL
Sample Request GET /rest/acls/guest HTTP/1.1
Accept: application/xml
Sample Reply HTTP/1.1 200 OK
## Create a New ACL

<table>
<thead>
<tr>
<th><strong>URL</strong></th>
<th>/acls/{acl-name}</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>POST</td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>ACL structure</td>
</tr>
<tr>
<td><strong>Returns</strong></td>
<td>202 No Content = OK</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td></td>
<td>409 Resource Conflict (if the ACL already exists)</td>
</tr>
</tbody>
</table>

### Valid Users
- UIM Users

### Required Permissions
- Web Service, Manage ACL

### Sample Request (XML)

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<acl>
  <name>restTestACL</name>
  <accountLink>23</accountLink>
  <permission>Assign</permission>
  <permission>Alarm Summary</permission>
</acl>
```

### Sample Reply (XML)

```
HTTP/1.1 204 No Content
```

### Sample Request (XML)

```
POST /rest/acls/restTestACL HTTP/1.1
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<acl>
  <name>restTestACL</name>
  <accountLink>23</accountLink>
  <permission>Assign</permission>
  <permission>Alarm Summary</permission>
</acl>
```

### Sample Reply (XML)

```
HTTP/1.1 204 No Content
```
Create a New ACL by Copying an Existing ACL

**URL**
/acls/{acl-name}/from/{template-acl-name}

**Method**
POST

**Input**
- Acl-name: the name of the ACL to be created
- Template-acl-name: the name of the ACL to be copied from
- Optionally: ACL structure containing ldap-group-name to match the new acl to

**Returns**
- 204 No Content = OK
- 400
- 401 Unauthorized
- 404 Not Found
- 409 Resource Conflict (if the ACL already exists)
- 400 Bad Request (if the template does not exist)

**Valid Users**
UIM Users

**Required Permissions**
Web Service, Manage ACL

**Sample Request (XML)**
POST /rest/acls/myCopiedGuestAcl/from/Guest HTTP/1.1
Content-Type: application/xml

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<acl>
  <name>myCopiedGuestAcl</name>
</acl>
```

**Sample Request (JSON)**
POST /rest/acls/myCopiedGuestAcl/from/Guest HTTP/1.1
Content-Type: application/json

```json
"acl": [
  {"name":"myCopiedGuestAcl"},
  {"ldapGroupName":"ldapgroupname"}
]
```

**Modify an Existing ACL**

**URL**
/acls/{acl-name}

**Method**
PUT
### Input
ACL structure containing the modified ACL.

### Returns
- 204 No Content = OK
- 401 Unauthorized
- 404 Not Found
- 409 Resource Conflict (if name in the url and in the structure don’t match)

### Valid Users
UIM Users

### Required Permissions
Web Service, Manage ACL

### Sample Request (XML)
```
PUT /rest/acls/restTestACL HTTP/1.1
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<acl>
  <name>restTestACL</name>
  <permission>Assign</permission>
  <permission>Alarm Summary</permission>
</acl>
```

### Sample Request (JSON)
```
PUT /rest/acls/restTestACL HTTP/1.1
Content-Type: application/json

{
  "name": "restTestACL",
  "permission": ["Assign", "Alarm Summary"]
}
```

### Sample Reply
HTTP/1.1 204 No Content

### Delete an ACL

<table>
<thead>
<tr>
<th>URL</th>
<th>/acls/{acl-name}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Input</td>
<td>Acl-name: the name of the acl to be deleted.</td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content = OK</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Manage ACL</td>
</tr>
</tbody>
</table>

### Sample Request
DELETE /rest/acls/restTestACL HTTP/1.1

### Sample Reply
HTTP/1.1 204 No Content
## Origin Related Calls

### Get All Origins

<table>
<thead>
<tr>
<th>URL</th>
<th>/origins</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK - OriginList structure</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/origins HTTP/1.1 Accept: application/xml</td>
</tr>
<tr>
<td>Sample Reply (XML)</td>
<td>HTTP/1.1 200 OK Content-Type: application/xml</td>
</tr>
<tr>
<td></td>
<td>&lt;?xml version=&quot;1.0&quot; encoding=&quot;UTF-8&quot; standalone=&quot;yes&quot;?&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;origins&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;origin&gt;customerA&lt;/origin&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;origin&gt;customerB&lt;/origin&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;origin&gt;primaryhub&lt;/origin&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;/origins&gt;</td>
</tr>
</tbody>
</table>

### Get All Origins and Related Accounts

This call returns a list of all origins and a list of account ids that are associated with this origin.

<table>
<thead>
<tr>
<th>URL</th>
<th>/origins/mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK - OriginList structure</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/origins/mapping HTTP/1.1 Accept: application/xml</td>
</tr>
<tr>
<td>Sample Reply</td>
<td>HTTP/1.1 200 OK</td>
</tr>
</tbody>
</table>
### (XML)

Content-Type: application/xml

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<originmappings>
    <originmapping>
        <account>0</account>
        <name>customerA</name>
    </originmapping>
    <originmapping>
        <account>3</account>
        <name>customerB</name>
    </originmapping>
    <originmapping>
        <account>3</account>
        <name>primaryhub</name>
    </originmapping>
</originmappings>
```

### Sample Reply (JSON)

HTTP/1.1 200 OK  
Content-Type: application/xml

```json
{"originmappings":
    [   
        {"account":0,"name":"customerA"},
        {"account":3,"name":"customerB"},
        {"account":3,"name":"primaryhub"}
    ]
}
```
Variable Related Calls

**Get Defined Variables**

<table>
<thead>
<tr>
<th>URL</th>
<th>/variables/{type} (valid types: QoS, SLO, NimBUS Request, Alarm Filter,</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
<td></td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK - VariableList structure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
<td></td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
<td></td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
<td></td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
<td></td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/variables/QoS HTTP/1.1 Accept: application/xml</td>
<td></td>
</tr>
<tr>
<td>Sample Reply</td>
<td>HTTP/1.1 200 OK Content-Type: application/xml</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;?xml version=&quot;1.0&quot; encoding=&quot;UTF-8&quot; standalone=&quot;yes&quot;?&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;variable-list&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;variable&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;name&gt;QoS1&lt;/name&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;status&gt;0&lt;/status&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;type&gt;QoS&lt;/type&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;value&gt;0.00&lt;/value&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;/variable&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;/variable-list&gt;</td>
<td></td>
</tr>
</tbody>
</table>

**Get a Variable**

<table>
<thead>
<tr>
<th>URL</th>
<th>/variables/{variable-name}/{type} (valid types: QoS, SLO, NimBUS Request, Alarm Filter,</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
<td></td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK - Variable structure</td>
<td></td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
<td></td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
<td></td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
<td></td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service</td>
<td></td>
</tr>
<tr>
<td>Sample Request</td>
<td>GET /rest/variables/QoS/QoS HTTP/1.1 Accept: application/xml</td>
<td></td>
</tr>
</tbody>
</table>
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<variable>
  <name>QoS1</name>
  <status>0</status>
  <type>QoS</type>
  <value>0.00</value>
</variable>
**Message Structure Definitions**

**Alarm Filter**

The Alarm Filter object that can be passed to various alarm related calls can contain any combination of the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The alarm id (nimid) of the alarm.</td>
</tr>
<tr>
<td>level</td>
<td>A comma-separated list of all levels that should be returned, where:</td>
</tr>
<tr>
<td></td>
<td>1 = Information, 2 = Warning, 3 = Minor, 4 = Major, 5 = Critical</td>
</tr>
<tr>
<td>hostname</td>
<td>The hostname. This value can use perl regular expression syntax.</td>
</tr>
<tr>
<td>source</td>
<td>The alarm source. This value can use perl regular expression syntax.</td>
</tr>
<tr>
<td>domain</td>
<td>The domain of the alarm. This value can use perl regular expression syntax.</td>
</tr>
<tr>
<td>hub</td>
<td>The hub of the alarm. This contains the first hub that received the alarm.</td>
</tr>
<tr>
<td></td>
<td>This value can use perl regular expression syntax.</td>
</tr>
<tr>
<td>robot</td>
<td>The robot name that raised the alarm. This value can use perl regular</td>
</tr>
<tr>
<td></td>
<td>expression syntax.</td>
</tr>
<tr>
<td>probe</td>
<td>The name of the probe that raised the alarm. This value can use perl</td>
</tr>
<tr>
<td></td>
<td>regular expression syntax.</td>
</tr>
<tr>
<td>subsystem_id</td>
<td>The subsystem_id of the alarm. See the subsystem-tree in the alarm server</td>
</tr>
<tr>
<td></td>
<td>(nas). This value can use perl regular expression syntax.</td>
</tr>
<tr>
<td>subsystem</td>
<td>The subsystem name of the alarm. See the subsystem-tree in the alarm server</td>
</tr>
<tr>
<td></td>
<td>(nas). This value can use perl regular expression syntax.</td>
</tr>
<tr>
<td>origin</td>
<td>The origin of the alarm. This value can use perl regular expression syntax.</td>
</tr>
<tr>
<td>message_count</td>
<td>The amount of times that this alarm has recurred. The value should be an</td>
</tr>
<tr>
<td></td>
<td>operator and an integer value.</td>
</tr>
<tr>
<td></td>
<td>For Json, valid operators: “=”, “&lt;”, “&lt;”, “&gt;”, “&lt;&gt;”, “!”</td>
</tr>
<tr>
<td></td>
<td>e.g. “&gt;2” will only return alarms that recurred more than twice.</td>
</tr>
<tr>
<td></td>
<td>For xml, valid operators: “=”, “&lt;”, “&amp;gth;”, “&gt;”, “!”</td>
</tr>
<tr>
<td></td>
<td>e.g. “&gt;2” will only return alarms that recurred more than twice.</td>
</tr>
<tr>
<td>message</td>
<td>The message text of the alarm. This value can use perl regular expression</td>
</tr>
<tr>
<td></td>
<td>syntax.</td>
</tr>
<tr>
<td>userTag1</td>
<td>The value of the user tag 1. This value can use perl regular expression</td>
</tr>
<tr>
<td></td>
<td>syntax.</td>
</tr>
<tr>
<td>userTag2</td>
<td>The value of the user tag 2. This value can use perl regular expression</td>
</tr>
<tr>
<td></td>
<td>syntax.</td>
</tr>
<tr>
<td>custom1</td>
<td>The value of the custom field 1. This value can use perl regular expression</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>custom2</td>
<td>The value of the custom field 2. This value can use Perl regular expression syntax.</td>
</tr>
<tr>
<td>custom3</td>
<td>The value of the custom field 3. This value can use Perl regular expression syntax.</td>
</tr>
<tr>
<td>custom4</td>
<td>The value of the custom field 4. This value can use Perl regular expression syntax.</td>
</tr>
<tr>
<td>custom5</td>
<td>The value of the custom field 5. This value can use Perl regular expression syntax.</td>
</tr>
<tr>
<td>assigned_to</td>
<td>The name of the user the alarm is currently assigned to. This value can use Perl regular expression syntax.</td>
</tr>
<tr>
<td>timeArrival</td>
<td>A date time value (format: 2011-12-15T01:35:55.524+01:00) specifying the start of the filter time. Only alarms that arrived at the alarm server after this date will be returned.</td>
</tr>
<tr>
<td>timeLast</td>
<td>The time the alarm was last updated.</td>
</tr>
<tr>
<td>visible</td>
<td>This is a Boolean. Valid values are:</td>
</tr>
<tr>
<td></td>
<td>- False</td>
</tr>
<tr>
<td></td>
<td>- True</td>
</tr>
<tr>
<td></td>
<td>If set to true, only visible alarms are returned. If set to false, only invisible alarms are returned.</td>
</tr>
</tbody>
</table>

Please note that all fields are evaluated on each alarm using the Boolean **AND**. That means that only alarms that match ALL of your filter criteria are returned.
## Maintenance Mode Related Calls

### Add a Schedule

**Note:** Please see appendix A for definition of the inputs to add_schedule.

<table>
<thead>
<tr>
<th>URL</th>
<th>/maintenance_mode/{domain}/{hub}/{robot}/add_schedule</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>POST</td>
</tr>
<tr>
<td><strong>Inputs</strong></td>
<td>Schedule Structure</td>
</tr>
<tr>
<td><strong>Returns</strong></td>
<td>200 OK &amp; Schedule Id XML/JSON</td>
</tr>
<tr>
<td></td>
<td>Error Message</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td><strong>Valid Users</strong></td>
<td>UIM Users</td>
</tr>
<tr>
<td><strong>Required Permissions</strong></td>
<td>Web Service, Basic Management</td>
</tr>
</tbody>
</table>

**Sample Request (XML)**

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<schedule>
  <name>Test Sched A</name>
  <description>Test schedule for Maintenance Mode REST API.</description>
  <start_date_time>
    <month>2</month>
    <day>14</day>
    <year>2014</year>
    <timestamp>
      <hours>15</hours>
      <minutes>30</minutes>
      <seconds>22</seconds>
    </timestamp>
  </start_date_time>
  <end_time>
    <type>duration</type>
    <end_date_time>
      <month></month>
      <day></day>
      <year></year>
      <timestamp>
        <hours>24</hours>
        <minutes></minutes>
        <seconds></seconds>
      </timestamp>
    </end_date_time>
    <duration>
      <hours>24</hours>
      <minutes></minutes>
      <seconds></seconds>
    </duration>
  </end_time>
</schedule>
```
```xml
<account_id>2</account_id>
<recurrence_pattern>Monthly</recurrence_pattern>
<recurrence_period>1</recurrence_period>
<recurrence_days_of_the_week></recurrence_days_of_the_week>
<recurrence_day_of_the_month>15</recurrence_day_of_the_month>
<recurrence_instance></recurrence_instance>
<recurrence_end_date_time>
  <month>2</month>
  <day>14</day>
  <year>2015</year>
  <timestamp>
    <hours>16</hours>
    <minutes>30</minutes>
    <seconds>22</seconds>
  </timestamp>
</recurrence_end_date_time>
<timezone>Australia/Lindeman</timezone>
</schedule>
```

**Sample Reply (XML)**

HTTP/1.1 200 OK
Content-Type: application/xml

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<schedule>
  <scheduleId>8</scheduleId>
</schedule>
```

**Sample Request (JSON)**

```
POST /rest/maintenance_mode/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/add_schedule HTTP/1.1
Content-Type: application/json
Accept: application/json

{
  "name": "Test Sched A",
  "description": "Test schedule for Maintenance Mode REST API",
  "start_date_time": {
    "month": "2",
    "day": "28",
    "year": "2014",
    "timestamp": {
      "hours": "15",
      "minutes": "30",
      "seconds": "22"
    }
  },
  "end_time": {
    "type": "duration",
    "end_date_time": {
      "month": "",
      "day": "",
      "year": "",
      "timestamp": {
        "hours": "",
        "minutes": "",
        "seconds": ""
      }
    },
    "duration": {
      "hours": "24",
      "minutes": "",
      "seconds": ""
    }
  },
  "account_id": "2",
  "recurrence_pattern": "3",
  "recurrence_period": 1,
  "recurrence_days_of_the_week": [
    "S"
  ],
  "recurrence_instance": 15,
  "timezone": "Australia/Lindeman"
}
```
### Add Computer Systems to a Schedule

**URL**
/maintenance_mode/{domain}/{hub}/{robot}/add_computer_systems_to_schedule/{schedule_id}

**Method**
POST

**Inputs**
ComputerSystemList structure containing ComputerSystemIds

**Returns**
204 No Content (=OK)
Error Message
401 Unauthorized
404 Not Found

**Valid Users**
UIM Users

**Required Permissions**
Web Service, Basic Management

**Sample Request (XML)**
```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<cs-list>
<cs>1</cs>
<cs>2</cs>
<cs>3</cs>
</cs-list>
```

**Sample Reply (XML)**
HTTP/1.1 204 No Content

**Sample Request (JSON)**
```json
POST /rest/maintenance_mode/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/add_computer_systems_to_schedule/8 HTTP/1.1
Content-Type: application/json
```

**Sample Reply (JSON)**
HTTP/1.1 204 No Content
### Remove Computer Systems from a Schedule

<table>
<thead>
<tr>
<th>URL</th>
<th>/maintenance_mode/{domain}/{hub}/{robot}/remove_computer_systems_from_schedule/{schedule_id}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Inputs</td>
<td>ComputerSystemList structure containing Computer System IDs</td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>Error Message</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Basic Management</td>
</tr>
</tbody>
</table>

**Sample Request (XML)**

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<cs-list>
  <cs>1</cs>
  <cs>2</cs>
  <cs>3</cs>
</cs-list>
```

**Sample Reply**

HTTP/1.1 204 No Content

**Sample Request (JSON)**

```json
{
  "cs":["1","2","3"]
}
```

### Modify a Schedule

<table>
<thead>
<tr>
<th>URL</th>
<th>/maintenance_mode/{domain}/{hub}/{robot}/modify_schedule/{schedule_id}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Inputs</td>
<td>Schedule Structure</td>
</tr>
<tr>
<td>Returns</td>
<td>200 OK &amp; Schedule Id XML/JSON</td>
</tr>
<tr>
<td></td>
<td>Error Message</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
</tbody>
</table>
Valid Users

<table>
<thead>
<tr>
<th>UIM Users</th>
</tr>
</thead>
</table>

Required Permissions

<table>
<thead>
<tr>
<th>Web Service, Basic Management</th>
</tr>
</thead>
</table>

Sample Request (XML)

```
PUT /rest/maintenance_mode/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/modify_schedule/8
HTTP/1.1
Content-Type: application/xml
Accept: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<schedule>
  <name>Test Sched A</name>
  <description>Test schedule for Maintenance Mode REST API.</description>
  <start_date_time>
    <month>2</month>
    <day>14</day>
    <year>2014</year>
    <timestamp>
      <hours>15</hours>
      <minutes>30</minutes>
      <seconds>22</seconds>
    </timestamp>
  </start_date_time>
  <end_time>
    <type>duration</type>
    <end_date_time>
      <month></month>
      <day></day>
      <year></year>
      <timestamp>
        <hours></hours>
        <minutes></minutes>
        <seconds></seconds>
      </timestamp>
    </end_date_time>
    <duration>
      <hours>24</hours>
      <minutes></minutes>
      <seconds></seconds>
    </duration>
  </end_time>
  <account_id>2</account_id>
  <recurrence_pattern>Monthly</recurrence_pattern>
  <recurrence_period>1</recurrence_period>
  <recurrence_days_of_the_week>Tuesday</recurrence_days_of_the_week>
  <recurrence_day_of_the_month>15</recurrence_day_of_the_month>
  <recurrence_instance>2</recurrence_instance>
  <recurrence_end_date_time>
    <month>2</month>
    <day>14</day>
    <year>2014</year>
    <timestamp>
      <hours>16</hours>
      <minutes>30</minutes>
      <seconds>22</seconds>
    </timestamp>
  </recurrence_end_date_time>
  <recurrence_instance>2</recurrence_instance>
  <timezone>Australia/Lindeman</timezone>
</schedule>
```
Sample Request (JSON)

```json
PUT /rest/maintenance_mode/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/add_schedule HTTP/1.1
Content-Type: application/json

{
    "name": "Test Sched A",
    "description": "Test schedule for Maintenance Mode REST API."
    "start_date_time": {
        "month": "2",
        "day": "28",
        "year": "2014",
        "timestamp": {
            "hours": "15",
            "minutes": "30",
            "seconds": "22"
        }
    },
    "end_time": {
        "type": "duration",
        "end_date_time": {
            "month": "",
            "day": "",
            "year": "",
            "timestamp": {
                "hours": "",
                "minutes": "",
                "seconds": ""
            }
        },
        "duration": {
            "hours": "24",
            "minutes": "",
            "seconds": ""
        }
    },
    "account_id": "2",
    "recurrence_pattern": "Monthly",
    "recurrence_period": "1",
    "recurrence_days_of_the_week": "",
    "recurrence_day_of_the_month": "15",
    "recurrence_instance": "",
    "recurrence_end_date_time": {
        "month": "2",
        "day": "14",
        "year": "2015",
        "timestamp": {
            "hours": "16",
            "minutes": "30",
            "seconds": "22"
        }
    },
    "timezone": "Australia/Lindeman"
}
```

Sample Reply (JSON)

```
HTTP/1.1 200 OK
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<schedule>
    <scheduleId>8</scheduleId>
</schedule>
```

Sample Reply

HTTP/1.1 200 OK
Content-Type: application/xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<schedule>
    <scheduleId>8</scheduleId>
</schedule>
```
### Delete a Schedule

**URL**  
/maintenance_mode/{domain}/{hub}/{robot}/delete_schedule/{schedule_id}

**Method**  
DELETE

**Inputs**  
-

**Returns**  
- 204 No Content (=OK)
- 401 Unauthorized
- 404 Not Found

**Valid Users**  
UIM Users

**Required Permissions**  
Web Service, Basic Management

**Sample Request**

```
DELETE rest/maintenance_mode/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/delete_schedule/6
HTTP/1.1
```

**Sample Reply**

```
HTTP/1.1 204 No Content
```

### Add Computer Systems to an Active Window

**URL**  
/maintenance_mode/{domain}/{hub}/{robot}/add_systems_to_active_window/{schedule_id}

**Method**  
POST

**Inputs**  
ComputerSystemList structure containing Computer System IDs

**Returns**  
- 204 No Content (=OK)
- Error Message
- 401 Unauthorized
- 404 Not Found

**Valid Users**  
UIM Users

**Required Permissions**  
Web Service, Basic Management

**Sample Request (XML)**

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<cs-list>
  <cs>7</cs>
  <cs>12</cs>
</cs-list>
```
### Get Historical Maintenance Windows

**URL**
```
/maintenance_mode/{domain}/{hub}/{robot}/
get_historical_maintenance_windows/{from}/{to}/{device_id}/{timezone}
```

Parameter explanation:
- “from” should be a date in the format yyyyddMMHHmm (year, day, month, hours, minutes. e.g. 201114071200 is 2011, 14, 07, 12, 00 or July 14, 2011 12pm)
- “to” should be a date in the format yyyyddMMHHmm (e.g. 201111100938)
- “device_id” the device id (eg. D387491CA4847AA722EBF4DA4A4B3C222) or “null”. If “null” is given to indicate no device id, all historical maintenance windows from the start to the end are returned.
- “timezone” the timezone the maintenance window start and end date-times will be displayed in. A list of valid timezones is provided in Appendix A.

**Method**
GET

**Inputs**

- 

**Returns**

- 204 No Content (=OK)
- 401 Unauthorized
- Error Message
- 404 Not Found

**Valid Users**
UIM Users

**Required Permissions**
Web Service, Basic Management

**Sample Request (XML)**
```
GET /rest/maintenance_mode/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/add_computer_systems_to_active_window/7 HTTP/1.1
Accept: application/xml
```

**Sample Reply (XML)**
```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<maintenance_window_list>
  <maintenance_window>
    <deviceId>DC34DFE247E1244E79D5B7F189760ED62</deviceId>
    <start_date_time>02/21/2014 11:30:26</start_date_time>
    <end_date_time>02/21/2014 11:30:26</end_date_time>
  </maintenance_window>
  <maintenance_window>
    <deviceId>DC34DFE247E1244E79D5B7F189760ED62</deviceId>
  </maintenance_window>
</maintenance_window_list>
```
Sample Reply (JSON)

```json
Content-Type: application/json

{
    "maintenance_window": [
        {
            "start_date_time": "02/25/2014 09:32:45",
            "end_date_time": "02/27/2014 09:32:45"
        },
        {
            "start_date_time": "02/25/2014 09:32:46",
            "end_date_time": "02/25/2014 17:30:00"
        },
        {
            "start_date_time": "02/25/2014 09:32:46",
            "end_date_time": "02/25/2014 16:30:00"
        },
        ...
    ]
}
```

Get the Next Fire Time for Schedules

**Note:** "utc_next_fire_time" is the date representation of the millisecond offset from the Unix epoch inserted into the database.

**URL**
/maintenance_mode/{domain}/{hub}/{robot}/get_next_fire_time_for_schedules?timezone=America/Denver

**Parameter explanation:**
- You can specify a timezone to display schedule next fire times in. If no timezone is specified, the timezone of the server on which the wasp probe is running will be used. A list of valid timezones is provided in Appendix A.

**Method**
GET

**Inputs**

**Returns**
204 No Content (=OK)
401 Unauthorized
Error Message
404 Not Found

**Valid Users**
UIM Users

**Required Permissions**
Web Service, Basic Management
Sample Request (XML) | GET /rest/maintenance_mode/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/get_next_fire_time_for_schedules?timezone=Australia/Lindeman HTTP/1.1
Accept: application/xml

Sample Reply (XML) | Content-Type: application/xml

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<schedule_fire_times>
  <fire_time>
    <schedule_id>50</schedule_id>
    <next_fire_time>03/05/2014 03:30:00</next_fire_time>  </fire_time>
  <fire_time>
    <schedule_id>53</schedule_id>
    <next_fire_time>03/16/2014 03:00:00</next_fire_time>  </fire_time>
  ...
</schedule_fire_times>
```

Sample Reply (JSON) | Content-Type: application/json

```json
{
  "fire_time": [
    {
      "schedule_id": "45",
      "next_fire_time": "03/04/2014 14:00:10",
    },
    {
      "schedule_id": "49",
      "next_fire_time": "03/06/2014 10:30:00",
    },
    ...
    {
      "schedule_id": "122",
      "next_fire_time": "03/14/2014 16:30:22"
    }
  ]
}
```

New Duration for an Active Window

<table>
<thead>
<tr>
<th>URL</th>
<th>/maintenance_mode/{domain}/{hub}/{robot}/new_duration_for_active_window/{schedule_id}/{duration_in_minutes}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
</tbody>
</table>
| Inputs | String schedule_id  
String duration |
| Returns | 204 No Content (=OK)  
401 Unauthorized  
Error Message  
404 Not Found |
<p>| Valid Users | UIM Users |
| Required | Web Service, Basic Management |</p>
<table>
<thead>
<tr>
<th>Permissions</th>
<th>PUT /rest/maintenance_mode/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/new_duration_for_active_window/7/30 HTTP/1.1</th>
</tr>
</thead>
</table>

**Stop Maintenance**

<table>
<thead>
<tr>
<th>URL</th>
<th>/maintenance_mode/{domain}/{hub}/robot/stop_maintenance/{schedule_id}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Inputs</td>
<td>ComputerSystemList structure containing ComputerSystemIds</td>
</tr>
<tr>
<td>Returns</td>
<td>204 No Content (=OK)</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Basic Management</td>
</tr>
</tbody>
</table>

**Sample Request (XML)**

```xml
<cs-list>
  <cs>7</cs>
  <cs>12</cs>
</cs-list>
```

**Sample Reply**

HTTP/1.1 204 No Content

**Sample Request (JSON)**

```json
{
  "cs": ["7", "12"]
}
```
Custom Property related calls

**Note:** The only supported custom property at this time is Origin. Origin is the property_key.

**Replace custom properties**

This call adds custom properties to the specified computer system, replacing any existing properties.

<table>
<thead>
<tr>
<th>URL</th>
<th>custom_properties/{domain}/{hub}/{robot}/replace/{cs_id}/{property_key}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input</td>
<td>List of property values</td>
</tr>
<tr>
<td>Returns</td>
<td>204 - No Content</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
</tbody>
</table>

**Sample Request**

```
POST /rest/custom_properties/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/replace/2/Origin HTTP/1.1
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<values_list>
  <value>CustomerA</value>
  <value>CustomerB</value>
  <value>CustomerC</value>
</values_list>
```

**Sample Reply (XML)**

HTTP/1.1 204 No Content

**Sample Request (JSON)**

```
POST /rest/custom_properties/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/replace/2/Origin HTTP/1.1
Content-Type: application/json

{"value": ["CustomerA", "CustomerB", "CustomerC"]}
```

**Add custom properties**

This call adds custom properties to the specified computer system.

<table>
<thead>
<tr>
<th>URL</th>
<th>custom_properties/{domain}/{hub}/{robot}/add/{cs_id}/{property_key}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Input</td>
<td>List of property values</td>
</tr>
<tr>
<td>Returns</td>
<td>204 - No Content</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
</tbody>
</table>

**Sample Request**

```
POST /rest/custom_properties/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/add/2/Origin HTTP/1.1
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<values_list>
  <value>CustomerA</value>
  <value>CustomerB</value>
  <value>CustomerC</value>
</values_list>
```

**Sample Reply (XML)**

HTTP/1.1 204 No Content

**Sample Request (JSON)**

```
POST /rest/custom_properties/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/add/2/Origin HTTP/1.1
Content-Type: application/json

{"value": ["CustomerA", "CustomerB", "CustomerC"]}
```
<table>
<thead>
<tr>
<th>Valid Users</th>
<th>UIM Users</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Required Permissions</strong></td>
<td>Web Service, Account Administration</td>
</tr>
</tbody>
</table>
| **Sample Request** | POST /rest/custom_properties/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/add/2/Origin HTTP/1.1 Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<values_list>
  <value>CustomerA</value>
  <value>CustomerB</value>
  <value>CustomerC</value>
</values_list>
| **Sample Reply (XML)** | HTTP/1.1 204 No Content |
| **Sample Request (JSON)** | POST /rest/custom_properties/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/add/2/Origin HTTP/1.1 Content-Type: application/json

{"value": ["CustomerA", "CustomerB", "CustomerC"]} |

**Get custom properties**
This call returns the values for the specified custom property.

| **URL** | custom_properties/\{domain\}/\{hub\}/\{robot\}/get/\{cs_id\}/\{property_key\} |
| **Method** | GET |
| **Input** | - |
| **Returns** | 200 - OK, Property List |
| | 401 Unauthorized |
| | 404 Not Found |
| **Valid Users** | UIM Users |
| **Required Permissions** | Web Service, Account Administration |
| **Sample Request** | GET /rest/custom_properties/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/get/2/Origin HTTP/1.1 Accept: application/xml |
| **Sample Reply (XML)** | HTTP/1.1 202 OK Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<values_list>
  <value>CustomerA</value>
  <value>CustomerB</value>
  <value>CustomerC</value>
</values_list>
| **Sample Reply (JSON)** | HTTP/1.1 200 OK Content-Type: application/json

{"value": ["CustomerA", "CustomerB", "CustomerC"]}
Remove custom properties

This call removes the specified values from the specified custom property. If the specified value is the only value, the property is removed. Otherwise, other values remain.

<table>
<thead>
<tr>
<th>URL</th>
<th>custom_properties/(domain)/(hub)/(robot)/ remove/(cs_id)/(property_key)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Input</td>
<td>List of property values</td>
</tr>
<tr>
<td>Returns</td>
<td>204 - No Content</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
</tbody>
</table>

Sample Request (XML)

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<values_list>
  <value>CustomerA</value>
  <value>CustomerB</value>
  <value>CustomerC</value>
</values_list>
```

Sample Request (JSON)

```json
{"value": ["CustomerA", "CustomerB", "CustomerC"]}
```

Sample Reply (XML)

HTTP/1.1 204 No Content

Clear custom properties

This call clears/removes the property and all of its values from the specified computer system.

<table>
<thead>
<tr>
<th>URL</th>
<th>custom_properties/(domain)/(hub)/(robot)/ clear/(cs_id)/(property_key)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Input</td>
<td>-</td>
</tr>
<tr>
<td>Returns</td>
<td>204 - No Content</td>
</tr>
<tr>
<td></td>
<td>401 Unauthorized</td>
</tr>
<tr>
<td></td>
<td>404 Not Found</td>
</tr>
<tr>
<td>Valid Users</td>
<td>UIM Users</td>
</tr>
<tr>
<td>Required Permissions</td>
<td>Web Service, Account Administration</td>
</tr>
</tbody>
</table>

Sample

DELETE /rest/custom_properties/w2k8r2-x64-lc_domain/w2k8r2-x64-lc_hub/w2k8r2-x64-lc/clear/2/Origin HTTP/1.1
<table>
<thead>
<tr>
<th>Request</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample Reply (XML)</td>
<td>HTTP/1.1 204 No Content</td>
</tr>
</tbody>
</table>
Appendix A – Definition of Inputs for the add_schedule Call

The minimum required inputs to add a schedule for maintenance mode are the schedule start date/time and end date/time or duration.

The add_schedule inputs are as follows:

name – A name to help identify the schedule.

description – A description to help identify the purpose of the schedule.

start_date_time – The day and time the maintenance period will start. Start date time must be greater than the current time. Enter integer values for month, day, and year. Optionally enter hours, minutes, and seconds.

For example, the following start_date_time represents February 14, 2014 at 12:00:00 am.

```
<start_date_time>
  <month>2</month>
  <day>14</day>
  <year>2014</year>
  <timestamp>
    <hours></hours>
    <minutes></minutes>
    <seconds></seconds>
  </timestamp>
</start_date_time>
```

The following start_date_time represents February 14, 2014 at 3:30:22 pm.

```
<start_date_time>
  <month>2</month>
  <day>14</day>
  <year>2014</year>
  <timestamp>
    <hours>15</hours>
    <minutes>30</minutes>
    <seconds>22</seconds>
  </timestamp>
</start_date_time>
```

end_time – The time the maintenance period will end. End time can be specified as a duration or as an end_date_time.

duration – The amount of time in minutes the maintenance period will last. Enter duration in the type field. Enter integer values for hours, minutes, or seconds and the system will convert the duration to minutes. Seconds are rounded to the nearest minute.

For example,

```
<end_time>
  <type>end_date_time</type>
</end_time>
```

start_date_time – The day and time the maintenance period will start. Start date time must be greater than the current time. Enter integer values for month, day, and year. Optionally enter hours, minutes, and seconds.

For example, the following start_date_time represents February 14, 2014 at 12:00:00 am.

```
<start_date_time>
  <month>2</month>
  <day>14</day>
  <year>2014</year>
  <timestamp>
    <hours></hours>
    <minutes></minutes>
    <seconds></seconds>
  </timestamp>
</start_date_time>
```

The following start_date_time represents February 14, 2014 at 3:30:22 pm.

```
<start_date_time>
  <month>2</month>
  <day>14</day>
  <year>2014</year>
  <timestamp>
    <hours>15</hours>
    <minutes>30</minutes>
    <seconds>22</seconds>
  </timestamp>
</start_date_time>
```
end_date_time – The date and time the maintenance period will end. Enter datetime in the type field. Enter integer values for month, day, and year. Optionally enter hours, minutes, and seconds. For example,

```xml
<end_time>
  <type>datetime</type>
  <end_date_time>
    <month>2</month>
    <day>21</day>
    <year>2014</year>
    <timestamp>
      <hours>15</hours>
      <minutes>30</minutes>
      <seconds>22</seconds>
    </timestamp>
    <timezone></timezone>
  </end_date_time>
  <duration>
    <hours></hours>
    <minutes></minutes>
    <seconds></seconds>
  </duration>
</end_time>
```

account_id – The integer id of an account.

recurrence_pattern – Leave blank for no recurrence. Enter daily, weekly, or monthly recurrence.

recurrence_period – Integer that indicates how often the recurrence pattern will occur. For example, for recurrence_patterns of daily, weekly, or monthly; recurrence_period = 3 is every 3 days, 3 weeks, or 3 months respectively; recurrence_period = 1 is daily, weekly, monthly respectively. Required if recurrence_instance is entered.

recurrence_days_of_the_week – Day of the week when recurrence will occur. If the schedule is weekly this can be a comma separated list for multiple days of the week. For example, “Tuesday,Thursday”.

recurrence_day_of_the_month – Integer day of the month when recurrence will occur. For example, day 2 of every month.
**recurrence_instance** – Only applicable for monthly recurrences. Valid values are 1, 2, 3, 4 or 5 (1st, 2nd, 3rd, 4th, or last). If this is used, recurrence_days_of_the_week should be specified as a single integer.

For example, the following specifies a maintenance period on the 4th Sunday of every month.

```
<recurrence_pattern>Monthly</recurrence_pattern>
<recurrence_period>1</recurrence_period>
<recurrence_days_of_the_week>Sunday</recurrence_days_of_the_week>
<recurrence_day_of_the_month></recurrence_day_of_the_month>
<recurrence_instance>4</recurrence_instance>
```

**recurrence_end_date_time** – The date and time when recurrence will end.

**Timezone** – The timezone for the schedule. Timezone must be one of the following:

- Etc/GMT+12
- Etc/GMT+11
- Pacific/Midway
- Pacific/Niue
- Pacific/Pago_Pago
- Pacific/Samoa
- US/Samoa
- America/Adak
- America/Atka
- Etc/GMT+10
- HST
- Pacific/Honolulu
- Pacific/Johnston
- Pacific/Rarotonga
- Pacific/Tahiti
- SystemV/HST10
- US/Aleutian
- US/Hawaii
- Pacific/Marquesas
- AST
- America/Anchorage
- America/Juneau
- America/Nome
- America/Sitka
- America/Yakutat
- Etc/GMT+9
- Pacific/Gambier
- SystemV/YST9
- SystemV/YST9YDT
- US/Alaska
- America/Dawson
- America/Ensenada
- America/Los_Angeles
- America/Metlakatla
- America/Santa_Isabel
- America/Tijuana
- America/Vancouver
- America/Whitehorse
- Canada/Pacific
- Canada/Yukon
- Etc/GMT+8
- Mexico/BajaNorte
- PST
- PST8PDT
- Pacific/Pitcairn
- SystemV/PST8
- SystemV/PST8PDT
- US/Pacific
- US/Pacific-New
- America/Boise
- America/Cambridge_Bay
- America/Chihuahua
- America/Creston
- America/Dawson_Creek
- America/Denver
- America/Edmonton
- America/Hermosillo
- America/Inuvik
- America/Mazatlan
- America/Ojinaga
- America/Phoenix
- America/Shiprock
- America/Yellowknife
- Canada/Mountain
- Etc/GMT+7
- MST
- MST7MDT
- Mexico/BajaSur
- Navajo
- PNT
- SystmV/MST7
- SystemV/MST7MDT
- US/Arizona
- US/Mountain
- America/Bahia_Banderas
- America/Belize
- America/Cancun
- America/Chicago
- America/Costa_Rica
- America/EI_Salvador
- America/Guatemala
- America/Indiana/Knox
- America/Indiana/Tell_City
- America/Knox_IN
- America/Managua
- America/Matamoros
- America/Menominee
- America/Merida
- America/Mexico_City
- America/Monterrey
- America/North_Dakota/Beulah
- America/North_Dakota/Center
- America/North_Dakota/New_Salem
- America/Rainy_River
- America/Regina
- America/Resolute
- America/Swift_Current
- America/Tegucigalpa
- America/Winnipeg
- CST
- CST6CDT
- Canada/Central
- Canada/East-Saskatchewan
- Canada/Saskatchewan
- Chile/EasterIsland
- Etc/GMT+6
- Mexico/General
- Pacific/Easter
- Pacific/Galapagos
- SystemV/CST6
- SystemV/CST6CDT
- US/Central
- US/Indiana-Starke
- America/Atikokan
- America/Bogota
- America/Cayman
- America/Coral_Harbour
- America/Detroit
- America/Fort_Wayne
- America/Grand_Turk
- America/Guayaquil
- America/Havana
- America/Indiana/Indianapolis
- America/Indiana/Marengo
- America/Indiana/Petersburg
- America/Indiana/Vevay
- America/Indiana/Vincennes
- America/Indiana/Winamac
- America/Indianapolis
- America/Iqaluit
- America/Jamaica
• America/Kentucky/Louisville
• America/Kentucky/Monticello
• America/Lima
• America/Louisville
• America/Montreal
• America/Nassau
• America/New_York
• America/Nipigon
• America/Panama
• America/Pangnirtung
• America/Port-au-Prince
• America/Thunder_Bay
• America/Toronto
• Canada/Eastern
• Cuba
• EST
• EST5EDT
• Etc/GMT+5
• IET
• Jamaica
• SystemV/EST5
• SystemV/EST5EDT
• US/East-Indiana
• US/Eastern
• US/Michigan
• America/Caracas
• America/Anguilla
• America/Antigua
• America/Argentina/San_Luis
• America/Aruba
• America/Asuncion
• America/Barbados
• America/Blanc-Sablon
• America/Boa_Vista
• America/Campo_Grande
• America/Cuiaba
• America/Curacao
• America/Dominica
• America/Eirunepe
• America/Glace_Bay
• America/Goose_Bay
• America/Grenada
• America/Guadeloupe
• America/Guyana
• America/Halifax
• America/Kralendijk
• America/La_Paz
• America/Lower_Princes
- America/Manaus
- America/Marigot
- America/Martinique
- America/Moncton
- America/Montserrat
- America/Port_of_Spain
- America/Porto_Acre
- America/Porto_Velho
- America/Puerto_Rico
- America/Rio_Branco
- America/Santiago
- America/Santo_Domingo
- America/St_Barthelemy
- America/St_Kitts
- America/St_Lucia
- America/St_Thomas
- America/St_Vincent
- America/Thule
- America/Tortola
- America/Virgin
- Antarctica/Palmer
- Atlantic/Bermuda
- Brazil/Acre
- Brazil/West
- Canada/Atlantic
- Chile/Continental
- Etc/GMT+4
- PRT
- SystemV/AST4
- SystemV/AST4ADT
- America/St_Johns
- CNT
- Canada/Newfoundland
- AGT
- America/Araguaina
- America/Argentina/Buenos_Aires
- America/Argentina/Catamarca
- America/Argentina/ComodRivadavia
- America/Argentina/Cordoba
- America/Argentina/Jujuy
- America/Argentina/La_Rioja
- America/Argentina/Mendoza
- America/Argentina/Rio_Gallegos
- America/Argentina/Salta
- America/Argentina/San_Juan
- America/Argentina/Tucuman
- America/Argentina/Ushuaia
- America/Bahia
• America/Belem
• America/Buenos_Aires
• America/Catamarca
• America/Cayenne
• America/Cordoba
• America/Fortaleza
• America/Godthab
• America/Jujuy
• America/Maceio
• America/Mendoza
• America/Miquelon
• America/Montevideo
• America/Paramaribo
• America/Recife
• America/Rosario
• America/Santarem
• America/Sao_Paulo
• Antarctica/Rothera
• Atlantic/Stanley
• BET
• Brazil/East
• Etc/GMT+3
• America/Noronha
• Atlantic/South_Georgia
• Brazil/DeNoronha
• Etc/GMT+2
• America/Scoresbysund
• Atlantic/Azores
• Atlantic/Cape_Verde
• Etc/GMT+1
• Africa/Abidjan
• Africa/Accra
• Africa/Bamako
• Africa/Belem
• Africa/Bissau
• Africa/Casablanca
• Africa/Conakry
• Africa/Dakar
• Africa/El_Aaiun
• Africa/Freetown
• Africa/Lome
• Africa/Monrovia
• Africa/Nouakchott
• Africa/Ouagadougou
• Africa/Sao_Tome
• Africa/Timbuktu
• America/Danmarkshavn
• Atlantic/Canary
• Atlantic/Faeroe
• Atlantic/Faroe
• Atlantic/Madeira
• Atlantic/Reykjavik
• Atlantic/St_Helena
• Eire
• Etc/GMT
• Etc/GMT+0
• Etc/GMT-0
• Etc/GMT0
• Etc/Greenwich
• Etc/UCT
• Etc/UTC
• Etc/Universal
• Etc/Zulu
• Europe/Belfast
• Europe/Dublin
• Europe/Guernsey
• Europe/Isle_of_Man
• Europe/Jersey
• Europe/Lisbon
• Europe/London
• GB
• GB-Eire
• GMT
• GMT0
• Greenwich
• Iceland
• Portugal
• UCT
• UTC
• Universal
• WET
• Zulu
• Africa/Algiers
• Africa/Bangui
• Africa/Brazzaville
• Africa/Ceuta
• Africa/Douala
• Africa/Kinshasa
• Africa/Lagos
• Africa/Libreville
• Africa/Luanda
• Africa/Malabo
• Africa/Ndjamena
• Africa/Niamey
• Africa/Porto-Novos
• Africa/Tunis
• Africa/Windhoek
• Arctic/Longyearbyen
• Atlantic/Jan_Mayen
• CET
• ECT
• Etc/GMT-1
• Europe/Amsterdam
• Europe/Andorra
• Europe/Belgrade
• Europe/Berlin
• Europe/Bratislava
• Europe/Brussels
• Europe/Budapest
• Europe/Copenhagen
• Europe/Gibraltar
• Europe/Ljubljana
• Europe/Luxembourg
• Europe/Madrid
• Europe/Malta
• Europe/Monaco
• Europe/Oslo
• Europe/Paris
• Europe/Podgorica
• Europe/Prague
• Europe/Rome
• Europe/San_Marino
• Europe/Sarajevo
• Europe/Skopje
• Europe/Stockholm
• Europe/Tirane
• Europe/Vaduz
• Europe/Vatican
• Europe/Vienna
• Europe/Warsaw
• Europe/Zagreb
• Europe/Zurich
• MET
• Poland
• ART
• Africa/Blantyre
• Africa/Bujumbura
• Africa/Cairo
• Africa/Gaborone
• Africa/Harare
• Africa/Johannesburg
• Africa/Kigali
• Africa/Lubumbashi
• Africa/Lusaka
• Africa/Maputo
• Africa/Maseru
• Africa/Mbabane
• Africa/Tripoli
• Asia/Amman
• Asia/Beirut
• Asia/Damascus
• Asia/Gaza
• Asia/Hebron
• Asia/Istanbul
• Asia/Jerusalem
• Asia/Nicosia
• Asia/Tel_Aviv
• CAT
• EET
• Egypt
• Etc/GMT-2
• Europe/Athens
• Europe/Bucharest
• Europe/Chisinau
• Europe/Helsinki
• Europe/Istanbul
• Europe/Kiev
• Europe/Mariehamn
• Europe/Nicosia
• Europe/Riga
• Europe/Simferopol
• Europe/Sofia
• Europe/Tallinn
• Europe/Tiraspol
• Europe/Uzhgorod
• Europe/Vilnius
• Europe/Zaporozhye
• Israel
• Libya
• Turkey
• Africa/Addis_Ababa
• Africa/Asmara
• Africa/Asmera
• Africa/Dar_es_Salaam
• Africa/Djibouti
• Africa/Juba
• Africa/Kampala
• Africa/Khartoum
• Africa/Mogadishu
• Africa/Nairobi
• Antarctica/Syowa
• Asia/Aden
- Asia/Baghdad
- Asia/Bahrain
- Asia/Kuwait
- Asia/Qatar
- Asia/Riyadh
- EAT
- Etc/GMT-3
- Europe/Kaliningrad
- Europe/Minsk
- Indian/Antananarivo
- Indian/Comoro
- Indian/Mayotte
- Asia/Riyadh87
- Asia/Riyadh88
- Asia/Riyadh89
- Mideast/Riyadh87
- Mideast/Riyadh88
- Mideast/Riyadh89
- Asia/Tehran
- Iran
- Asia/Baku
- Asia/Dubai
- Asia/Muscat
- Asia/Tbilisi
- Asia/Yerevan
- Etc/GMT-4
- Europe/Moscow
- Europe/Samara
- Europe/Volgograd
- Indian/Mahe
- Indian/Mauritius
- Indian/Reunion
- NET
- W-SU
- Asia/Kabul
- Antarctica/Mawson
- Asia/Aqtau
- Asia/Aqtobe
- Asia/Ashgabat
- Asia/Ashkhabad
- Asia/Dushanbe
- Asia/Karachi
- Asia/Oral
- Asia/Samarkand
- Asia/Tashkent
- Etc/GMT-5
- Indian/Kerguelen
- Indian/Maldives
- PLT
- Asia/Calcutta
- Asia/Colombo
- Asia/Kolkata
- IST
- Asia/Kathmandu
- Asia/Katmandu
- Antarctica/Vostok
- Asia/Almaty
- Asia/Bishkek
- Asia/Dacca
- Asia/Dhaka
- Asia/Qyzylorda
- Asia/Thimbu
- Asia/Thimphu
- Asia/Yekaterinburg
- BST
- Etc/GMT-6
- Indian/Chagos
- Asia/Rangoon
- Indian/Cocos
- Antarctica/Davis
- Asia/Bangkok
- Asia/Ho_Chi_Minh
- Asia/Hovd
- Asia/Jakarta
- Asia/Novokuznetsk
- Asia/Novosibirsk
- Asia/Omsk
- Asia/Phnom_Penh
- Asia/Pontianak
- Asia/Phnom Penh
- Asia/Saigon
- Asia/Vientiane
- Etc/GMT-7
- Indian/Christmas
- VST
- Antarctica/Casey
- Asia/Brunei
- Asia/Choibalsan
- Asia/Chongqing
- Asia/Chungking
- Asia/Harbin
- Asia/Hong_Kong
- Asia/Kashgar
- Asia/Krasnoyarsk
- Asia/Kuala_Lumpur
- Asia/Kuching
- Asia/Macao
• Asia/Macau
• Asia/Makassar
• Asia/Manila
• Asia/Shanghai
• Asia/Singapore
• Asia/Taipei
• Asia/U Jung_Pandang
• Asia/Ulaanbaatar
• Asia/Ulan_Bator
• Asia/Urumqi
• Australia/Perth
• Australia/West
• CTT
• Etc/GMT-8
• Hongkong
• PRC
• Singapore
• Australia/Eucla
• Asia/Dili
• Asia/Irkutsk
• Asia/Jayapura
• Asia/Pyongyang
• Asia/Seoul
• Asia/Tokyo
• Etc/GMT-9
• JST
• Japan
• Pacific/Palau
• ROK
• ACT
• Australia/Adelaide
• Australia/Broken_Hill
• Australia/Darwin
• Australia/North
• Australia/South
• Australia/Yancowinna
• AET
• Antarctica/DumontDUrville
• Asia/Yakutsk
• Australia/ACT
• Australia/Brisbane
• Australia/Canberra
• Australia/Currie
• Australia/Hobart
• Australia/Lindeman
• Australia/Melbourne
• Australia/NSW
• Australia/Queensland
- Australia/Sydney
- Australia/Tasmania
- Australia/Victoria
- Etc/GMT-10
- Pacific/Chuuk
- Pacific/Guam
- Pacific/Port_Moresby
- Pacific/Saipan
- Pacific/Truk
- Pacific/Yap
- Australia/LHI
- Australia/Lord_Howe
- Antarctica/Macquarie
- Asia/Sakhalin
- Asia/Vladivostok
- Etc/GMT-11
- Pacific/Efate
- Pacific/Guadalcanal
- Pacific/Kosrae
- Pacific/Noumea
- Pacific/Pohnpei
- Pacific/Ponape
- SST
- Pacific/Norfolk
- Antarctica/McMurdo
- Antarctica/South_Pole
- Asia/Anadyr
- Asia/Kamchatka
- Asia/Magadan
- Etc/GMT-12
- Kwajalein
- NST
- NZ
- Pacific/Auckland
- Pacific/Fiji
- Pacific/Funafuti
- Pacific/Kwajalein
- Pacific/Majuro
- Pacific/Nauru
- Pacific/Tarawa
- Pacific/Wake
- Pacific/Wallis
- NZ-CHAT
- Pacific/Chatham
- Etc/GMT-13
- MIT
- Pacific/Apia
- Pacific/Enderbury
- Pacific/Fakaofo
- Pacific/Tongatapu
- Etc/GMT-14
- Pacific/Kiritimati
Appendix B – License Information

The RESTful Web Services contain and utilize components from the Apache Software Foundation which are subject to the following terms.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.
1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not
remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.


You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.
You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections
2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.
This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

----------
NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble
The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.)

You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification"). Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
   a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
   b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
   c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

   a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

   c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.
4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.
This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License may
add an explicit geographical distribution limitation excluding those
countries, so that distribution is permitted only in or among countries
not thus excluded. In such case, this License incorporates the
limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time. Such new versions will
be similar in spirit to the present version, but may differ in detail
to address new problems or concerns.

Each version is given a distinguishing version number. If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Program does not specify a version
number of this License, you may choose any version ever published by
the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the
author to ask for permission. For software which is copyrighted by the
Free Software Foundation, write to the Free Software Foundation; we
sometimes make exceptions for this. Our decision will be guided by the
two goals of preserving the free status of all derivatives of our free
software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE
ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.
END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) year name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.
You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.
## Appendix C – Valid Values to Set Language for Liferay Users

<table>
<thead>
<tr>
<th>Language</th>
<th>Code</th>
<th>Region</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>en</td>
<td>en-US</td>
</tr>
<tr>
<td>German</td>
<td>de</td>
<td>de-DE</td>
</tr>
<tr>
<td>Spanish</td>
<td>es</td>
<td>es-ES</td>
</tr>
<tr>
<td>Japanese</td>
<td>ja</td>
<td>ja-JP</td>
</tr>
<tr>
<td>Korean</td>
<td>ko</td>
<td>ko-KR</td>
</tr>
<tr>
<td>Portuguese</td>
<td>pt</td>
<td>pt-BR</td>
</tr>
<tr>
<td>Chinese</td>
<td>zh</td>
<td>zh-CN</td>
</tr>
</tbody>
</table>