**Eureka Discovery Registry Server REST operations**

Following are the REST operations available for non-java applications to use Eureka.

**The appID** is the name of the application and **instanceID** is the unique id associated with the instance. In AWS cloud, instanceID is the **instance id** of the instance and in other data centers, it is the **hostname** of the instance.

For JSON/XML, the content types supplied must be **application/xml** or **application/json**.

|  |  |  |
| --- | --- | --- |
| **Operation** | **HTTP action** | **Description** |
| Register new application instance | POST /eureka/v2/apps/**appID** | Input: JSON/XML payload HTTP Code: 204 on success |
| De-register application instance | DELETE /eureka/v2/apps/**appID**/**instanceID** | HTTP Code: 200 on success |
| Send application instance heartbeat | PUT /eureka/v2/apps/**appID**/**instanceID** | HTTP Code: \* 200 on success \* 404 if **instanceID** doesn’t exist |
| Query for all instances | GET /eureka/v2/apps | HTTP Code: 200 on success Output: JSON/XML |
| Query for all **appID** instances | GET /eureka/v2/apps/**appID** | HTTP Code: 200 on success Output: JSON/XML |
| Query for a specific **appID**/**instanceID** | GET /eureka/v2/apps/**appID**/**instanceID** | HTTP Code: 200 on success Output: JSON/XML |
| Query for a specific **instanceID** | GET /eureka/v2/instances/**instanceID** | HTTP Code: 200 on success Output: JSON/XML |
| Take instance out of service | PUT /eureka/v2/apps/**appID**/**instanceID**/status?value=OUT\_OF\_SERVICE | HTTP Code: \* 200 on success \* 500 on failure |
| Move instance back into service (remove override) | DELETE /eureka/v2/apps/**appID**/**instanceID**/status?value=UP (The value=UP is optional, it is used as a suggestion for the fallback status due to removal of the override) | HTTP Code: \* 200 on success \* 500 on failure |
| Update metadata | PUT /eureka/v2/apps/**appID**/**instanceID**/metadata?key=value | HTTP Code: \* 200 on success \* 500 on failure |
| Query for all instances under a particular **vip address** | GET /eureka/v2/vips/**vipAddress** | \* HTTP Code: 200 on success Output: JSON/XML  \* 404 if the **vipAddress** does not exist. |
| Query for all instances under a particular **secure vip address** | GET /eureka/v2/svips/**svipAddress** | \* HTTP Code: 200 on success Output: JSON/XML  \* 404 if the **svipAddress** does not exist. |

**REGISTER**

When you register, you’ll need to post an XML (or JSON) body that conforms to this XSD:

<?xml version="1.0" encoding="UTF-8"?>

<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" attributeFormDefault="unqualified">

    <xsd:element name="instance">

        <xsd:complexType>

            <xsd:all>

                <!-- hostName in ec2 should be the public dns name, within ec2 public dns name will

                     always resolve to its private IP -->

                <xsd:element name="hostName" type="xsd:string" />

                <xsd:element name="app" type="xsd:string" />

                <xsd:element name="ipAddr" type="xsd:string" />

  <xsd:element name="vipAddress" type="xsd:string" />

  <xsd:element name="secureVipAddress" type="xsd:string" />

                <xsd:element name="status" type="statusType" />

                <xsd:element name="port" type="xsd:positiveInteger" minOccurs="0" />

                <xsd:element name="securePort" type="xsd:positiveInteger" />

<xsd:element name="homePageUrl" type="xsd:string" />

<xsd:element name="statusPageUrl" type="xsd:string" />

<xsd:element name="healthCheckUrl" type="xsd:string" />

              <xsd:element ref="dataCenterInfo" minOccurs="1" maxOccurs="1" />

                <!-- optional -->

                <xsd:element ref="leaseInfo" minOccurs="0"/>

                <!-- optional app specific metadata -->

                <xsd:element name="metadata" type="appMetadataType" minOccurs="0" />

            </xsd:all>

        </xsd:complexType>

    </xsd:element>

    <xsd:element name="dataCenterInfo">

        <xsd:complexType>

             <xsd:all>

                 <xsd:element name="name" type="dcNameType" />

                 <!-- metadata is only required if name is Amazon -->

                 <xsd:element name="metadata" type="amazonMetdataType" minOccurs="0"/>

             </xsd:all>

        </xsd:complexType>

    </xsd:element>

    <xsd:element name="leaseInfo">

        <xsd:complexType>

            <xsd:all>

                <!-- (optional) if you want to change the length of lease - default if 90 secs -->

                <xsd:element name="evictionDurationInSecs" minOccurs="0"  type="xsd:positiveInteger"/>

            </xsd:all>

        </xsd:complexType>

    </xsd:element>

    <xsd:simpleType name="dcNameType">

        <!-- Restricting the values to a set of value using 'enumeration' -->

        <xsd:restriction base = "xsd:string">

            <xsd:enumeration value = "MyOwn"/>

            <xsd:enumeration value = "Amazon"/>

        </xsd:restriction>

    </xsd:simpleType>

    <xsd:simpleType name="statusType">

        <!-- Restricting the values to a set of value using 'enumeration' -->

        <xsd:restriction base = "xsd:string">

            <xsd:enumeration value = "UP"/>

            <xsd:enumeration value = "DOWN"/>

            <xsd:enumeration value = "STARTING"/>

<xsd:enumeration value = "OUT\_OF\_SERVICE"/>

<xsd:enumeration value = "UNKNOWN"/>

</xsd:restriction>

    </xsd:simpleType>

    <xsd:complexType name="amazonMetdataType">

        <!-- From <a class="jive-link-external-small" href="http://docs.amazonwebservices.com/AWSEC2/latest/DeveloperGuide/index.html?AESDG-chapter-instancedata.html" target="\_blank">http://docs.amazonwebservices.com/AWSEC2/latest/DeveloperGuide/index.html?AESDG-chapter-instancedata.html</a> -->

        <xsd:all>

            <xsd:element name="ami-launch-index" type="xsd:string" />

            <xsd:element name="local-hostname" type="xsd:string" />

            <xsd:element name="availability-zone" type="xsd:string" />

            <xsd:element name="instance-id" type="xsd:string" />

            <xsd:element name="public-ipv4" type="xsd:string" />

            <xsd:element name="public-hostname" type="xsd:string" />

            <xsd:element name="ami-manifest-path" type="xsd:string" />

            <xsd:element name="local-ipv4" type="xsd:string" />

            <xsd:element name="hostname" type="xsd:string"/>

            <xsd:element name="ami-id" type="xsd:string" />

            <xsd:element name="instance-type" type="xsd:string" />

        </xsd:all>

    </xsd:complexType>

    <xsd:complexType name="appMetadataType">

        <xsd:sequence>

            <!-- this is optional application specific name, value metadata -->

            <xsd:any minOccurs="0" maxOccurs="unbounded" processContents="skip"/>

        </xsd:sequence>

    </xsd:complexType>

</xsd:schema>

**RENEW**

Example : PUT /eureka/v2/apps/MYAPP/i-6589ef6

Response:  
Status: 200 (on success)  
404 (eureka doesn’t know about you, Register yourself first)  
500 (failure)

**CANCEL**

(If Eureka doesn’t get heartbeats from the service node within the evictionDurationInSecs, then the node will get automatically de-registered )

Example : DELETE /eureka/v2/apps/MYAPP/i-6589ef6

Response:  
Status: 200 (on success)  
500 (failure)

-----------------