layout: page title: Block Storage API title_section: The API specifications for HP Block Storage service. description: "Block Storage API Specifications" group: apispec

{% include JB/setup %}

1. Overview

Brief introduction and overview of the service and its intended use.

The Bock team is developing the Block Storage Service for Nova. The service provides Virtual Block devices for virtual machines executing in Nova. The capability is derived form the original DM-Cache capability implemented as part of the Service Utility Platform (SUP). Bock volumes can store boot images, user data or both. We will target Nova-Diablo for the initial release of Bock. In the Nova architecture Bock is implemented as a set of nova volume driver classes. These classes provide a set of methods that allow different types of storage plugins.

1.1 API Maturity Level

State the maturity level in which the API is in currently, based on the pre-defined stages i.e. Alpha, Beta, Public (Current), Deprecated, GA (Release to General Availability, SLAs defined. The versions schema, status field, supports an enumeration of ALPHA, BETA, CURRENT and DEPRECATED. The versions->status field should correspond to the Maturity Level for the API, i.e. ALPHA for Alpha stage, BETA for Beta stage, CURRENT for Public and GA stages, DEPRECATED for all other versions of the API that are not supported anymore.

Maturity Level: Beta stage - Bock is currently in restricted private beta in AW2/AZ2

Version API Status: BETA

2. Architecture View

Example info. and documentation for Bock is available on the Bock Wiki page.

2.1 Overview

References to architectural details of the service.

2.2 Conceptual/Logical Architecture View

Describe the logical components of the system and their responsibilities

2.3 Infrastructure Architecture View

Describe how the API fits into the overall HPCS Infrastructure

2.4 Entity Relationship Diagram

Describe the relationships between the various entities (resources) involved in the API

3. Account-level View

Describe the relationship of the API with respect to the accounts, groups, tenants, regions, availability zones etc.

3.1 Accounts

Describe the structure of the user accounts, groups and tenants. Currently this might be described separately in context of Control Services, but in future each service API needs to state their usage. In future CS might support complex group hierarchies, enterprise account structures while there maybe a phased adoption by individual service APIs

3.2 Regions and Availability Zones

Describe the availability of the service API in different regions and availability zones. State plans for future expansion as well.

Region(s): region-a

Availability Zone(s): az-1, az-2, az-3

Future Expansion: region-b

3.3 Service Catalog

Describe if the service API is exposed via the service catalog. Reference the fragment of the service catalog showing the structure.

The Block Storage is exposed in the service catalog, as shown in the following fragment:

```
{
    "name": "Block Storage",
    "type": "volume",
    "endpoints": [
      {
        "tenantId": "{tenant_id}",
        "publicURL": "https:\/\/az-1.region-a.geo-1.compute.hpcloudsvc.com\/v1.1\/{tenant_id
        "region": "az-1.region-a.geo-1",
        "versionId": "1.1",
        "versionInfo": "https:\/\/az-1.region-a.geo-1.compute.hpcloudsvc.com\/v1.1\/",
        "versionList": "https:\/\/az-1.region-a.geo-1.compute.hpcloudsvc.com"
      },
      {
        "tenantId": "{tenant_id}",
        "publicURL": "https:\/\/az-3.region-a.geo-1.compute.hpcloudsvc.com\/v1.1\/{tenant_id
        "region": "az-3.region-a.geo-1",
        "versionId": "1.1",
        "versionInfo": "https:\/\/az-3.region-a.geo-1.compute.hpcloudsvc.com\/v1.1\/",
        "versionList": "https:\/\/az-3.region-a.geo-1.compute.hpcloudsvc.com"
      },
      {
        "tenantId": "{tenant_id}",
        "publicURL": "https:\/\/az-2.region-a.geo-1.compute.hpcloudsvc.com\/v1.1\/{tenant_id
        "region": "az-2.region-a.geo-1",
        "versionId": "1.1",
        "versionInfo": "https:\/\/az-2.region-a.geo-1.compute.hpcloudsvc.com\/v1.1\/",
        "versionList": "https:\/\/az-2.region-a.geo-1.compute.hpcloudsvc.com"
      }
    ]
}
```

4. REST API Specifications

Describe the API specifications, namely the API operations, and its details, documenting the naming conventions, request and response formats, media type support, status codes, error conditions, rate limits, quota limits, and specific business rules.

4.1 Service API Operations

Host: https://az-1.region-a.geo-1.compute.hpcloudsvc.com

BaseURI: {Host}/v1.1/{tenant_id}

Admin URI: N/A

Resource	Operation	HTTP Method	Path	JSON/XML?	Privilege Level

Versions	Version information	GET	{Host}/v1.1/	Y/**N**	
	Version list	GET	{Host}/	Y/**N**	
Volumes	List volumes	GET	{BaseUri}/os-volumes	Y/**N**	
	Create new volume	POST	{BaseUri}/os-volumes	Y/**N**	
	Get volume details	GET	{BaseUri}/os-volumes/{volume_id}	Y/**N**	
	Delete volume	GET	{BaseUri}/os-volumes/{volume_id}	Y/**N**	
Servers	List attached volumes	GET	{BaseUri}/servers/{server_id}/os-volume_attachments	Y/**N**	
	Attach a volume	POST	{BaseUri}/servers/{server_id}/os-volume_attachments/{volume_id}	Y/**N**	
	Detach a volume	DELETE	{BaseUri}/servers/{server_id}/os-volume_attachments/{volume_id}	Y/**N**	
Snapshots	List volume snapshots	GET	{BaseUri}/os-snapshots	Y/**N**	
	Create new volume snapshot	POST	{BaseUri}/os-snapshots	Y/**N**	
	Get volume snapshot details	GET	{BaseUri}/os-snapshots/{snapshot_id}	Y/**N**	
	Delete volume snapshot	DELETE	{BaseUri}/os-snapshots/{snapshot_id}	Y/**N**	

4.2 Common Request Headers

List the common response headers i.e. X-Auth-Token, Content-Type, Content-Length, Date etc.

4.3 Common Response Headers

*List the common response headers i.e. Content-Type, Content-Length, Connection, Date, ETag, Server, etc. *

4.4 Service API Operation Details

The following section, enumerates each resource and describes each of its API calls as listed in the Service API Operations section, documenting the naming conventions, request and response formats, status codes, error conditions, rate limits, quota limits, and specific business rules.

4.4.1 Versions

Describe versions and what information they provide.

Status Lifecycle

N/A

Rate Limits

N/A

Quota Limits

N/A

Business Rules

None.

4.4.1.1 Get Version Info
GET /v1.1/

Lists the information about the current version of the API, like media types (json and xml), links to the public endpoint url, links to the WADL (if any), and links to the API documentation.

Request Data

URL Parameters

None.

Data Parameters

This call does not require a request body.

Success Response

Status Code

200 - OK

Response Data

JSON

```json
{
    "version":{
        "status":"CURRENT",
        "updated":"2011-01-21T11:33:21Z",
        "media-types":[
            {
                "base":"application/xml",
                "type":"application/vnd.openstack.compute-v1.1+xml"
            },
            {
                "base":"application/json",
                "type":"application/vnd.openstack.compute-v1.1+json"
            }
        ],
        "id":"v1.1",
        "links":[
            {
                "href":"https://az-1.region-a.geo-1.compute.hpcloudsvc.com/v1.1/",
                "rel":"self"
            },
            {
                "href":"http://docs.rackspacecloud.com/servers/api/v1.1/cs-devguide-20110125.pdf",
                "type":"application/pdf",
                "rel":"describedby"
            },
            {
                "href":"http://docs.rackspacecloud.com/servers/api/v1.1/application.wadl",
                "type":"application/vnd.sun.wadl+xml",
                "rel":"describedby"
            }
        ]
    }
}
```

XML

```
Not Supported.
```

Error Response

Status Code

500 - Internal Server Error

Response Data

JSON

```
{"cloudServersFault": {"message": "Server Error, please try again later.", "code": 500}}
```

XML

```
<xml data structure here>
```

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" {BaseUri}/v1.1/
```

Additional Notes

None.

4.4.1.2 List Versions
GET /

The versions schema, status field, supports an enumeration of ALPHA, BETA, CURRENT and DEPRECATED. The versions->status field should correspond to the Maturity Level for the API, i.e. ALPHA for Experimental, BETA for Exploratory, CURRENT for Public and GA, DEPRECATED for all other versions of the API that are not supported anymore.

Lists the information about all supported and deprecated versions with links to the public endpoint urls.

Request Data

URL Parameters

None.

Data Parameters

This call does not require a request body.

Success Response

Status Code

200 - OK

Response Data

JSON

```
{
    "versions":[
        {
            "status":"CURRENT",
            "updated":"2011-01-21T11:33:21Z",
            "id":"v1.1",
            "links":[
                {
                    "href":"https://az-1.region-a.geo-1.compute.hpcloudsvc.com/v1.1/",
                    "rel":"self"
                }
            ]
        },
        {
            "status":"DEPRECATED",
            "updated":"2011-01-21T11:33:21Z",
            "id":"v1.0",
            "links":[
                {
                    "href":"https://az-1.region-a.geo-1.compute.hpcloudsvc.com/v1.0/",
                    "rel":"self"
                }
            ]
        }
    ]
}
```

XML

```
Not Supported.
```

Error Response

Status Code

None.

Response Data

JSON

```
JSON format here
```

XML

```
XML format here
```

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" {BaseUri}/
```

Additional Notes

None.

4.4.2 Volumes

Describe volumes and how they relate to the overall Compute API.

Status Lifecycle

"creating" => "available" (after creation) => "in-use" (after attach) => "available" (after detach)

Rate Limits

N/A

Quota Limits

The quota as configured for the tenant governs the size of the volume that can be created. An exception will be raised if quota is exceeded. Sample exception:

```
413 Entity Too Large
{
   "overLimit":{
      "message":"Volume quota exceeded. You cannot create a volume of size 1G",
      "code":413,
      "retryAfter":0
   }
}
```

Business Rules

1. When creating a volume from an exsiting snapshot, passing in a size in the create volume call has no effect. The size of the snapshot is use to create the new volume. Currently, if you use the nova api, you have to specify a size. What will actually happen is that the size you specify will be recorded in the database and used for describe and quota operations, but the actual bock volume will be the same size as the origin volume. This is a bug.

4.4.2.1 List Volumes #### {#list_volumes}
GET /os-volumes

Lists the block storage volumes.

Request Data

URL Parameters

None.

Data Parameters

This call does not require a request body.

Success Response

Status Code

200 - OK

Response Data

JSON

```
{
    "volumes":[
      {
          "status":"available",
          "displayDescription":null,
          "availabilityZone":"nova",
          "displayName":"Test",
          "attachments":[{}],
          "volumeType":null,
          "snapshotId":"",
          "size":1,
          "id":4183,
          "createdAt":"2012-06-27 19:20:35",
          "metadata":{
          }
      },
      {
          "status":"available",
          "displayDescription":null,
          "availabilityZone":"nova",
          "displayName":null,
          "attachments":[{}],
          "volumeType":null,
          "snapshotId":"",
          "size":1,
          "id":4261,
          "createdAt":"2012-06-29 16:57:02",
          "metadata":{
          }
      }
    ]
}
```

XML

```
Not Supported.
```

Error Response

Status Code

None.

Response Data

JSON

```
JSON format here
```

XML

```
XML format here
```

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" \
         {BaseUri}/v1.1/{tenant_id}/os-volumes
```

Additional Notes

None.

4.4.2.2 Create new volume #### {#create_new_volume}
POST /os-volumes

Create a new block storage volume of a given size.

Request Data

URL Parameters

None.

Data Parameters

- *size* - integer - Size of the volume in GBs
- *display_name* (Optional) - string - Name of the volume
- *display_description* (Optional) - Description for the volume
- *snapshot_id* (Optional) - integer - Id of the volume snapshot to create the volume from
- *metadata* (Optional) - hash or dictionary - Metadata key/value pairs

JSON

```
{
    "volume":{
        "size":1,
        "display_name":"Test Volume",
        "display_description":"Test Volume desc"
    }
}
```

XML

```
Not Supported.
```

Optional:

Metadata: Metadata can be added while creating a volume by passing a key/value pair in the request body. See example below.

Success Response

Status Code

200 - OK

Response Data

JSON

```
{
    "volume":{
        "status":"creating",
        "displayDescription":"Test Volume desc",
        "availabilityZone":"nova",
        "displayName":"Test Volume",
        "attachments":[{}],
        "volumeType":null,
        "snapshotId":"",
        "size":1,
        "id":4279,
        "createdAt":"2012-06-29 19:24:03",
        "metadata":{
        }
    }
}
```

XML

```
Not Supported.
```

Error Response

Status Code

None.

Response Data

JSON

JSON format here

XML

Not Supported.

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" \
   -H "Content-Type: application/json" \
   {BaseUri}/v1.1/{tenant_id}/os-volumes \
   -X POST -d '{"volume": {"size": 1, "display_name": "Test Volume","display_description": "Te
```

Optional: Example showing creating a Volume from a Snapshot

Request Data

URL Parameters

None.

Data Parameters

JSON

```
{
   "volume":{
      "size":1,
      "snapshot_id":31,
      "display_name":"Test Volume from snapshot 31",
      "display_description":"Test Volume desc"
   }
}
```

XML

Not Supported.

Success Response

Status Code

200 - OK

Response Data

JSON

```json
{
    "volume":{
        "status":"creating",
        "displayDescription":"Test Volume desc",
        "availabilityZone":"nova",
        "displayName":"Test Volume from snapshot 31",
        "attachments":[{}],
        "volumeType":null,
        "snapshotId":31,
        "size":1,
        "id":4279,
        "createdAt":"2012-06-29 19:24:03",
        "metadata":{
        }
    }
}
```

XML

```
Not Supported.
```

Error Response

Status Code

None.

Response Data

JSON

```
JSON format here
```

XML

```
Not Supported.
```

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" \
   -H "Content-Type: application/json" \
   {BaseUri}/v1.1/{tenant_id}/os-volumes \
   -X POST -d '{"volume": {"size": 1, "snapshot_id": 31, "display_name": "Test Volume from sn
```

Optional: Example showing adding metadata

Request Data

URL Parameters

None.

Data Parameters

JSON

```
{
    "volume":{
        "size":1,
        "display_name":"Test Volume",
        "display_description":"Test Volume desc",
        "metadata":{
            "VolMeta":"VolMetaValue"
        }
    }
}
```

XML

```
Not Supported.
```

Success Response

Status Code

200 - OK

Response Data

JSON

```
{
    "volume":{
        "status":"creating",
        "displayDescription":"Test Volume desc",
        "availabilityZone":"nova",
        "displayName":"Test Volume",
        "attachments":[{}],
        "volumeType":null,
        "snapshotId":"",
        "size":1,
        "id":4279,
        "createdAt":"2012-06-29 19:24:03",
        "metadata":{
            "VolMeta":"VolMetaValue"
        }
    }
}
```

XML

```
Not Supported.
```

Error Response

Status Code

None.

Response Data

JSON

```
JSON format here
```

XML

```
Not Supported.
```

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" \
    -H "Content-Type: application/json" \
    {BaseUri}/v1.1/{tenant_id}/os-volumes \
    -X POST -d '{"volume": {"size": "1", "display_name": "Test Volume","display_description": "
```

Additional Notes

None.

4.4.2.3 Get Volume details

GET /os-volumes/{volume_id}

Gets the details of the block storage volume specified by {volume_id}.

Request Data

URL Parameters

None.

Data Parameters

This call does not require a request body.

Success Response

Status Code

200 - OK

Response Data

JSON

```
{
   "volume":{
      "status":"available",
      "displayDescription":"Test Volume desc",
      "availabilityZone":"nova",
      "displayName":"Test Volume",
      "attachments":[{}],
      "volumeType":null,
      "snapshotId":"",
      "size":1,
      "id":4279,
      "createdAt":"2012-06-29 19:24:03",
      "metadata":{
      }
   }
}
```

XML

```
Not Supported.
```

Error Response

Status Code

None.

Response Data

JSON

```
JSON format here
```

XML

```
Not Supported.
```

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" \
        {BaseUri}/v1.1/{tenant_id}/os-volumes/{volume_id}
```

Additional Notes

None.

4.4.2.4 Delete Volume
DELETE /os-volumes/{volume_id}

Delete the block storage volume specified by {volume_id}.

Request Data

URL Parameters

None.

Data Parameters

This call does not require a request body.

Success Response

Status Code

202 - Accepted

Response Data

This call does not return a response body.

Error Response

Status Code

None.

Response Data

JSON

```
JSON format here
```

XML

```
Not Supported.
```

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" \
        {BaseUri}/v1.1/{tenant_id}/os-volumes/{volume_id} -X DELETE
```

Additional Notes

None.

4.4.3 Servers

Describe the role of the server resource with respect to the Block Storage service

Status Lifecycle

"creating" => "available" (after creation) => "in-use" (after attach) => "available" (after detach)

Rate Limits

Describe Rate-limits with respect to the attach and detach calls that involve servers

None.

Quota Limits

Describe Quota-limits with respect block storage that involve servers

None.

Business Rules

1. A volume can only be attached when the status of the volume is "available".
2. A volume is attached when the status of the volume is "in-use".
3. A volume can only be detached when the status of the volume is "in-use".
4. A volume can be attached to only one server instance.
5. A server can have multiple volumes attached to it.
6. The device name should be see valid and same device name cannot be repeated.
7. The device needs to be mounted on the server, before it can be used.

4.4.3.1 List attached volumes for a server
GET /servers/{server_id}/os-volume_attachments

Gets the block storage volumes attached to the server instance specified by {server_id}.

Request Data

URL Parameters

None.

Data Parameters

This call does not require a request body.

Success Response

Status Code

200 - OK

Response Data

JSON

```
{
   "volumeAttachments":[
      {
         "device":"/dev/vdf",
         "serverId":258315,
         "id":5825,
         "volumeId":5825
      }
   ]
}
```

XML

Not Supported.

Error Response

Status Code

None.

Response Data

JSON

JSON format here

XML

Not Supported.

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" \
        {BaseUri}/v1.1/{tenant_id}/servers/{server_id}/os-volume_attachments
```

Additional Notes

None.

4.4.3.2 Attach volume to a server
POST /servers/{server_id}/os-volume_attachments

Attach the block storage volume specified in the request body to the server instance specified by {server_id}.

Request Data

URL Parameters

None.

Data Parameters

- *volumeId* - integer - Id of the volume to attach
- *device* - string - Name of the device mount point

JSON

```
{
    "volumeAttachment":{
        "volumeId":"5825",
        "device":"/dev/sdf"
    }
}
```

XML

```
Not Supported.
```

Success Response

Status Code

200 - OK

Response Data

JSON

```
{
    "volumeAttachment":{
        "id":"5825",
        "volumeId":"5825"
    }
}
```

XML

Not Supported.

Error Response

Status Code

None.

Response Data

JSON

JSON format here

XML

Not Supported.

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" \
        {BaseUri}/v1.1/{tenant_id}/servers/{server_id}/os-volume_attachments \
        -X POST -d '{"volumeAttachment": {"volumeId": "5825","device": "/dev/sdf"}}'
```

Additional Notes

None.

4.4.3.3 Detach volume from a server
DELETE /servers/{server_id}/os-volume_attachments/{volume_id}

Detach the block storage volume specified by {volume_id} from the server instance specified by {server_id}.

Request Data

URL Parameters

None.

Data Parameters

This call does not require a request body.

None.

Success Response

Status Code

202 - Accepted

Response Data

This call does not return a response body.

Error Response

Status Code

None.

Response Data

JSON

```
JSON format here
```

XML

```
Not Supported.
```

Curl Example

```
curl -i -H "X-Auth-Token: {Auth_Token}" \
      {BaseUri}/v1.1/{tenant_id}/servers/{server_id}/os-volume_attachments/{volume_id}
      -X DELETE
```

Additional Notes

None.

4.4.4 Snapshots

Enumurate the various API calls for this resource

5. Additional References

5.1 Resources

Wiki Page: Wiki page

Code Repo: Link to the internal Github repo

API Lead Contact: Tim Reddin

---

6. Glossary

Put down definitions of terms and items that need explanation.