使用Glance的元数据定义目录公共API

更新日期：'THU MAR 1 07:26:57 2018 , COMMIT 968F4AE'

Glance服务为供应商，管理员，服务和用户托管一个通用API，以有意义地定义可用的键/值对和标记元数据。其目的是为OpenStack用户的工件，服务和项目启用更好的元数据协作。

这涉及可用于不同类型资源（图像，工件，体积，风味，聚合等）的可用元数据的定义。一个定义包括属性类型，它的关键，它的描述以及它的约束。该目录不会存储特定实例属性的值。

例如，核心数量的虚拟CPU拓扑属性的定义将包括要使用的密钥，说明和值约束，例如要求它是整数。因此，可能通过Horizon的用户将能够搜索此目录以列出可添加到风味或图像的可用属性。他们将看到列表中的虚拟CPU拓扑属性，并知道它必须是一个整数。在Horizon示例中，当用户添加属性时，其密钥和值将存储在拥有该资源的服务中（Nova for flavor和Glance for images）。

图表：https (https://wiki.openstack.org/w/images/b/bb/Glance-Metadata-API.png)：//wiki.openstack.org/w/images/b/bb/Glance-Metadata-API.png
(https://wiki.openstack.org/w/images/b/bb/Glance-Metadata-API.png)

概览元数据定义目录实施从API版本v2开始。

认证¶

一警依赖于Keystone和OpenStack Identity API来处理客户端的身份验证。您必须从Keystone获取身份验证令牌，并通过**X-Auth-Token**头部将其与所有API请求一起发送至Glance。Glance会通知Keystone验证令牌有效性并获取您的身份凭证。

有关与Keystone (../admin/authentication.html#authentication)集成的更多信息，请参阅使用Keystone (../admin/authentication.html#authentication)进行验证 (../admin/authentication.html#authentication)。

使用2.x版¶

出于示例的目的，假定**http://glance.openstack.example.org**在默认端口80上的URL上运行一个Glance API服务器。

列出可用命名空间¶

我们希望看到经过身份验证的用户有权访问的可用名称空间的列表。这包括用户拥有的命名空间，与用户共享的命名空间和公共命名空间。

我们发出一个**GET**请求来**http://glance.openstack.example.org/v2/metadefs/namespaces** 检索这个可用名称空间的列表。数据以下格式作为JSON编码映射返回：

```
{
    "命名空间"：[
        {
            "命名空间"："myNameSpace对象"，
            "DISPLAY_NAME"："我的用户友好的名称空间"，
            "说明"："我的说明"，
            "可见性"："公共"，
            "保护"：真实的，
            "所有者"："测试所有者"，
            "自我"："/ v2 / metadefs / namespaces / MyNamespace"，
            "架构"："/ v2 / schemas / metadefs /命名空间"，
            "created_at"："2014-08-28T17：13：06Z"，
            "updated_at"："2014-08-28T17：13：06Z"，
            "resource_type_associations"：[
                {
                    "name"："OS :: Nova :: Aggregate"，
                    "created_at"："2014-08-28T17：13：06Z "，
                    "updated_at"："2014-08-28T17：13：06Z"
                },
                {
                    "name"："OS :: Nova :: Flavor"，
                    "prefix"："aggregate_instance_extra_specs："，
                    "created_at"："2014- 08-28T17：13：06Z"，
                    "updated_at"："2014-08-28T17：13：06Z"
                }
            ]
        }
    ],
    "第一"："/ v2 / metadefs / namespaces？sort_key = created_at＆sort_dir = asc"，
    "schema"："/ v2 / schemas / metadefs / namespaces"
}
```

> **⊘ 注意**
>
> 列表名称空间仅显示每个名称空间的摘要，包括计数和资源类型关联。包括所有对象定义，属性定义等在内的详细响应将仅在每个单独的GET命名空间请求中可用。

过滤命名空间列表¶

`GET /v2/metadefs/namespaces`请求采用查询参数来过滤返回的名称空间列表。以下列表详细说明这些查询参数。

- `resource_types=RESOURCE_TYPES`

  筛选`resource_types`逗号分隔列表中的名称空间`RESOURCE_TYPES`。

GET资源还接受其他查询参数：

- `sort_key=KEY`

  Results will be ordered by the specified sort attribute `KEY`. Accepted values include `namespace`, `created_at` (default) and `updated_at`.

- `sort_dir=DIR`

  Results will be sorted in the direction `DIR`. Accepted values are `asc` for ascending or `desc` (default) for descending.

- `marker=NAMESPACE`

  A namespace identifier marker may be specified. When present only namespaces which occur after the identifier `NAMESPACE` will be listed, i.e. the namespaces which have a *sort_key* later than that of the marker `NAMESPACE` in the *sort_dir* direction.

- `limit=LIMIT`

  When present the maximum number of results returned will not exceed `LIMIT`.

> **⊘ Note**
>
> If the specified `LIMIT` exceeds the operator defined limit (api_limit_max) then the number of results returned may be less than `LIMIT`.

- `visibility=PUBLIC`

  An admin user may use the *visibility* parameter to control which results are returned (PRIVATE or PUBLIC).

Retrieve Namespace¶

We want to see a more detailed information about a namespace that the authenticated user has access to. The detail includes the properties, objects, and resource type associations.

We issue a `GET` request to `http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}` to retrieve the namespace details. The data is returned as a JSON-encoded mapping in the following format:

```
{
    "namespace": "MyNamespace",
    "display_name": "My User Friendly Namespace",
    "description": "My description",
    "visibility": "public",
    "protected": true,
    "owner": "The Test Owner",
    "schema": "/v2/schemas/metadefs/namespace",
    "resource_type_associations": [
        {
            "name": "OS::Glance::Image",
            "prefix": "hw_",
            "created_at": "2014-08-28T17:13:06Z",
            "updated_at": "2014-08-28T17:13:06Z"
        },
        {
            "name": "OS::Cinder::Volume",
            "prefix": "hw_",
            "properties_target": "image",
            "created_at": "2014-08-28T17:13:06Z",
            "updated_at": "2014-08-28T17:13:06Z"
        },
        {
            "name": "OS::Nova::Flavor",
            "prefix": "filter1:",
            "created_at": "2014-08-28T17:13:06Z",
            "updated_at": "2014-08-28T17:13:06Z"
        }
    ],
    "properties": {
        "nsprop1": {
            "title": "My namespace property1",
            "description": "More info here",
            "type": "boolean",
            "default": true
        },
        "nsprop2": {
            "title": "My namespace property2",
            "description": "More info here",
            "type": "string",
            "default": "value1"
        }
    },
    "objects": [
        {
            "name": "object1",
            "description": "my-description",
            "self": "/v2/metadefs/namespaces/MyNamespace/objects/object1",
            "schema": "/v2/schemas/metadefs/object",
            "created_at": "2014-08-28T17:13:06Z",
            "updated_at": "2014-08-28T17:13:06Z",
            "required": [],
            "properties": {
                "prop1": {
                    "title": "My object1 property1",
                    "description": "More info here",
                    "type": "array",
                    "items": {
                        "type": "string"
                    }
                }
            }
        },
        {
            "name": "object2",
            "description": "my-description",
            "self": "/v2/metadefs/namespaces/MyNamespace/objects/object2",
            "schema": "/v2/schemas/metadefs/object",
            "created_at": "2014-08-28T17:13:06Z",
            "updated_at": "2014-08-28T17:13:06Z",
            "properties": {
                "prop1": {
                    "title": "My object2 property1",
                    "description": "More info here",
                    "type": "integer",
                    "default": 20
                }
            }
        }
    ]
}
```

Retrieve available Resource Types¶

We want to see the list of all resource types that are available in Glance

We issue a **GET** request to **http://glance.openstack.example.org/v2/metadefs/resource_types** to retrieve all resource types.

The data is returned as a JSON-encoded mapping in the following format:

```
{
  "resource_types": [
    {
      "created_at": "2014-08-28T17:13:04Z",
      "name": "OS::Glance::Image",
      "updated_at": "2014-08-28T17:13:04Z"
    },
    {
      "created_at": "2014-08-28T17:13:04Z",
      "name": "OS::Cinder::Volume",
      "updated_at": "2014-08-28T17:13:04Z"
    },
    {
      "created_at": "2014-08-28T17:13:04Z",
      "name": "OS::Nova::Flavor",
      "updated_at": "2014-08-28T17:13:04Z"
    },
    {
      "created_at": "2014-08-28T17:13:04Z",
      "name": "OS::Nova::Aggregate",
      "updated_at": "2014-08-28T17:13:04Z"
    },
    {
      "created_at": "2014-08-28T17:13:04Z",
      "name": "OS::Nova::Server",
      "updated_at": "2014-08-28T17:13:04Z"
    }
  ]
}
```

Retrieve Resource Types associated with a Namespace¶

We want to see the list of resource types that are associated for a specific namespace

We issue a **GET** request to **http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}/resource_types** to retrieve resource types.

The data is returned as a JSON-encoded mapping in the following format:

```
{
  "resource_type_associations" : [
    {
      "name" : "OS::Glance::Image",
      "prefix" : "hw_",
      "created_at": "2014-08-28T17:13:04Z",
      "updated_at": "2014-08-28T17:13:04Z"
    },
    {
      "name" :"OS::Cinder::Volume",
      "prefix" : "hw_",
      "properties_target" : "image",
      "created_at": "2014-08-28T17:13:04Z",
      "updated_at": "2014-08-28T17:13:04Z"
    },
    {
      "name" : "OS::Nova::Flavor",
      "prefix" : "hw:",
      "created_at": "2014-08-28T17:13:04Z",
      "updated_at": "2014-08-28T17:13:04Z"
    }
  ]
}
```

Add Namespace¶

We want to create a new namespace that can contain the properties, objects, etc.

We issue a **POST** request to add an namespace to Glance:

```
POST http://glance.openstack.example.org/v2/metadefs/namespaces/
```

The input data is an JSON-encoded mapping in the following format:

```
{
  "namespace": "MyNamespace",
  "display_name": "My User Friendly Namespace",
  "description": "My description",
  "visibility": "public",
  "protected": true
}
```

> ✔ Note
>
> Optionally properties, objects and resource type associations could be added in the same input. See GET Namespace output above(input will be similar).

Update Namespace¶

We want to update an existing namespace

We issue a **PUT** request to update an namespace to Glance:

```
PUT http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}
```

The input data is similar to Add Namespace

Delete Namespace¶

We want to delete an existing namespace including all its objects, properties etc.

We issue a **DELETE** request to delete an namespace to Glance:

```
DELETE http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}
```

Associate Resource Type with Namespace¶

We want to associate a resource type with an existing namespace

We issue a **POST** request to associate resource type to Glance:

```
POST http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}/resource_types
```

The input data is an JSON-encoded mapping in the following format:

```
{
        "name" :"OS::Cinder::Volume",
        "prefix" : "hw_",
        "properties_target" : "image",
        "created_at": "2014-08-28T17:13:04Z",
        "updated_at": "2014-08-28T17:13:04Z"
}
```

Remove Resource Type associated with a Namespace¶

We want to de-associate namespace from a resource type

We issue a **DELETE** request to de-associate namespace resource type to Glance:

```
DELETE http://glance.openstack.example.org/v2//metadefs/namespaces/{namespace}/resource_types/{resource_type}
```

List Objects in Namespace¶

We want to see the list of meta definition objects in a specific namespace

We issue a **GET** request to **http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}/objects** to retrieve objects.

The data is returned as a JSON-encoded mapping in the following format:

```
{
    "objects": [
    {
        "name": "object1",
        "description": "my-description",
        "self": "/v2/metadefs/namespaces/MyNamespace/objects/object1",
        "schema": "/v2/schemas/metadefs/object",
        "created_at": "2014-08-28T17:13:06Z",
        "updated_at": "2014-08-28T17:13:06Z",
        "required": [],
        "properties": {
            "prop1": {
                "title": "My object1 property1",
                "description": "More info here",
                "type": "array",
                "items": {
                    "type": "string"
                }
            }
        }
    },
    {
        "name": "object2",
        "description": "my-description",
        "self": "/v2/metadefs/namespaces/MyNamespace/objects/object2",
        "schema": "/v2/schemas/metadefs/object",
        "created_at": "2014-08-28T17:13:06Z",
        "updated_at": "2014-08-28T17:13:06Z",
        "properties": {
            "prop1": {
                "title": "My object2 property1",
                "description": "More info here",
                "type": "integer",
                "default": 20
            }
        }
    }
    ],
    "schema": "/v2/schemas/metadefs/objects"
}
```

Add object in a specific namespace¶

We want to create a new object which can group the properties

We issue a **POST** request to add object to a namespace in Glance:

```
POST http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}/objects
```

The input data is an JSON-encoded mapping in the following format:

```
{
  "name": "StorageQOS",
  "description": "Our available storage QOS.",
  "required": [
      "minIOPS"
  ],
  "properties": {
      "minIOPS": {
          "type": "integer",
          "description": "The minimum IOPs required",
          "default": 100,
          "minimum": 100,
          "maximum": 30000369
      },
      "burstIOPS": {
          "type": "integer",
          "description": "The expected burst IOPs",
          "default": 1000,
          "minimum": 100,
          "maximum": 30000377
      }
  }
}
```

Update Object in a specific namespace¶

We want to update an existing object

We issue a **PUT** request to update an object to Glance:

```
PUT http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}/objects/{object_name}
```

The input data is similar to Add Object

Delete Object in a specific namespace¶

We want to delete an existing object.

We issue a **DELETE** request to delete object in a namespace to Glance:

```
DELETE http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}/objects/{object_name}
```

Add property definition in a specific namespace¶

We want to create a new property definition in a namespace

We issue a **POST** request to add property definition to a namespace in Glance:

```
POST http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}/properties
```

The input data is an JSON-encoded mapping in the following format:

```
{
    "name": "hypervisor_type",
    "title" : "Hypervisor",
    "type": "array",
    "description": "The type of hypervisor required",
    "items": {
        "type": "string",
        "enum": [
            "hyperv",
            "qemu",
            "kvm"
        ]
    }
}
```

Update property definition in a specific namespace¶

We want to update an existing object

We issue a **PUT** request to update an property definition in a namespace to Glance:

```
PUT http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}/properties/{property_name}
```

The input data is similar to Add property definition

Delete property definition in a specific namespace¶

We want to delete an existing object.

We issue a **DELETE** request to delete property definition in a namespace to Glance:

```
DELETE http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}/properties/{property_name}
```

API Message Localization¶

Glance supports HTTP message localization. For example, an HTTP client can receive API messages in Chinese even if the locale language of the server is English.

How to use it¶

To receive localized API messages, the HTTP client needs to specify the **Accept-Language** header to indicate the language to use to translate the message. For more info about Accept-Language, please refer http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html (http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html)

A typical curl API request will be like below:

```
curl -i -X GET -H 'Accept-Language: zh' -H 'Content-Type: application/json'
http://glance.openstack.example.org/v2/metadefs/namespaces/{namespace}
```

Then the response will be like the following:

```
HTTP/1.1 404 Not Found
Content-Length: 234
Content-Type: text/html; charset=UTF-8
X-Openstack-Request-Id: req-54d403a0-064e-4544-8faf-4aeef086f45a
Date: Sat, 22 Feb 2014 06:26:26 GMT

<html>
<head>
<title>404 Not Found</title>
</head>
<body>
<h1>404 Not Found</h1>
&#25214;&#19981;&#21040;&#20219;&#20309;&#20855;&#26377;&#26631;&#35782; aaa &#30340;&#26144;&#20687;<br /><br />
</body>
</html>
```

❶ Note

确保目标Glance服务器上的/ usr / share / locale-langpack /下有语言包。

《 (glanceclient.html) 》 (signature.html) 🐞 (https://bugs.launchpad.net/glance/+filebug?field.title=Using%20Glance%E2%80%99s%20Metadata%20Definitions%20Catalog%20Public%20APIs%20in%20glance&field.comment=%0A%0A%0AThis bug tracker is for errors with the documentation, use the following as a template and remove or add fields as you see fit. Convert [ ] into [x] to check boxes:%0A%0A- [ ] This doc is inaccurate in this way: _____%0A- [ ] This is a doc addition request.%0A- [ ] I have a fix to the document that I can paste below including example: input and output. %0A%0AIf you have a troubleshooting or support issue, use the following resources:%0A%0A - Ask OpenStack: http://ask.openstack.org%0A - The mailing list: http://lists.openstack.org%0A - IRC: 'openstack' channel on Freenode%0A%0A----------------------------------- %0ARelease:%2016.0.1.dev1%20on%20'Thu%20Mar%201%2007:26:57%202018,%20commit%20968f4ae'%0ASHA:%20968f4ae9ce244d9372cb3e8f45acea9d557f317d%0ASo https://docs.openstack.org/glance/queens/user/glancemetadefcatalogapi.html&field.tags=)

更新日期：'THU MAR 1 07:26:57 2018，COMMIT 968F4AE'

🐞 发现错误？报告错误 (HTTPS://BUGS.LAUNCHPAD.NET/GLANCE/+FILEBUG?
FIELD.TITLE=USING%20GLANCE%E2%80%99S%20METADATA%20DEFINITIONS%20CATALOG%20PUBLIC%20APIS%20IN%20GLANCE&FIELD.COMMENT=%0A%0A%0ATHIS BUG TRACKER IS FOR ERRORS WITH THE DOCUMENTATION, USE THE FOLLOWING AS A TEMPLATE AND REMOVE OR ADD FIELDS AS YOU SEE FIT. CONVERT [ ] INTO [X] TO CHECK BOXES:%0A%0A- [ ] THIS DOC IS INACCURATE IN THIS WAY: _____%0A- [ ] THIS IS A DOC ADDITION REQUEST.%0A- [ ] I HAVE A FIX TO THE DOCUMENT THAT I CAN PASTE BELOW INCLUDING EXAMPLE: INPUT AND OUTPUT. %0A%0AIF YOU HAVE A TROUBLESHOOTING OR SUPPORT ISSUE, USE THE FOLLOWING RESOURCES:%0A%0A - ASK OPENSTACK: HTTP://ASK.OPENSTACK.ORG%0A - THE MAILING LIST: HTTP://LISTS.OPENSTACK.ORG%0A - IRC: 'OPENSTACK' CHANNEL ON FREENODE%0A%0A----------------------------------- %0ARELEASE:%2016.0.1.DEV1%20ON%20'THU%20MAR%201%2007:26:57%202018,%20COMMIT%20968F4AE'%0ASHA:%20968F4AE9CE244D9372CB3E8F45ACEA9D557F317D%0ASOURCE:%20HTTPS:// HTTPS://DOCS.OPENSTACK.ORG/GLANCE/QUEENS/USER/GLANCEMETADEFCATALOGAPI.HTML&FIELD.TAGS=)

❓ 问题吗？ (HTTP://ASK.OPENSTACK.ORG)

⬅

OpenStack文档 ▾

一瞥16.0.1.dev1
(../index.html)

页面内容

OpenStack的

- 项目 (http://openstack.org/projects/)
- OpenStack安全 (http://openstack.org/projects/openstack-security/)
- 常见问题 (http://openstack.org/projects/openstack-faq/)
- 博客 (http://openstack.org/blog/)
- 新闻 (http://openstack.org/news/)

社区

- 用户组 (http://openstack.org/community/)
- 活动 (http://openstack.org/community/events/)
- 工作 (http://openstack.org/community/jobs/)
- 公司 (http://openstack.org/foundation/companies/)
- 有助于 (http://docs.openstack.org/infra/manual/developers.html)

文档

- OpenStack手册 (http://docs.openstack.org)
- 入门 (http://openstack.org/software/start/)
- API文档 (http://developer.openstack.org)
- 维基 (https://wiki.openstack.org)

品牌与法律

- 标志和指南 (http://openstack.org/brand/)
- 商标政策 (http://openstack.org/brand/openstack-trademark-policy/)
- 隐私政策 (http://openstack.org/privacy/)
- OpenStack CLA (https://wiki.openstack.org/wiki/How_To_Contribute#Contributor_License_Agreement)

保持联系

(https://t(whittpes://o/whitteprs/o/faw/t/eblonoadlsk/hookw/dompnosntaacdke)pcamny/user/OppStackStackFoundation)