CloudControl REST API

v2.1

Last Updated: **Dec 08 2015**
Revision: **1.02**

Contents

0. Recent Document Revisions

0.1. Changes Introduced in Revision 1.02 (this document)

Correction to the URL for the **Upgrade Server Virtual Hardware** function.

0.2. Changes Introduced in Revision 1.01

Correction to minor error in the sample response for the **Get Server** function.

0.3. Changes Introduced in Revision 1.00

This is a major documentation revision corresponding to a full CloudControl release and a versioning event for CloudControl API 2 from version 2.0 to version 2.1.

All previous API 2 documentation revisions correspond to the previous major API version 2.0, which can be downloaded here:

https://community.opsourcecloud.net/Browse.jsp?id=e5b1a66815188ad439f76183b401f026.

This documentation revision includes the following changes:

API 2.1 includes the following API functions which replace their API 2.0 namesakes:

- Infrastructure API:
    - **List Data Centers** has the following new information added:
        - CPU Speeds supported at each data center.
        - Maximum VMware Virtual Hardware version supported at each data center.
        - Visibility of VLAN Security Group enablement at each data center.
- Server API – Server Management:
    - **Deploy Server** has the following options added, which can be used to override settings on the source Server Image:
        - Memory
        - CPU count
        - Cores per Socket
        - CPU Speed (refer to **List Data Centers** for source values)
        - DNS
        - Microsoft Time Zone for Servers deployed from Microsoft Windows Server Images only.
    - **List Servers** and **Get Server** have the following information added about each Server:
        - Cores per Socket
        - CPU Speed
        - Virtual Hardware.

API 2.1 includes the following API functions which replace API 0.9 functions and had no counterpart in API 2.0:

- Server Image API; four API functions for retrieving information about Server Images at a geographic region:
    - **List OS Images.**
    - **Get OS Image**.
    - **List Customer Images**.
    - **Get Customer Image** .

API 2.1 includes the following new API functions:

- Infrastructure API:

    - **List Operating Systems** allows a user to list all of the Operating Systems supported at a given Data Center.
- Server Image API:
    - **Edit Customer Image Metadata** enables a user to update metadata about a given Customer Image; description, cpuSpeed, operatingSystemId and disk speeds.
- VLAN Security Groups are introduced:
    - **Create Security Group**
    - **List Security Groups**
    - **Add NIC to Security Group**
    - **Remove NIC from Security Group**
    - **Edit Security Group**
    - **Delete Security Group**.

This documentation revision also includes the follow corrections and minor updates:

- Element capitalization in the **List Servers** sample payload response is corrected.
- **Edit Network Domain** is updated to reflect that it can return an INCOMPATIBLE_DEPENDENCY response if it contains any VLANs with a VLAN Security Group in place.

1. Introduction

1.1. Overview

This document is intended as a high-level integration guide for developers wishing to automate the provisioning and management of Cloud resources.

It provides an overall picture of the CloudControl API 2.1 interface, with detailed API definitions required to develop application integration code against MCP 2.0 data centers, which are described in a following section.

1.2. Current API 2 Version

The current API 2 version is **API 2.1**.

This is the version should be used in all API URLs as described in the following sections and shown in each API function definition within the main content of the documentation.

1.3. MCP Data Center Identification and Support in this API Version

1.3.1. MCP 2.0

As of March 2015, new data center locations with new network architectures and functionality were introduced into some Geographic Regions and Cloud Service Providers.

These locations are defined as “MCP 2.0” architecture vs. the preexisting MCP 1.0 architecture.

MCP 2.0 locations use a completely different networking architecture and therefore require different functons to control the networking aspects of the service. For more details, see the Community article here:

Understanding MCP 1.0 vs. MCP 2.0 Data Center Locations

1.3.2. MCP 2.0 and API 2

In conjunction with the release of MCP 2.0, Cloud API 2.0 was released as the first version in the API 2 series.

Cloud API 2.0 and subsequent API 2 versions provide support for MCP 2.0 functions. Please refer to the most recent API 2 documentation for the most up to date specifics of MCP 2.0 functions.

Over time, we will update API 2 to provide ***all of the non-deprecated functionality*** provided by API 0.9, with support for both MCP 1.0 and MCP 2.0 architectures.

The initial releases of API 2 focus solely on providing functions needed to support unique MCP 2.0 functionality. Therefore, in the meantime, API integrations wanting to take advantage of MCP 2.0 locations will need to use a mix of API 0.9 and API 2 functions.

In API 2 you can identify which type of MCP architecture is associated with a data center through the **List Data Centers** function. It is important to identify the type of MCP location because some of the functions described in this API 2 document do not work with MCP 1.0 locations. This will be the case until we finish updating API 2 to support all non-deprecated API 0.9 functions.

Throughout this document and the 0.9 API document (Cloud-REST-API-v09.pdf) we have added a new “**Supported For**” row below “**Type**” on each API function **Request** definition listing one of four possible values to describe the data center location types that the function applies to:

Type	HTTP GET, HTTP POST, etc.
Supported for	*N/A* – Function does not relate to assets within data center locations *MCP 1.0* and *MCP 2.0* – Function will work with both types of data center locations *MCP 1.0* – Function will only work with MCP 1.0 data center locations. MCP 2.0 – Function will only work with MCP 2.0 data center locations.

1.4. API URL Structure

API URLs vary depending on your Cloud vendor and with which Geographic Region you are attempting to interface. A list of the available API URLs associated with each Cloud vendor is included in the **Cloud**

Vendor API URLs section further below. Note that not all Cloud implementations operate in all geographic regions.

Substitute the appropriate URL for the Geographic Region and Cloud product you wish to work with into each documented REST API URL pattern as required. For example, if you are using the Dimension Data Public CaaS implementation and want to use the Deploy Server API in the North America Geographic Region, the substitution would be as follows:

```
https://<Cloud API URL>/caas/2.1/server/deployServer
```

becomes:

```
https://api-na.dimensiondata.com/caas/2.1/server/deployServer
```

1.4.1. Cloud Vendor API URLs

Dimension Data Public CaaS (including legacy OpSource and Bluefire customers)	
North America (NA)	api-na.dimensiondata.com
Europe (EU)	api-eu.dimensiondata.com
Australia (AU)	api-au.dimensiondata.com
Africa (AF)	api-mea.dimensiondata.com
Asia Pacific (AP)	api-ap.dimensiondata.com
South America (SA)	api-latam.dimensiondata.com
Canada(CA)	api-canada.dimensiondata.com

Dimension Data's Government Community Cloud	
Canberra (CANBERRA)	api-canberra.dimensiondata.com

IS Public Cloud	
North America (NA)	usapi.cloud.is.co.za
Europe (EU)	euapi.cloud.is.co.za
Australia (AU)	auapi.cloud.is.co.za
Africa (AF)	meaapi.cloud.is.co.za
Asia Pacific (AP)	apapi.cloud.is.co.za
South America (SA)	latamapi.cloud.is.co.za
Canada (CA)	canadaapi.cloud.is.co.za

NTTA Cloud	
North America (NA)	cloudapi.nttamerica.com
Europe (EU)	eucloudapi.nttamerica.com
Australia (AU)	aucloudapi.nttamerica.com
Africa (AF)	sacloudapi.nttamerica.com
Asia Pacific (AP)	hkcloudapi.nttamerica.com

Cisco	
North America (NA)	iaas-api-na.cisco-ccs.com
Europe (EU)	iaas-api-eu.cisco-ccs.com
Australia (AU)	iaas-api-au.cisco-ccs.com
Africa (AF)	iaas-api-mea.cisco-ccs.com
Asia Pacific (AP)	iaas-api-ap.cisco-ccs.com
South America (SA)	iaas-api-sa.cisco-ccs.com
Canada (CA)	iaas-api-ca.cisco-ccs.com

Med-1	
Israel (IL)	api.cloud.med-1.com
North America (NA)	api-na.cloud.med-1.com

Europe (EU)	api-eu.cloud.med-1.com
Australia (AU)	api-au.cloud.med-1.com
Africa (AF)	api-af.cloud.med-1.com
Asia Pacific (AP)	api-ap.cloud.med-1.com
South America (SA)	api-sa.cloud.med-1.com
Canada (CA)	api-ca.cloud.med-1.com

Indosat Cloud	
Indonesia (ID)	iaas-api.indosat.com
North America (NA)	iaas-usapi.indosat.com
Europe (EU)	iaas-euapi.indosat.com
Australia (AU)	iaas-auapi.indosat.com
Africa (AF)	iaas-afapi.indosat.com

BSNL IDC	
India (IN)	api.bsnlcloud.com
North America (NA)	usapi.bsnlcloud.com
Europe (EU)	euapi.bsnlcloud.com
Australia (AU)	auapi.bsnlcloud.com
Africa (AF)	afapi.bsnlcloud.com

RootAxcess Cloud	
North America (NA)	api.rootaxesscloud.com

Tenzing Everest Cloud	
North America (NA)	api.cloud.tenzing.com

PWW Cloud Connect	
North America (NA)	api.pwwcloudconnect.net

1.5. API Groups

Similar to the previous API versions, API 2.1 is divided into top-level functional groups.

Each top-level group can be broken into smaller sets of closely related API functions. For example, those specific to managing one type of entity such as management of the Firewall Rules within a Firewall Policy.

The following groups are currently available. Each group has a corresponding URL component, which allows the user to quickly see which group a given URL belongs to.

Group	***URL Component***	***Sample URL***
Infrastructure	`/infrastructure/`	`https://<Cloud API URL>/caas/2.1/{org-id}/`**`infrastructure`**`/datacenter`
Network	`/network/`	`https://<Cloud API URL>/caas/2.1/{org-id}/`**`network`**`/natRule/{nat-rule-id}`
Server	`/server/`	`https://<Cloud API URL>/caas/2.1/{org-id}/`**`server`**`/addNic`
Server Image	`/image/`	`https://<Cloud API URL>/caas/2.1/{org-id}/`**`image`**`/osImage`

*Note: a quirk of this grouping approach is that in some cases API functions can appear to have duplication in the URL; for example the **List Servers** API function, where "/server/server" is correct and a valid part of the URL.*

- The **Infrastructure API** group gathers functions related to meta-level information about data center locations and geographic regions described further below.

- The **Network API** supports management of the components used to build and protect your networks within the Cloud. Sub-groups include Network Domains, VLANs, Firewall Policy and Network Address Translation (NAT).
- The **Server API** operations enable deployment and management of Servers and the ability to manage their network connectivity through additional Network Interface Controllers (NICs).
- The **Image API** operations enable management and querying of information regarding Server Images, which are used as the inputs for deploying new Cloud Servers.

1.6. General Approach

The general approach that the Cloud REST API adopts is as follows:

- APIs relating to the creation/deployment, deletion or update of resources (Network Domains, VLANs, Firewall Rules, Servers, etc.) are implemented as HTTP POST operations, which accept a structured payload as the request body and return a response format indicating success or failure of the request.
- Longer running API functions or those involving integration with other complex systems are asynchronous and shorter running API functions are synchronous. The API definition will indicate in each case whether a function is asynchronous or synchronous in the "Processing" section of the **Request** definition.
- APIs that retrieve details of individual resources or lists of resources are implemented as HTTP GET requests which expect the identifier of the resource to be included in the URL. These operations are read-only.
- Throughout the document, examples are provided of valid inputs for each of the APIs along with the set of error response codes specific to each function, where relevant.
- For some API fields we give specific permitted sets of values. For example *IPV4* and *IPV6* on various Firewall Rule functions. We may expand such sets of values over time as additional functionality is added within the same API version. Integrating code should be tolerant of additions.

1.6.1. Authentication

The Cloud REST API is authenticated using a standard, pre-emptive HTTP BASIC authentication request header.

All API requests are authenticated using either the Primary Administrator account credentials or those for Sub-Administrator accounts that have been created by the Primary Administrator.

API requests using Sub-Administrator account credentials are subject to the roles applied to that Sub-Administrator account when created or last updated. A request using Sub-Administrator credentials must have the requisite roles or an HTTP 403 response will be returned, as documented in the **Common API Responses** section below. The **Roles** section of the **Request** definition on every API function indicates the roles required for the API function. For example:

- *"All roles"* means that any user can execute the function user their credentials
- *"primary administrator"* means that only the Primary Administrator user can use a function
- *"network"* indicates that Sub-Administrator users with the network role can use the function.

Note: the Primary Administrator user of an organization can execute any function within this document.

1.7. Request and Response Formats (MIME types)

API 2.1 supports use of both XML and JSON as request and response types.

We recommend using either XML or JSON as intermixing the two could make your integration more difficult to maintain.

Include request headers as tabled below to ensure the system expects the MIME type of payload that you are sending and expect to receive in response:

Payload Type	HTTP Method	HTTP Header	HTTP Header Value
XML	POST	`Content-Type`	`application/xml`
XML	POST	`Accept*`	`application/xml`
XML	GET	`Accept`	`application/xml`
JSON	POST	`Content-Type`	`application/json`

JSON	POST	Accept**	application/json
JSON	GET	Accept	application/json

*An Accept header is optional in order to receive an XML response

**An Accept is required in order to receive a JSON response.

1.8. Common Response Payloads

A common response format is used to indicate ***successful responses for creation and modification* APIs** and for ***failure conditions (faults)*** across all API functions.

Each element of the response is explained in the table below followed by an XML and a JSON payload example. The **response** format includes the following information for JSON and XML payloads:

Identifier	When	Description
requestId	Required	A unique identifier, which can be used to assist diagnosis if support is required.
operation	Required	The API function, which was executed.
responseCode	Required	The status of the response. For a successful response to a **synchronous** request this will be "**OK**". For a successful response to an **asynchronous** request this will be "**IN_PROGRESS**", to indicate that the asynchronous operation has begun, as shown above.
message	Required	A human readable description of the response. The content of *message* is subject to change without notice. Do not introduce a dependency on specific *message* text into your integration.
info	Optional	Optionally an arbitrary number of info elements can be included to add further information contextually relevant to a successful response.
warning	Optional	Optionally an arbitrary number of warn elements can be included to add further information contextually relevant to a successful response. For example if further action is advised.
error	Optional	Optionally an arbitrary number of error elements can be included to add further information contextually relevant to an error response.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response requestId="na9/2015-01-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e" xmlns="urn:didata.com:api:cloud:types">
  <operation>DEPLOY_NETWORK_DOMAIN</operation>
  <responseCode>IN_PROGRESS</responseCode>
  <message>Request to Deploy Network Domain 'A Network Domain' has been
accepted and is being processed</message>
</response>
```

JSON

```
{
    "operation": "DEPLOY_NETWORK_DOMAIN",
    "responseCode": "IN_PROGRESS",
    "message": "Request to deploy Network Domain 'A Network Domain' has been
accepted and is being processed.",
    "info": [
        {
            "name": "networkDomainId",
            "value": "f14a871f-9a25-470c-aef8-51e13202e1aa"
        }
    ],
    "warning": [],
    "error": [],
    "requestId": "na9/2015-02-11T08:09:51.432-05:00/b0d7bb86-e95a-43f3-a579-
```

```
4a9259a806b8"
}
```

1.9. Considerations for POST API Functions

1.9.1. "Edit" Functions

API functions for editing an existing resource require some extra consideration when dealing with *optional* properties on the API request.

The following use cases must be satisfied:

- Does the integrator want to update an optional property?
- Does the integrator want to remove an optional property?
- Does the integrator want to leave the value of an optional property unchanged?

The Cloud Control API 2.1 API uses the XML Schema "*nillable*" approach as the means of support for each of these use cases. This is a straight forward approach which follows the rules below:

Use Case	**Approach**
Update an optional property.	Include the property in the request payload with its new value.
Delete (null out) an optional property.	Include the property in the payload, with NO value and use the "nil=true" attribute to indicate that it is to be deleted.
Leave an optional property unchanged.	Omit the property completely from the request payload.

The following examples illustrate each use case in XML and JSON:

XML Examples:

Use Case	**Example**
Update an optional property.	`<description>new text</description>`
Delete an individual optional property.	`<description xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>`
Delete a list property. Note that a single element matching the type for the list entries is submitted with nil="true".	`<iruleId xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>`
Leave an optional property unchanged.	*Submit nothing for the property.*

JSON Examples:

Use Case	**Example**
Update an optional property.	`"description": "new text"`
Delete an individual optional property.	`"description": { "nil": true }`
Delete a list property. Note that a single element matching the type for the list entries is submitted with nil="true".	`"iruleId": [ { "nil": true } ]`
Leave an optional property unchanged.	*Submit nothing for the property.*

1.9.2. Default Values in JSON

There is a limitation in the JSON conversion layer, which means that if a **required** integer or Boolean value is missing from the request payload, the system will use the default primitive value, which can result in undesirable behaviour.

An example of each case:

- In the **Edit Customer Image Metadata** function, *scsiId* is a required property for the *disk* element but if *scsiId* is **not** supplied in the JSON request payload the converter will automatically default to using “0” as this is the default for the given primitive.

  This means that the integrator updates the disk speed for scsiId “0” if they supply only one disk element and don't specify the scsiId, which might not have been their intention.

- Similarly in the **Create Firewall Rule** function, *enabled* is a required Boolean field. If omitted in a JSON payload, the system fall back on the primitive default of “false”, which could result in an unexpected enablement status for the newly created Firewall Rule.

The recommendation is to confirm and ensure that the **required** inputs specified in the XSD and API documentation are included in each JSON request payload.

1.10. API Response Codes

Many response codes are common to all API functions or to certain groups of functions.

These include the HTTP 200 response indicating a successful API request and a number of error situations as detailed in the table below.

Notes:

- Common responses are not duplicated on to the individual API function definitions within this document. They are listed below as the single point of reference.
- We may add new response codes to any API response without API versioning. Integration code should be tolerant of this through appropriate "else clause" error handling. Any new response codes added will use the HTTP 400 (Error) HTTP response code.

Response Code	HTTP Code	API Group	Message	Description
-	200 (OK)	All	n/a	Request completed successfully. Refer to the specific API function definition for details.
-	401 (Unauthorized)	All	n/a	Failed BASIC Authentication.
AUTHORIZATION_FAILURE	403 (Forbidden)	All	Your credentials do not match the organization Id in the request URL or your user does not have the right role to invoke this operation.	The supplied credentials must match secondary identification such as an Organization identifier ({org-id}) in the request URL.
n/a	404 (Not Found)	All	n/a	Standard HTTP 404 for an unknown resource URL.
n/a	500 (Server Error)	All	n/a	Standard HTTP 500 for an Internal Server Error.
INVALID_INPUT_DATA	400 (Error)	All	n/a	Indicates that a

				correction to input data such as paging or filtering options or field validation rules is required, which should be addressed by the integrator. Refer to the specific API function definition for details.
ORGANIZATION_NOT_VERIFIED	400 (Error)	Any	Organization <org-id> is unverified.	Can by thrown by an API which is unavailable to organizations that have not gone through the account verification process. **Please contact your account representative.**
SYSTEM_ERROR	400 (Error)	Any	Unexpected integration error.	Can be thrown by any **Synchronous POST** API. If you receive this response code, you will need to open a Support case.
RETRYABLE_SYSTEM_ERROR	400 (Error)	Any	Unexpected integration error. Please try again later.	Can be thrown by any **Synchronous POST** API. If you receive this response code, you can safely retry the operation.
UNEXPECTED_ERROR	400 (Error)	All	Unexpected error.	If you receive this response code, you should open a Support case to report the error.

1.11. Namespace

The CloudControl API 2.1 uses a single namespace as shown below. All entities are defined within this namespace. For example:

<datacenters xmlns="urn:didata.com:api:cloud:type" ... >.

Caution: In coding against the CloudControl API, do **NOT** hardcode references to any namespace prefixes received on responses. For example, in the XML below, the prefix for the received resource is ns2.

```
xmlns:ns2="urn:didata.com:api:cloud:type"
```

However, this **ns2** prefix is not guaranteed to be associated with any specific resource over time.

Specifically, as new releases of the Cloud platform are deployed to introduce new functionality, there is a reasonable probability that a prefix previously seen returned with one Cloud resource could be returned with *another* Cloud resource.

This is a recognized feature of SOAP / REST. As such, third party libraries like JAXB and JIBX are aware of the transient nature of namespace prefixes. Therefore, we would strongly recommend integrating using mechanisms such as these.

For example these three XML examples should all be treated identically:

- <ns2:datacenters xmlns:ns2="urn:didata.com:api:cloud:type" ... >
- <ns6:datacenters xmlns:ns6="urn:didata.com:api:cloud:type" ... >
- <datacenters xmlns="urn:didata.com:api:cloud:type" ... >

If you decide to proceed with a more fluid integration such as XPath or RegEx, then please keep in mind the need to be tolerant of these transient prefixes.

1.12. XSD

XSD can be downloaded from the Community at:

https://community.opsourcecloud.net/View.jsp?procId=8831f69f4274966cd5143b549fb31f82

Even if you intend to integrate using JSON, we strongly recommend downloading the XSD because it is a valuable source of information for understanding payload structures and can be used to generate objects in your chosen language, which can in turn be used to produce JSON payloads for API input.

1.13. Multi-Location Data Centers

Important Note: For an overview of the Multi-Location Data Centers concept, see the Community article Introduction to Multiple Data Center Locations in the same Geographic Region.

A given Geographic Region may have one or more data centers available, all of which can be managed through the API for that geographic Region. The list of available data centers can be identified using the **List Data Centers** API function in the Infrastructure group.

When a Network Domain is deployed, it is deployed into a specific MCP 2.0 data center as specified in the inputs to the **Deploy Network Domain** function. VLANs, Cloud Servers, Public IPv4 Address Blocks, Server Images, and Customer Images are examples of data center-specific objects. All VLANs deployed on a Network Domain will therefore be located in the data center in which the Network Domain was deployed and Servers linked to those VLANs must be deployed from an Server Images or Customer Images located in the same data center.

When a Customer Image is created from a Server, it will be located in the same data center and can only be used for deployments within that data center. Network Domains, VLANs and Servers cannot be copied between data centers. However, Customer Images can be copied between data centers in the same geographic region using the 0.9 API **Copy Customer Image to Data Center** function.

1.14. Multi-Geography Regions

Important Note: For an overview of the Multi-Geography Regions concept, see the Community article Introduction to Multi-Geography Regions.

Multi-Geography Regions leverage the same API structure but use a different API URL to control Cloud Networks and Cloud Servers in all data center locations in that Geographic Region. The same authentication details are used across all Geographic data centers, meaning the same username and password will function for each Geographic API URL that has been enabled on the account.

Currently, customers cannot programmatically enable Geographic Regions. Instead, the method varies depending on the Cloud vendor:

- Some Cloud providers allow the Primary Administrator to enable Geographic Regions through the Cloud UI as described in the community article How to Sign Up for an Additional Geographic Region Using the Administrative UI.
- Other Cloud providers enable Geographic Regions on the “back end” and require you to contact your sales representative for assistance.

Different Cloud providers also offer different Geographic Regions. A complete list of the available regions is available associated with a given provider can be found in the **API URL Structure** section of this

document. Again, note that the API for a given Geographic Region is only available if the relevant region has been enabled for your Cloud account.

Other than using a different URL for control of assets deployed in that region, other functionality remains the same. Like multiple data center locations, Cloud Servers, Server Images, and Customer Images are all data center-specific. However, the API for a given Geographic region will only provide information and control about assets in that region. So when calling the Europe API, the system will only respond with information about data center locations and assets available in Europe. There are no API functions that provide a "global" view across multiple geographic regions. Instead each geographic API is the "hub" that provides information and control over data center locations within that region.

Customer Images cannot be directly copied between different Geographic Regions. However, there are a series of related functions that support copying an exported Customer Image between Geographic Regions.

These API functions are part of the 0.9 API (Cloud-REST-API-v09.pdf) and include **Copy OVF Package from Remote Geo**, **List Remote OVF Package Copying in Progress**, and **List Remote OVF Package Copy History**. Please see the Community article How to Copy a Customer Image between Different Geographic Regions for an overview of the process and refer to the 0.9 API documentation for details of these functions.

Each account also has a "Home" Geographic Region. All functions related to the creation, modification, or deletion of administrators are only available within the Home Geographic Region. These functions include the **Add/Update/Delete Sub-Administrator Account** functions and the **Designate Primary Administrator** function, all of which are within the 0.9 API and beyond the scope of this document.

Lastly, please note that the 0.9 API functions related to Cloud Files - also beyond the scope of this document - are not available in all Geographic regions because Cloud Files is currently available only in the North America geographic region. Even within that region, Cloud Files is not available through all Cloud service providers.

1.15. Paging and Filtering for List API Functions

All API 2.1 functions which return a list of results based on an HTTP GET query conform to a consistent approach for **paging** behavior and provide users a means to specify **filters** which limit the results returned to the specific resources they are interested in by specifying some simple request parameters.

These features are designed to reduce the size of the XML response for quicker processing by both the Cloud software and client applications.

All such functions inherit the common behavior outlined below and the documented definition of each API details the specifics for that function.

1.15.1. Paging and Filtering Overview

Paging is the act of splitting a large list of results into consumable chunks as a means of improving performance and general manageability.

Filtering is a means of list size reduction by applying request criteria to narrow the matched set of results.

The following general rules apply across all APIs developed with the paging and filtering approach:

- Paging and filtering behavior is applied through the use of optional parameters.
- Each API has documented *default* behavior as a fallback when no parameters are supplied.
- Parameters can be supplied in *any* order.
- Inter-dependencies between parameters will be avoided and are documented on the rare occasions they arise.
- Parameters are treated as *case-insensitive*.
- Parameter names and some parameter values are based on keywords and fieldNames, which will be published in the API documentation.
    - A fieldName corresponds to a property of the type of resource being requested. For example the deployment date of a server; *createTime*.
    - A keyword can be an operator or other indication of desired behavior. Keywords are common across all API functions (within a major release) but are not necessarily supported by all functions. For example indicating that servers should be listed in descending order of deployment; *createTime.DESCENDING*

    - Keywords are also case insensitive; *createTime.DESCENDING* and *created.descending* are equally correct.
- Filters are applied through the supply of a number of parameters, with keyword comparative operators and keyword values on the API request.
- The set of fieldNames and filters for a given API function can be enhanced over time with the publication of new values. For example, an addition to the **List Servers** API to limit the results by the number of CPUs on each server returned.
- Dates are defined in XSD format but can be passed in full (including time) or short (date only) format. For example; "20120901T00:00:00Z" and "20120901" equate to the same value and time zone.

Paging

Request

Paging is supported with the introduction of three optional request parameters.

Parameter	Usage	Example
pageSize	Optional. May only be specified once. Must be a positive integer between 1 and a maximum value documented for each API function.	pageSize=50 *Means "return 50 per page"*.
pageNumber	Optional. May only be specified once. Must be a positive integer. If the value supplied exceeds the number of pages for the request, the last page will be returned.	pageNumber=3 *Means "return the 3rd page".* *If pageSize is 50, then return records 101-through-150.*
orderBy	Optional. May only be specified once. Comma separated list of *fieldNames*, each with an optional, directional *keyword*. By default each fieldName will be in ascending order unless otherwise documented.	orderBy=datacenterId,createTime.DESCENDING *Means "sort first by datacenterId ascending, then by createTime descending".*

Important note: any errors in the provision of these parameters will be returned with an HTTP 400 response and an **INVALID_INPUT_DATA** response code.

Response

The top-level element for the list of items returned in response to a paged query always includes the following common attributes:

Attribute Name	Attribute Details	Description
totalCount	Non-negative Integer	The total number of result records matching request criteria.

pageCount	Non-negative Integer	The number of result records in this page.
pageNumber	Non-negative Integer	The page number requested by the user or the default page (1) if none supplied.
pageSize	Non-negative Integer	The page size used; either specified in the request or the default for the API function being requested.

XML

In XML responses the paging information appears as attributes on the top-level element. For example, this heavily truncated **List Servers** XML response:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<servers xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="2"
totalCount="2" pageSize="250">
    <Server id="1c7762ca-f379-4eef-b08e-aa526d602589" datacenterId="NA9">
    ...
    </Server>
    <Server id="1c7762ca-f379-4eef-b08e-aa526d602589" datacenterId="NA9" >
    ...
    </Server>
    ...
</servers>
```

JSON

In JSON responses the paging information appears as properties sibling to the top-level element. For example, this heavily truncated **List Servers** JSON response:

```
{
    "server": [
        {
            "name": "Production Server 1"
             ...
        },
        {
            "name": "Production Server 2"
             ...
        }
    ],
    "pageNumber": 1,
    "pageCount": 2,
    "totalCount": 2,
    "pageSize": 250
}
```

Filtering and Operators

The documentation for each API will include the filtering criteria that it supports. The filter criteria affect only the number of rows being returned, the XML structure (or columns) returned is not affected by the filter criteria.

The usage of filters is shown below. “EQUALS” (single value) and “IN” (multiple value) queries are supported as shown below:

Usage	**Examples**
<fieldName>[.comparativeOperator]=<value>	**EQUALS:** & datacenterId=NA1

	Means return only Servers in datacenter “NA1” **IN:** &datacenterId=NA1& datacenterId=NA2 *A multivalued example including a comparator, meaning in* datacenterId *“NA1”* ***or*** *in* datacenterId *“NA2”*.
“*”	The “*” wildcard character can be used within a <*value*> to match against 0 or more characters for specific operators. Each operator supporting wildcard states that it supports wildcard within its own definition. Note that the wildcard asterisk can be escaped by using a double "**" to match a single "*".
LIKE (uses wildcard)	&name.**LIKE**=*Win*ows* An API definition can declare filters compatible with the LIKE operator. The example above uses three wildcards to indicate that the string is to be matched anywhere within the “name” filter field. Without the leading wildcard, the string would be treated as a “starts with” comparison. Similarly, without the trailing wildcard, the string would be treated as an “ends with” comparison.
MINIMUM, MIN or GE	An API definition can declare filters compatible with the GE operator. GE is synonymous with MIN and MINIMUM; meaning "greater than or equal to". &createTime.**GE**=20150901T00:00:00Z *means with date after* or identical to *2015-09-01 00:00:00*
MAXIMUM, MAX or LE	An API definition can declare filters compatible with the LE operator. LE is synonymous with MAX and MAXIMUM; meaning "less than or equal to". &createTime.**LE**=20150901T00:00:00Z *means with date before or identical to 2015-09-01 00:00:00*
GREATER_THAN or GT	An API definition can declare filters compatible with the GT operator. GT is synonymous with GREATER_THAN. &createTime.**GT**=20150901T00:00:00Z *means with date after 2015-09-01 00:00:00*
LESS_THAN or LT	An API definition can declare filters compatible with the LT operator. LT is synonymous with LESS_THAN. &createTime.**LT**=20150901T00:00:00Z *means with date before 2015-09-01 00:00:00*

1.15.2. Sample request URLs demonstrating Paging and Filtering behavior

The following selection of sample request URLs use the **List Servers** API function to demonstrate some the concepts outlined in the previous sections.

URL	**Description**

https://<Cloud API URL>/caas/2.1/{org-id}/server/server	Default API call with no parameters. Lists the first 250 servers belonging to the specified {org-id}, ordered by date of deployment (“created”) ascending.
https://<Cloud API URL>/caas/2.1/{org-id}/server/server?pageSize=100&pageNumber=2	Indicates a page size of 100 and requests the second page of results. I.e. from 101 to 200 inclusive) belonging to the specified {org-id}, ordered by date of deployment ascending (default).
https://<Cloud API URL>/caas/2.1/{org-id}/server/server?pageNumber=2&orderBy=datacenterId.DESCENDING,name	Requests the second page of results, relying on the default value of 250 for page size. I.e. from 251 to 500 inclusive) belonging to the specified {org-id}, ordered by datacenterId; descending, and then by name ascending (default).
https://<Cloud API URL>/caas/2.1/{org-id}/server/server?orderBy=name&deployed=false&datacenterId=NA1	Lists the first 250 servers that are pending deployment, belonging to the specified {org-id}, in datacenterId NA1, ordered by name ascending (default).
https://<Cloud API URL>/caas/2.1/{org-id}/server/server?deployed=true&vlanId=c01cfe97-00c7-11e2-b29c-001517c4643e	Lists the first 250 servers that have completed deployment, belonging to the specified {org-id}, resident on the VLAN identified by vlanId and ordered by date of deployment ascending (default).

2. Infrastructure API

2.1. List Data Centers

2.1.1. Description

This function identifies the list of data centers available to the organization of the authenticating user and the attributes associated each data center.

The attributes of each data center include a unique identifier “id”, the appropriate VPN URL corresponding to that location. The default data center for the organization is marked by the boolean “default” attribute.

Each data center is identified as using either MCP 1.0 or 2.0 architecture. The architecture is identified in the response within the networking properties by the “type” field as highlighted below.

```
<networking maintenanceStatus="NORMAL" type="1">
     <property value="20" name="MAX_SERVER_TO_VIP_CONNECTIONS"/>
</networking>
```

For **code integration** usage, a type=”1” value indicates a MCP 1.0 data center. A type=”2” value indicates a MCP 2.0 data center. This is of use for integrators in differentiating between the operations available at a given data center.

For **visual** reference a “type” property is given on the <datacenter> element too, e.g. “type=”MCP 2.0”. These values should not be used for your code integration.

For more details, see **MCP Location Identification and Support in this API version** in the Overview portion of this document.

Components within the infrastructure of a data center are subject to maintenance schedules, maintenance is indicated using **maintenanceStatus** elements. Where maintenanceStatus elements are present, their values can be as follows:

- NORMAL is the expected value for the majority of the time. It indicates that maintenance is not pending or in progress and that the infrastructure component is available for management.
- PENDING_MAINTENANCE indicates that maintenance is about to begin and that management functions are not available.
- IN_MAINTENANCE indicates that maintenance is in progress and that management functions are not available.

2.1.2. Request Details

URL	https://<Cloud API URL>/caas/2.1/org-id}/infrastructure/datacenter[?] Paging/Ordering Optional Parameters: [&pageSize=] [&pageNumber=] [&orderBy=] Filter Optional Parameters: [&id=]
Type	HTTP GET
Supported for	MCP 1.0 and MCP 2.0
Processing	Synchronous
Roles	All roles

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter

id	String	Identifies an individual Data Center. Human-readable, unique *datacenter id* value	Yes	Yes

2.1.3. Response Details

The following table details the information that can be returned for a given data center in the response list.

Attribute/Property	Description
id	Identifies the Data Center in the form of a human-readable, unique *datacenter id* value.
type	Indicates the infrastructure type of the data center for **visual** reference and subject to change. Should not be used for code integration but can be used for display purposes.
displayName	Informational display name of the data center. Useful for display purposes.
city	City where the data center is located.
state	State where the data center is located (where relevant).
country	Country where the data center is located.
vpnUrl	The VPN URL to use to access Cloud resources deployed at the data center.
ftpsHost	The FTPS host to use to upload OVF Packages for import as Customer Images in the data center or from which to download OVF Packages that have exported from Customer Images located at the data center.
networking	A set of attributes defining the networking infrastructure of the data center.
type	The networking infrastructure type of the data center for programmatic use. Currently the following values can be returned. This set could increase over time: • "1" indicates MCP 1.0 • "2" indicates MCP 2.0.
maintenanceStatus	Describes whether or not the networking infrastructure is in some maintenance state.
properties	The ***networking*** section for MCP 1.0 data centers includes the following properties: *MAX_SERVER_TO_VIP_CONNECTIONS* Refers to the maximum amount of Public IP addresses that can have "Server to VIP" functionality enabled at an MCP 1.0 data center: How to Manage Public IP Addresses on a Cloud Network in a MCP 1.0 location The ***networking*** section for MCP 2.0 data centers includes properties

	defining the limitations for Connection Limits and Connection Rate Limits on Nodes and Virtual Listeners: • MAX_NODE_CONNECTION_LIMIT • MAX_NODE_CONNECTION_RATE_LIMIT • MAX_VIRTUAL_LISTENER_CONNECTION_LIMIT • MAX_VIRTUAL_LISTENER_CONNECTION_RATE_LIMIT Refer to the **Network API – VIPs** sections of this document for Node and Virtual Listener creation functions and to the Community for detailed information regarding Node and Virtual Listener creation at MCP 2.0 data centers: Introduction to VIPs in MCP 2.0
hypervisor	A set of attributes defining the hypervisor infrastructure of the data center.
type	The underlying hypervisor used at the data center. Usually “VMWARE” but some private instances may be “HYPERV“.
maintenanceStatus	Describes whether or not the hypervisor infrastructure is in some maintenance state.
diskSpeed	Each data center includes a list of ***diskSpeed*** elements describing the available storage performance levels at the data center. For further information about disk speeds please review the Community article: Introduction to Cloud Server Disks and Disk Speeds (Tiered Storage).
id	A human-readable value which uniquely (to the data center) identifies the disk speed for programmatic use.
default	The default storage speed for the data center is indicated with the Boolean "default" attribute. The default storage speed is used when a new server is deployed, or additional storage is added to a server.
available	A Boolean indicating whether or not the disk speed is available for use.
displayName	Informational; a name for display purposes.
abbreviation	Informational; a short term which is useful for display purposes.
description	Informational; details the given disk speed.
cpuSpeed	Each data center includes a list of ***cpuSpeed*** elements describing the available processing levels at the data center.
id	A human-readable value which uniquely (to the data center) identifies the CPU speed for programmatic use.
default	The default CPU speed for the data center is indicated with the Boolean "default" attribute. If no CPU speed is provided on a **Deploy Server** request, the default CPU speed will be inserted.
available	A Boolean indicating whether or not the CPU speed is available for use.
displayName	Informational; a name display purposes.

description	Informational; details the given CPU speed.
properties	The ***hypervisor*** section can contain a set of optional properties. Currently the possible set is the following: *MIN_DISK_SIZE_GB* The minimum size permitted for an additional storage disk. *MAX_DISK_SIZE_GB* The maximum size permitted for an additional storage disk. *MAX_TOTAL_ADDITIONAL_STORAGE_GB* Defines the maximum amount of storage that can be added to a Server. In most cases it is ***greater*** than *MAX_TOTAL_IMAGE_STORAGE_GB*, meaning that it is possible to create Servers that cannot be cloned. This is desirable for most users where a small Customer Image is maintained as the source for deploying Servers to which storage is then added. *MAX_TOTAL_IMAGE_STORAGE_GB* Servers with **total storage greater** than the value of *MAX_TOTAL_IMAGE_STORAGE_GB* cannot be cloned (**Clone Server to Create Customer Image** function) to create new Customer Images. *MAX_CPU_COUNT* The maximum supported number of CPUs per Server. *MIN_MEMORY_GB* The minimum amount of memory permitted per Server. *MAX_MEMORY_GB* The maximum amount of memory permitted per Server. *VMWARE_HARDWARE_VERSION* The highest supported version of VMware Virtual Hardware. *VLAN_SECURITY_GROUPS_ENABLED* Presence of this property with a “true” value indicates that the VLAN Security Group feature is enabled at the given data center.
backup	Describes the status of the Cloud Backup infrastructure at the data center.
type	Indicates the backup infrastructure in use at the data center. Currently the value will always be “COMMVAULT”. Other types may be released over time.
maintenanceStatus	Describes whether or not the backup infrastructure is in some maintenance state.
properties	The ***backup*** section can contain a set of optional properties. Currently there are none published but properties could be added at any time so integrating code should support this eventuality.
consoleAccess	Describes the status of the hypervisor console infrastructure at the data center.
maintenanceStatus	Describes whether or not the console infrastructure is in some maintenance state.

properties	The ***consoleAccess*** section can contain a set of optional properties. Currently there are none published but properties could be added at any time so integrating code should support this eventuality.
monitoring	Describes the status of the Cloud Server monitoring infrastructure at the data center.
maintenanceStatus	Describes whether or not the monitoring infrastructure is in some maintenance state.
properties	The ***monitoring*** section can contain a set of optional properties. Currently there are none published but properties could be added at any time so integrating code should support this eventuality.

Below are XML and JSON sample payloads, each includes an MCP 1.0 and MCP 2.0 type data center response.

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<datacenters xmlns="urn:didata.com:api:cloud:types" pageNumber="1"
pageCount="2" totalCount="2" pageSize="250">
    <datacenter id="NA5" type="MCP 1.0">
        <displayName>North America 5</displayName>
        <city>Ashburn</city>
        <state>Virginia</state>
        <country>US</country>
        <vpnUrl>https://na.cloud-vpn.net/na5</vpnUrl>
        <ftpsHost>ftps-na5.cloud-vpn.net</ftpsHost>
        <networking type="1" maintenanceStatus="NORMAL">
            <property name="MAX_SERVER_TO_VIP_CONNECTIONS" value="10"/>
        </networking>
        <hypervisor type="VMWARE" maintenanceStatus="NORMAL">
            <diskSpeed id="STANDARD" default="true" available="true">
                <displayName>Standard Speed</displayName>
                <abbreviation>STD</abbreviation>
                <description>Standard Disk Speed</description>
            </diskSpeed>
            <diskSpeed id="ECONOMY" default="false" available="false">
                <displayName>Economy</displayName>
                <abbreviation>ECN</abbreviation>
                <description>Slower than Standard. Uses 7200 RPM disk without
Fast Cache.</description>
                <unavailableReason>Hardware change.</unavailableReason>
            </diskSpeed>
            <cpuSpeed id="HIGHPERFORMANCE" default="false" available="true">
                <displayName>High Performance</displayName>
                <description>High Performance CPU speed.</description>
            </cpuSpeed>
            <cpuSpeed id="STANDARD" default="true" available="true">
                <displayName>Standard</displayName>
                <description>Standard CPU Speed</description>
            </cpuSpeed>
            <property name="MIN_DISK_SIZE_GB" value="10"/>
            <property name="MAX_DISK_SIZE_GB" value="1000"/>
            <property name="MAX_TOTAL_ADDITIONAL_STORAGE_GB" value="750"/>
            <property name="MAX_TOTAL_IMAGE_STORAGE_GB" value="2600"/>
            <property name="MAX_CPU_COUNT" value="4"/>
            <property name="MIN_MEMORY_GB" value="4"/>
            <property name="MAX_MEMORY_GB" value="64"/>
            <property name="VMWARE_HARDWARE_VERSION" value="vmx-10"/>
        </hypervisor>
        <backup type="COMMVAULT" maintenanceStatus="NORMAL"/>
        <consoleAccess maintenanceStatus="NORMAL"/>
```

```
        <monitoring maintenanceStatus="NORMAL"/>
    </datacenter>
    <datacenter id="NA9" type="MCP 2.0">
        <displayName>North America 9</displayName>
        <city>Ashburn</city>
        <state>Virginia</state>
        <country>US</country>
        <vpnUrl>https://na.cloud-vpn.net/na9</vpnUrl>
        <ftpsHost>ftps-na9.cloud-vpn.net</ftpsHost>
        <networking type="2" maintenanceStatus="NORMAL">
            <property name="MAX_NODE_CONNECTION_LIMIT" value="100000"/>
            <property name="MAX_NODE_CONNECTION_RATE_LIMIT" value="4000"/>
            <property name="MAX_VIRTUAL_LISTENER_CONNECTION_LIMIT"
value="100000"/>
            <property name="MAX_VIRTUAL_LISTENER_CONNECTION_RATE_LIMIT"
value="4000"/>
        </networking>
        <hypervisor type="VMWARE" maintenanceStatus="NORMAL">
            <diskSpeed id="STANDARD" default="true" available="true">
                <displayName>Standard Speed</displayName>
                <abbreviation>STD</abbreviation>
                <description>Standard Disk Speed</description>
            </diskSpeed>
            <diskSpeed id="HIGHPERFORMANCE" default="false" available="true">
                <displayName>High Performance</displayName>
                <abbreviation>HPF</abbreviation>
                <description>Faster than Standard. Uses 15000 RPM disk with
Fast Cache.</description>
            </diskSpeed>
            <cpuSpeed id="STANDARD" default="true" available="true">
                <displayName>Standard</displayName>
                <description>Standard CPU Speed</description>
            </cpuSpeed>
            <cpuSpeed id="HIGHPERFORMANCE" default="false" available="true">
                <displayName>High Performance</displayName>
                <description>High Performance CPU speed.</description>
            </cpuSpeed>
            <property name="MIN_DISK_SIZE_GB" value="10"/>
            <property name="MAX_DISK_SIZE_GB" value="1000"/>
            <property name="MAX_TOTAL_ADDITIONAL_STORAGE_GB" value="2500"/>
            <property name="MAX_TOTAL_IMAGE_STORAGE_GB" value="2500"/>
            <property name="MAX_CPU_COUNT" value="8"/>
            <property name="MIN_MEMORY_GB" value="1"/>
            <property name="MAX_MEMORY_GB" value="64"/>
            <property name="VMWARE_HARDWARE_VERSION" value="vmx-10"/>
        </hypervisor>
        <backup type="COMMVAULT" maintenanceStatus="NORMAL"/>
        <consoleAccess maintenanceStatus="NORMAL"/>
        <monitoring maintenanceStatus="NORMAL"/>
    </datacenter>
</datacenters>
```

JSON

```
{
  "datacenter": [
    {
      "displayName": "North America 5",
      "city": "Ashburn",
      "state": "Virginia",
      "country": "US",
      "vpnUrl": "https://na.cloud-vpn.net/na5",
      "ftpsHost": "ftps-na5.cloud-vpn.net",
      "networking": {
        "property": [
          {
            "name": "MAX_SERVER_TO_VIP_CONNECTIONS",
            "value": "10"
```

```
          }
        ],
        "type": "1",
        "maintenanceStatus": "NORMAL"
      },
      "hypervisor": {
        "diskSpeed": [
          {
            "displayName": "Standard Speed",
            "abbreviation": "STD",
            "description": "Standard Disk Speed",
            "id": "STANDARD",
            "available": true,
            "default": true
          },
          {
            "displayName": "Economy",
            "abbreviation": "ECN",
            "description": "Slower than Standard. Uses 7200 RPM disk without
Fast Cache.",
            "unavailableReason": "Hardware capacity.",
            "id": "ECONOMY",
            "available": false,
            "default": false
          }
        ],
        "cpuSpeed": [
          {
            "displayName": "High Performance",
            "description": "High Performance CPU speed.",
            "id": "HIGHPERFORMANCE",
            "available": true,
            "default": false
          },
          {
            "displayName": "Standard",
            "description": "Standard CPU Speed",
            "id": "STANDARD",
            "available": true,
            "default": true
          }
          ],
        "property": [
          {
            "name": "MIN_DISK_SIZE_GB",
            "value": "10"
          },
          {
            "name": "MAX_DISK_SIZE_GB",
            "value": "1000"
          },
          {
            "name": "MAX_TOTAL_ADDITIONAL_STORAGE_GB",
            "value": "750"
          },
          {
            "name": "MAX_TOTAL_IMAGE_STORAGE_GB",
            "value": "2600"
          },
          {
            "name": "MAX_CPU_COUNT",
            "value": "4"
          },
          {
            "name": "MIN_MEMORY_GB",
            "value": "4"
          },
          {
            "name": "MAX_MEMORY_GB",
```

```
            "value": "64"
          },
          {
            "name": "VMWARE_HARDWARE_VERSION",
            "value": "vmx-10"
          }
        ],
        "type": "VMWARE",
        "maintenanceStatus": "NORMAL"
      },
      "backup": {
        "property": [],
        "maintenanceStatus": "NORMAL"
      },
      "consoleAccess": {
        "property": [],
        "maintenanceStatus": "NORMAL"
      },
      "monitoring": {
        "property": [],
        "maintenanceStatus": "NORMAL"
      },
      "id": "NA5",
      "type": "MCP 1.0"
    },
    {
      "displayName": "North America 9",
      "city": "Ashburn",
      "state": "Virginia",
      "country": "US",
      "vpnUrl": "https://na.cloud-vpn.net/na9",
      "ftpsHost": "ftps-na9.cloud-vpn.net",
      "networking": {
        "property": [
          {
            "name": "MAX_NODE_CONNECTION_LIMIT",
            "value": "100000"
          },
          {
            "name": "MAX_NODE_CONNECTION_RATE_LIMIT",
            "value": "4000"
          },
          {
            "name": "MAX_VIRTUAL_LISTENER_CONNECTION_LIMIT",
            "value": "100000"
          },
          {
            "name": "MAX_VIRTUAL_LISTENER_CONNECTION_RATE_LIMIT",
            "value": "4000"
          }
        ],
        "type": "2",
        "maintenanceStatus": "NORMAL"
      },
      "hypervisor": {
        "diskSpeed": [
          {
            "displayName": "Standard Speed",
            "abbreviation": "STD",
            "description": "Standard Disk Speed",
            "id": "STANDARD",
            "available": true,
            "default": true
          },
          {
            "displayName": "High Performance",
            "abbreviation": "HPF",
            "description": "Faster than Standard. Uses 15000 RPM disk with
Fast Cache.",
```

```
            "id": "HIGHPERFORMANCE",
            "available": true,
            "default": false
          }
        ],
        "cpuSpeed": [
          {
            "displayName": "Standard",
            "description": "Standard CPU Speed",
            "id": "STANDARD",
            "available": true,
            "default": true
          },
          {
            "displayName": "High Performance",
            "description": "High Performance CPU speed.",
            "id": "HIGHPERFORMANCE",
            "available": true,
            "default": false
          }
        ],
        "property": [
          {
            "name": "MIN_DISK_SIZE_GB",
            "value": "10"
          },
          {
            "name": "MAX_DISK_SIZE_GB",
            "value": "1000"
          },
          {
            "name": "MAX_TOTAL_ADDITIONAL_STORAGE_GB",
            "value": "2500"
          },
          {
            "name": "MAX_TOTAL_IMAGE_STORAGE_GB",
            "value": "2500"
          },
          {
            "name": "MAX_CPU_COUNT",
            "value": "8"
          },
          {
            "name": "MIN_MEMORY_GB",
            "value": "1"
          },
          {
            "name": "MAX_MEMORY_GB",
            "value": "64"
          },
          {
            "name": "VMWARE_HARDWARE_VERSION",
            "value": "vmx-10"
          }
        ],
        "type": "VMWARE",
        "maintenanceStatus": "NORMAL"
      },
      "backup": {
        "property": [],
        "maintenanceStatus": "NORMAL"
      },
      "consoleAccess": {
        "property": [],
        "maintenanceStatus": "NORMAL"
      },
      "monitoring": {
        "property": [],
        "maintenanceStatus": "NORMAL"
```

```
      },
      "id": "NA9",
      "type": "MCP 2.0"
    }
  ],
  "pageNumber": 1,
  "pageCount": 2,
  "totalCount": 2,
  "pageSize": 250
}
```


2.1.4. Response Codes

See ***Common Response Codes***. This function returns no additional response codes.

2.3. List Operating Systems

2.3.1. Description

Lists all of the Operating Systems available at an MCP 2.0 data center. Various filter values can be used to narrow the list returned as detailed below.

2.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/infrastructure/operatingSystem?`
	
	`Filter Required Parameters:`
	`  datacenterId=`
	
	`Filter Optional Parameters:`
	`  [&id=]`
	`  [&name=]`
	`  [&family=]`
	
	`Paging/Ordering Optional Parameters:`
	`  [&pageSize=]`
	`  [&pageNumber=]`
	`  [&orderBy=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	String	Identifies an individual Operating System. Supports the LIKE operator. *id=REDHAT664*	Yes	Yes
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes
name	String	Identifies Operating Systems by name. Supports the LIKE operator. *name.LIKE=UBUNTU**	Yes	Yes
family	String	Identifies a set of Operating Systems with a common family. Currently the set of values is limited to to “UNIX” and “WINDOWS”. This set can increase at any time, so integration should be coded defensively.	Yes	Yes

		family=UNIX		

2.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<operatingSystems xmlns="urn:didata.com:api:cloud:types" pageNumber="1"
pageCount="29" totalCount="29" pageSize="250">
  <operatingSystem id="CENTOS532" displayName="CENTOS5/32" family="UNIX"/>
  <operatingSystem id="CENTOS564" displayName="CENTOS5/64" family="UNIX"/>
  <operatingSystem id="CENTOS632" displayName="CENTOS6/32" family="UNIX"/>
  <operatingSystem id="CENTOS664" displayName="CENTOS6/64" family="UNIX"/>
  <operatingSystem id="REDHAT432" displayName="REDHAT4/32" family="UNIX"/>
  <operatingSystem id="REDHAT464" displayName="REDHAT4/64" family="UNIX"/>
  <operatingSystem id="REDHAT532" displayName="REDHAT5/32" family="UNIX"/>
  <operatingSystem id="REDHAT564" displayName="REDHAT5/64" family="UNIX"/>
  <operatingSystem id="REDHAT632" displayName="REDHAT6/32" family="UNIX"/>
  <operatingSystem id="REDHAT664" displayName="REDHAT6/64" family="UNIX"/>
  <operatingSystem id="SLES1164" displayName="SUSE11/64" family="UNIX"/>
  <operatingSystem id="SUSE1164" displayName="SUSE11/64" family="UNIX"/>
  <operatingSystem id="UBUNTU1064" displayName="UBUNTU10/64" family="UNIX"/>
  <operatingSystem id="UBUNTU1264" displayName="UBUNTU12/64" family="UNIX"/>
  <operatingSystem id="UBUNTU864" displayName="UBUNTU8/64" family="UNIX"/>
  <operatingSystem id="WIN2003E32" displayName="WIN2003E/32"
family="WINDOWS"/>
  <operatingSystem id="WIN2003E64" displayName="WIN2003E/64"
family="WINDOWS"/>
  <operatingSystem id="WIN2003S32" displayName="WIN2003S/32"
family="WINDOWS"/>
  <operatingSystem id="WIN2003S64" displayName="WIN2003S/64"
family="WINDOWS"/>
  <operatingSystem id="WIN2008E32" displayName="WIN2008E/32"
family="WINDOWS"/>
  <operatingSystem id="WIN2008E64" displayName="WIN2008E/64"
family="WINDOWS"/>
  <operatingSystem id="WIN2008R2DC64" displayName="WIN2008R2DC/64"
family="WINDOWS"/>
  <operatingSystem id="WIN2008R2E64" displayName="WIN2008R2E/64"
family="WINDOWS"/>
  <operatingSystem id="WIN2008R2S64" displayName="WIN2008R2S/64"
family="WINDOWS"/>
  <operatingSystem id="WIN2008S32" displayName="WIN2008S/32"
family="WINDOWS"/>
  <operatingSystem id="WIN2008S64" displayName="WIN2008S/64"
family="WINDOWS"/>
  <operatingSystem id="WIN2012DC64" displayName="WIN2012DC/64"
family="WINDOWS"/>
  <operatingSystem id="WIN2012R2DC64" displayName="WIN2012R2DC/64"
family="WINDOWS"/>
  <operatingSystem id="WIN2012S64" displayName="WIN2012S/64"
family="WINDOWS"/>
</operatingSystems>
```

JSON

```
{
  "operatingSystem": [
    {
      "id": "CENTOS532",
      "displayName": "CENTOS5/32",
      "family": "UNIX"
    },
    {
      "id": "CENTOS564",
      "displayName": "CENTOS5/64",
      "family": "UNIX"
```

```
    },
    {
      "id": "CENTOS632",
      "displayName": "CENTOS6/32",
      "family": "UNIX"
    },
    {
      "id": "CENTOS664",
      "displayName": "CENTOS6/64",
      "family": "UNIX"
    },
    {
      "id": "REDHAT432",
      "displayName": "REDHAT4/32",
      "family": "UNIX"
    },
    {
      "id": "REDHAT464",
      "displayName": "REDHAT4/64",
      "family": "UNIX"
    },
    {
      "id": "REDHAT532",
      "displayName": "REDHAT5/32",
      "family": "UNIX"
    },
    {
      "id": "REDHAT564",
      "displayName": "REDHAT5/64",
      "family": "UNIX"
    },
    {
      "id": "REDHAT632",
      "displayName": "REDHAT6/32",
      "family": "UNIX"
    },
    {
      "id": "REDHAT664",
      "displayName": "REDHAT6/64",
      "family": "UNIX"
    },
    {
      "id": "SLES1164",
      "displayName": "SUSE11/64",
      "family": "UNIX"
    },
    {
      "id": "SUSE1164",
      "displayName": "SUSE11/64",
      "family": "UNIX"
    },
    {
      "id": "UBUNTU1064",
      "displayName": "UBUNTU10/64",
      "family": "UNIX"
    },
    {
      "id": "UBUNTU1264",
      "displayName": "UBUNTU12/64",
      "family": "UNIX"
    },
    {
      "id": "UBUNTU864",
      "displayName": "UBUNTU8/64",
      "family": "UNIX"
    },
    {
      "id": "WIN2003E32",
      "displayName": "WIN2003E/32",
```

```
      "family": "WINDOWS"
    },
    {
      "id": "WIN2003E64",
      "displayName": "WIN2003E/64",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2003S32",
      "displayName": "WIN2003S/32",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2003S64",
      "displayName": "WIN2003S/64",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2008E32",
      "displayName": "WIN2008E/32",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2008E64",
      "displayName": "WIN2008E/64",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2008R2DC64",
      "displayName": "WIN2008R2DC/64",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2008R2E64",
      "displayName": "WIN2008R2E/64",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2008R2S64",
      "displayName": "WIN2008R2S/64",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2008S32",
      "displayName": "WIN2008S/32",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2008S64",
      "displayName": "WIN2008S/64",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2012DC64",
      "displayName": "WIN2012DC/64",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2012R2DC64",
      "displayName": "WIN2012R2DC/64",
      "family": "WINDOWS"
    },
    {
      "id": "WIN2012S64",
      "displayName": "WIN2012S/64",
      "family": "WINDOWS"
    }
  ],
  "pageNumber": 1,
```

```
  "pageCount": 29,
  "totalCount": 29,
  "pageSize": 250
}
```

2.3.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

Response Code	Message
INVALID_INPUT_DATA	datacenterId is required. *See **List Data Centers**.*

3. Network API – Network Domain Management

3.1. Deploy Network Domain

3.1.1. Description

A Network Domain is the fundamental building block for your MCP 2.0 Cloud deployment. At least one Network Domain and one VLAN must be deployed before you can deploy your first Server in an MCP 2.0 data center.

Refer to the Community for further details:

https://community.opsourcecloud.net/View.jsp?procId=994fa801956149b3861e428801f9888f

3.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/deployNetworkDomain`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Asynchronous. For status refer to **Get Network Domain** and **List Network Domains**.
Roles	"primary administrator" and "network"

XML Request Sample

```
<deployNetworkDomain xmlns="urn:didata.com:api:cloud:types">
  <datacenterId>NA9</datacenter>
  <name>A Network Domain</name>
  <description>Network Domain for production VLANs and Servers</description>
  <type>ESSENTIALS</type>
</deployNetworkDomain>
```

JSON Request Sample

```
{
    "datacenterId": "NA9",
    "name": "A Network Domain",
    "description": "Network Domain for production VLANs and Servers",
    "type": "ESSENTIALS"
}
```

Request Properties

Field	Required	Type and Constraints
datacenterId	Yes	The datacenter.id of the Data Center where the Network Domain should be deployed. See **List Data Centers**.
name	Yes	Minimum length: 1 character. Maximum length: 75 characters.
description	No	Maximum length: 255 characters.
type	Yes	Must be one of: • ESSENTIALS • ADVANCED.

3.1.3. Response Details

The response includes an "*info*" property named "*networkDomainId*". The value of this property contains the unique identifier of the Network Domain being deployed and can be used in conjunction with **Get Network Domain** and or **List Network Domains** to retrieve the status of this asynchronous operation.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response requestId="na/2015-01-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e" xmlns="urn:didata.com:api:cloud:types">
  <operation>DEPLOY_NETWORK_DOMAIN</operation>
  <responseCode>IN_PROGRESS</responseCode>
  <message>Request to Deploy Network Domain 'A Network Domain' has been
accepted and is being processed</message>
  <info name="networkDomainId" value="f14a871f-9a25-470c-aef8-51e13202e1aa"/>
</response>
```

JSON

```
{
    "operation": "DEPLOY_NETWORK_DOMAIN",
    "responseCode": "IN_PROGRESS",
    "message": "Request to deploy Network Domain 'A Network Domain' has been
accepted and is being processed.",
    "info": [
        {
            "name": "networkDomainId",
            "value": "f14a871f-9a25-470c-aef8-51e13202e1aa"
        }
    ],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-02-11T08:09:51.432-05:00/b0d7bb86-e95a-43f3-a579-
4a9259a806b8"
}
```

3.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
EXCEEDS_LIMIT	An unverified Organization may deploy a maximum of <maxValue> Network Domains in a data center. *The client organization identified by {org-id} has not been verified. Please contact your account representative for verification.*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	<datacenterId> must be provided. *See **List Data Centers**.* Invalid <name>: exceeds maximum length of 75 characters. Invalid <name>: name must be provided and must be at least 1 character in length. Invalid <description>: exceeds maximum length of 255 characters. <type> must be provided and must be one of ESSENTIALS or ADVANCED.
NAME_NOT_UNIQUE	Another Network Domain named '<name>' already exists in data center <datacenterId>.

	A Network Domain already exists with the same name.
OPERATION_NOT_SUPPORTED	Data center <datacenterId> is not a MCP 2.0 data center. *This operation can only be executed at an MCP 2.0 data center. Refer to **MCP Location Identification and Support in this API Version** for background information. Refer to **List Data Centers** to see the full list of available data centers and their types.*
OUT_OF_RESOURCES	No unassigned Network Domains are currently available in data center <datacenterId>.
RESOURCE_NOT_FOUND	Data center <datacenterId> not found.

3.2. List Network Domains

3.2.1. Description

Lists all of the Network Domains belonging to the organization identified by the {org-id} parameter. Various filter values can be used to narrow the list returned as detailed below.

Note that it is more efficient to use **Get Network Domain** than **List Network Domains** (filtered by *id*) if you wish to retrieve *state* status or details of a single Network Domain.

3.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/networkDomain[?]`
	
	`Paging/Ordering Optional Parameters:`
	`[&pageSize=]`
	`[&pageNumber=]`
	`[&orderBy=]`
	
	`Filter Optional Parameters:`
	`[&id=]`
	`[&datacenterId=]`
	`[&name=]`
	`[&type=]`
	`[&state=]`
	`[&createTime=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	All roles

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual Network Domain. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes
name	String	Identifies Network Domains by their *name*. *name=ANetworkDomain* Supports the use of the LIKE comparator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	Yes
type	String	Identifies Network Domains by their *type*; “ESSENTIALS” or “ADVANCED”.	Yes	Yes
state	String	Identifies Network Domains by their *state*.	Yes	Yes

		Case insensitive. The initial possible set of values for *networkDomain.state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.		
createTime	Date	Identifies the date of creation of Network Domains. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes

3.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<networkDomains
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="2"
totalCount="2" pageSize="250">
    <networkDomain id="484174a2-ae74-4658-9e56-50fc90e086cf" datacenter="NA9">
        <name>Production Network Domain</name>
        <description></description>
        <type>ESSENTIALS</type>
        <snatIpv4Address>165.180.8.8</snatIpv4Address>
        <createTime>2015-02-12T18:18:14.000Z</createTime>
        <state>NORMAL</state>
    </networkDomain>
    <networkDomain id="8cdfd607-f429-4df6-9352-162cfc0891be" datacenter="NA9">
        <name>Development Network Domain</name>
        <description>This is a new Network Domain</description>
        <type>ESSENTIALS</type>
        <snatIpv4Address>165.180.9.252</snatIpv4Address>
        <createTime>2015-02-24T10:47:21.000Z</createTime>
        <state>NORMAL</state>
    </networkDomain>
</networkDomains>
```

JSON

```
{
    "networkDomain": [
        {
            "name": "Production Network Domain",
            "description": "",
            "type": "ESSENTIALS",
            "snatIpv4Address": "165.180.8.8",
            "createTime": "2015-02-24T10:47:21.000Z",
            "state": "NORMAL",
            "id": "484174a2-ae74-4658-9e56-50fc90e086cf",
            "datacenter": "NA9"
        },
        {
```

```
            "name": "Development Network Domain",
            "description": "This is a new Network Domain",
            "type": "ESSENTIALS",
            "snatIpv4Address": "165.180.9.252",
            "createTime": "2015-02-24T10:47:21.000Z",
            "state": "NORMAL",
            "id": "8cdfd607-f429-4df6-9352-162cfc0891be",
            "datacenter": "NA9"
        }
    ],
    "pageNumber": 1,
    "pageCount": 2,
    "totalCount": 2,
    "pageSize": 250
}
```


3.2.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

3.3. Get Network Domain

3.3.1. Description

Returns details of a single Network Domain belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get Network Domain** than **List Network Domains** (filtered by *id*) if you wish to retrieve *state* status or details of a single Network Domain.

3.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/networkDomain/{networkdomain-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	All roles

3.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<networkDomain
    xmlns="urn:didata.com:api:cloud:types" id="8cdfd607-f429-4df6-9352-
162cfc0891be" location="NA9">
    <name>Development Network Domain</name>
    <description>This is a new Network Domain</description>
    <type>ESSENTIALS</type>
    <snatIpv4Address>165.180.9.252</snatIpv4Address>
    <createTime>2015-02-24T10:47:21.000Z</createTime>
    <state>NORMAL</state>
</networkDomain>
```

JSON

```
{
    "name": "Development Network Domain",
    "description": "This is a new Network Domain",
    "type": "ESSENTIALS",
    "snatIpv4Address": "165.180.9.252",
    "createTime": "2015-02-24T10:47:21.000Z",
    "state": "NORMAL",
    "id": "8cdfd607-f429-4df6-9352-162cfc0891be",
    "datacenter": "NA9"
}
```

3.3.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Network Domain {networkDomain-id} not found.

3.4. Edit Network Domain

3.4.1. Description

Updates the meta data associated with a Network Domain belonging to the organization identified by {org-id}.

3.4.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/editNetworkDomain`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<editNetworkDomain xmlns="urn:didata.com:api:cloud:types" id="8cdfd607-f429-4df6-9352-162cfc0891be">
   <name>Development Network Domain</name>
   <description>Development Network Domain</description>
   <type>ESSENTIALS</type>
</editNetworkDomain>
```

JSON Request Sample

```
{
  "id": "8cdfd607-f429-4df6-9352-162cfc0891be",
  "name": "Development Network Domain",
  "description": "Encapsulates all our development Cloud resources.",
  "type": "ESSENTIALS"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Network Domain to be edited.
name	No	Minimum length: 1 characters. Maximum length: 75 characters.
description	No	Maximum length: 255 characters.
type	No	Must be one of: • ESSENTIALS • ADVANCED.

3.4.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-02-24T09:05:41.790-05:00/b24cbc7d-0648-4070-889f-7ab97bf21941">
    <operation>EDIT_NETWORK_DOMAIN</operation>
    <responseCode>OK</responseCode>
    <message>Network Domain 'Development Network Domain' was edited successfully.</message>
</response>
```

JSON

```
{
    "operation": "EDIT_NETWORK_DOMAIN",
    "responseCode": "OK",
    "message": " Network Domain 'Development Network Domain' was edited
successfully.",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-02-24T09:12:09.204-05:00/7b2c505c-4874-4b5c-8dff-
15be0c169a7d"
}
```

3.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	Network Domain '<name>' cannot be changed to ESSENTIALS because it contains at least one Node that has non-default settings. *The Node underlying a NAT Rule can provoke this response if it has been altered from default settings.*
INCOMPATIBLE_DEPENDENCY	'<name>' has Server Anti-Affinity Rules. All Server Anti-Affinity Rules must be deleted before the Network Domain type can be downgraded from ADVANCED to ESSENTIALS. Attempt to downgrade Network Domain <name> from ADVANCED to ESSENTIALS is invalid because it at least one Node that is not in use by a NAT. Attempt to downgrade Network Domain <name> from ADVANCED to ESSENTIALS is invalid because it has at least one Pool. Attempt to downgrade Network Domain <name> from ADVANCED to ESSENTIALS is invalid because it has at least one Virtual Listener. Attempt to downgrade Network Domain <name> from ADVANCED to ESSENTIALS is invalid because it has at least one VLAN with a VLAN Security Group.
INVALID_INPUT_DATA	id must be provided. Network Domain name must be between 1 and 75 characters in length. Invalid description: exceeds maximum length of 255 characters. type must be one of ESSENTIALS or ADVANCED.
NAME_NOT_UNIQUE	Another Network Domain named '<name>' already exists in data center <datacenterId>.
RESOURCE_BUSY	Another operation is in progress on Network Domain with id <id>. Please try again later.
RESOURCE_LOCKED	Network Domain with id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Network Domain <id> not found.

3.5. Delete Network Domain

3.5.1. Description

Deletes a Network Domain belonging to the organization identified by {org-id}.

3.5.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/deleteNetworkDomain`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Asynchronous. For status refer to **Get Network Domain** and **List Network Domain**.
Roles	"primary administrator" and "network"

XML Request Sample

```
<deleteNetworkDomain xmlns="urn:didata.com:api:cloud:types" id="8cdfd607-f429-4df6-9352-162cfc0891be"/>
```

JSON Request Sample

```
{
  "id": "8cdfd607-f429-4df6-9352-162cfc0891be"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Network Domain to be deleted.

3.5.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-02-24T10:19:41.471-05:00/27083502-a2ea-4331-b0cb-c2a11bb044b0">
    <operation>DELETE_NETWORK_DOMAIN</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to Delete Network Domain (Id: 8cdfd607-f429-4df6-9352-162cfc0891be) has been accepted and is being processed</message>
</response>
```

JSON

```
{
    "operation": "DELETE_NETWORK_DOMAIN",
    "responseCode": "IN_PROGRESS",
    "message": "Request to Delete Network Domain (Id: 8cdfd607-f429-4df6-9352-162cfc0891be) has been accepted and is being processed",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-02-24T10:17:57.718-05:00/ea47512c-c656-49cc-9878-15ee0305081e"
}
```

3.5.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
HAS_DEPENDENCY	Network Domain '<name>' cannot be deleted because it has at least one IP Block. Network Domain '<name>' cannot be deleted because it has at least one VLAN. Network Domain '<name>' cannot be deleted because it has at least one NAT Rule. Network Domain '<networkDomainName>' cannot be deleted because it has at least one Pool. Network Domain '<networkDomainName>' cannot be deleted because it has at least one Node. Network Domain '<networkDomainName>' cannot be deleted because it has at least one Virtual Listener.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	id must be provided.
RESOURCE_BUSY	Another operation is in progress on Network Domain with id <id>. Please try again later.
RESOURCE_LOCKED	Network Domain with id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Network Domain <id> not found.

4. Network API – VLAN Management

4.1. Deploy VLAN

4.1.1. Description

This function deploys a VLAN on a Network Domain in an MCP 2.0 data center location.

4.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/deployVlan`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Asynchronous. For status refer to **Get VLAN** and **List VLANs**.
Roles	"primary administrator" and "network"

XML Request Sample

```
<deployVlan xmlns="urn:didata.com:api:cloud:types">
  <networkDomainId>553f26b6-2a73-42c3-a78b-6116f11291d0</networkDomainId>
  <name>Production VLAN</name>
  <description>For hosting our Production Cloud Servers</description>
  <privateIpv4BaseAddress>10.3.4.0</privateIpv4BaseAddress>
  <!-- Optional. Defaults to /24 if not included -->
  <privateIpv4PrefixSize>23</privateIpv4PrefixSize>
</deployVlan>
```

JSON Request Sample

```
{
    "networkDomainId": "484174a2-ae74-4658-9e56-50fc90e086cf",
    "name": "Production VLAN",
    "description": "For hosting our Production Cloud Servers",
    "privateIpv4BaseAddress": "10.0.3.0",
    "privateIpv4PrefixSize": 23
}
```

Request Properties

Field	Required	Type and Constraints
networkDomainId	Yes	The “id” of a Network Domain belonging to {org-id} within the same MCP 2.0 data center.
name	Yes	Minimum length: 1 character. Maximum length: 75 characters.
description	No	Maximum length: 255 characters.
privateIpv4BaseAddress	Yes	RFC1918 Dot-decimal representation of an IPv4 address. For example: “10.0.4.0”. Must be unique within the Network Domain.
privateIpv4PrefixSize	No	An Integer between 16 and 24, which represents the size of the VLAN to be deployed and must be consistent with the privateIpv4BaseAddress provided. If this property is not provided, the VLAN will default to being “/24”.

4.1.3. Response Details

The response includes an "*info*" property named "*vlanId*". The value of this property contains the unique identifier of the VLAN being deployed and can be used in conjunction with **Get VLAN** and or **List VLANs** to retrieve the status of this asynchronous operation.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-02-
13T05:58:00.905-05:00/04a82e12-2d51-420e-84f5-bf2630c5a5e5">
    <operation>DEPLOY_VLAN</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to deploy VLAN 'Production VLAN' has been accepted and is
being processed.</message>
    <info name="vlanId" value="cee8df03-9117-44cc-baaa-631ffa099683"/>
</response>
```

JSON

```
{
    "operation": "DEPLOY_VLAN",
    "responseCode": "IN_PROGRESS",
    "message": "Request to deploy VLAN 'Production VLAN' has been accepted and
is being processed.",
    "info": [
        {
            "name": "vlanId",
            "value": "0e56433f-d808-4669-821d-812769517ff8"
        }
    ],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-02-13T05:56:44.573-05:00/c6888cbc-e253-4807-96c4-
de84ca3e9109"
}
```

4.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
EXCEEDS_LIMIT	The maximum permitted number (<maxValue>) of VLANs that can be deployed on a Network Domain has already been reached. An unverified Organization may deploy a maximum of <maxValue> VLANs in a data center. Please contact your account representative for verification.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	<field name> must be provided. Invalid name: exceeds maximum length of 75 characters. Invalid name: name must be provided and must be at least 1 character in length. Invalid description: exceeds maximum length of 255 characters. privateIpv4BaseAddress <address> is invalid. It must be an IPv4 address on a /24 CIDR boundary (i.e. it must end with

	.0). *Only if privateIpv4PrefixSize is **not** provided.* privateIpv4PrefixSize must be an integer between 16 and 24. The privateIpv4BaseAddress/privateIpv4PrefixSize combination is invalid. The privateIpv4BaseAddress must be on a valid CIDR boundary for the specified privateIpv4PrefixSize.
IP_ADDRESS_NOT_UNIQUE	Another VLAN using privateIpv4BaseAddress <privateIpv4BaseAddress> already exists in Network Domain '<network domain name>'. The requested VLAN space <privateIpv4BaseAddress >/<privateIpv4PrefixSize > overlaps with space used by an existing VLAN <vlan name> (<existing privateIpv4BaseAddress >/<existing privateIpv4PrefixSize >) in the same Network Domain Virtual Listener(s) ({virtual-listener-name(s)}) is listening on an address/addresses in the requested VLAN subnet. You must remove the Virtual Listener(s) before creating the VLAN of this size. *Note that the response will include error elements describing up to 10 such dependencies.* There is at least one NAT Rule or Node defined that falls into reserved IP addresses x.x.x.0 - x.x.x.5, or x.x.x.7 or last IP (broadcast) address of the requested subnet. It must be removed before deploying a VLAN of the requested size. *Note that the response will include error elements describing up to 10 such dependencies.*
NAME_NOT_UNIQUE	Another VLAN named '<name>' already exists in data center <datacenterId>.
OUT_OF_RESOURCES	No unassigned VLANs currently available in data center <datacenterId>.
RESOURCE_BUSY	Another operation is in progress on Network Domain with id <networkDomainId>. Please try again later. Another Deploy or Delete Network (or VLAN) operation is in progress. Please try again later.
RESOURCE_LOCKED	Network Domain with id <networkDomainId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Network Domain <networkDomainId> not found. *The value passed for networkDomainId does not correspond to a valid Network Domain belonging to {org-id}. See **List Network Domains**.*

4.2. List VLANs

4.2.1. Description

Lists all of the VLANs on a particular Network Domain at an MCP 2.0 data center, belonging to the organization identified by the {org-id} parameter. Various filter values can be used to narrow the list returned as detailed below.

Note that it is more efficient to use **Get VLAN** than **List VLANs** (filtered by *id*) if you wish to retrieve *state* status or details of a single VLAN.

4.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/vlan[?]`
	
	`Paging/Ordering Optional Parameters:`
	`[&pageSize=]`
	`[&pageNumber=]`
	`[&orderBy=]`
	
	`Filter Optional Parameters:`
	`[&id=]`
	`[&networkDomainId=]`
	`[&datacenterId=]`
	`[&name=]`
	`[&privateIpv4Address=]`
	`[&ipv6Address=]`
	`[&state=]`
	`[&createTime=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual VLAN. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
networkDomainId	uuid (String)	Identifies an individual Network Domain. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes
name	String	Identifies VLANs by their *name*. *name=ProductionVLAN* Supports the use of the LIKE comparator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	Yes

privateIpv4Address	String	Identifies VLANs by their base network IPv4 address. *privateIpv4Address=10.1.1.0*	Yes	Yes
ipv6Address	String	Identifies VLANs by their base network IPv6 address. *ipv6Address= 2607:f480:1111:1102:0:0:0:0*	Yes	Yes
state	String	Identifies VLANs by their *state*. Case insensitive. The initial possible set of values for *vlan.state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.	Yes	Yes
createTime	Date	Identifies the date of creation of VLANs. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes

4.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<vlans
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="1"
totalCount="1" pageSize="250">
    <vlan id="0e56433f-d808-4669-821d-812769517ff8" location="NA9">
        <networkDomain id="484174a2-ae74-4658-9e56-50fc90e086cf"
name="Production Network Domain"/>
        <name>Production VLAN</name>
        <description>For hosting our Production Cloud Servers</description>
        <privateIpv4Range address="10.0.3.0" prefixSize="24"/>
        <ipv4GatewayAddress>10.0.3.1</ipv4GatewayAddress>
        <ipv6Range address="2607:f480:1111:1153:0:0:0:0" prefixSize="64"/>
        <ipv6GatewayAddress>2607:f480:1111:1153:0:0:0:1</ipv6GatewayAddress>
        <createTime>2015-02-13T10:56:44.000Z</createTime>
        <state>NORMAL</state>
    </vlan>
</vlans>
```

JSON

```
{
    "vlan": [
        {
            "networkDomain": {
                "id": "484174a2-ae74-4658-9e56-50fc90e086cf",
```

```
                "name": "Production Network Domain"
            },
            "name": "Production VLAN",
            "description": "For hosting our Production Cloud Servers",
            "privateIpv4Range": {
                "address": "10.0.3.0",
                "prefixSize": 24
            },
            "ipv4GatewayAddress": "10.0.3.1",
            "ipv6Range": {
                "address": "2607:f480:1111:1153:0:0:0:0",
                "prefixSize": 64
            },
            "ipv6GatewayAddress": "2607:f480:1111:1153:0:0:0:1",
            "createTime": 1423825004000,
            "state": "NORMAL",
            "id": "0e56433f-d808-4669-821d-812769517ff8",
            "datacenterId": "NA9"
        }
    ],
    "pageNumber": 1,
    "pageCount": 1,
    "totalCount": 1,
    "pageSize": 250
}
```


4.2.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

4.3. Get VLAN

4.3.1. Description

Returns details of a single VLAN belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get VLAN** than **List VLANs** (filtered by *id*) if you wish to retrieve *state* status or details of a single VLAN.

4.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/vlan/{vlan-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

4.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<vlan
    xmlns="urn:didata.com:api:cloud:types" id="0e56433f-d808-4669-821d-
812769517ff8" location="NA9">
    <networkDomain id="484174a2-ae74-4658-9e56-50fc90e086cf" name="Production
Network Domain"/>
    <name>Production VLAN</name>
    <description>For hosting our Production Cloud Servers</description>
    <privateIpv4Range address="10.0.3.0" prefixSize="24"/>
    <ipv4GatewayAddress>10.0.3.1</ipv4GatewayAddress>
    <ipv6Range address="2607:f480:1111:1153:0:0:0:0" prefixSize="64"/>
    <ipv6GatewayAddress>2607:f480:1111:1153:0:0:0:1</ipv6GatewayAddress>
    <createTime>2015-02-13T10:56:44.000Z</createTime>
    <state>NORMAL</state>
</vlan>
```

JSON

```
{
    "networkDomain": {
        "id": "484174a2-ae74-4658-9e56-50fc90e086cf",
        "name": "Production Network Domain"
    },
    "name": "Production VLAN",
    "description": "For hosting our Production Cloud Servers",
    "privateIpv4Range": {
        "address": "10.0.3.0",
        "prefixSize": 24
    },
    "ipv4GatewayAddress": "10.0.3.1",
    "ipv6Range": {
        "address": "2607:f480:1111:1153:0:0:0:0",
        "prefixSize": 64
    },
    "ipv6GatewayAddress": "2607:f480:1111:1153:0:0:0:1",
    "createTime": 1423825004000,
    "state": "NORMAL",
    "id": "0e56433f-d808-4669-821d-812769517ff8",
    "datacenterId": "NA9"
}
```

4.3.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	VLAN {vlan-id} not found. *The value passed for {vlan-id} does not correspond to a valid VLAN belonging to {org-id}. See **List VLANs**.*

4.4. Edit VLAN

4.4.1. Description

Updates the meta data associated with a VLAN belonging to the organization identified by {org-id}.

4.4.2. Request Details

URL	https://<Cloud API URL>/caas/2.1/{org-id}/network/editVlan
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<editVlan xmlns="urn:didata.com:api:cloud:types" id="0e56433f-d808-4669-821d-
812769517ff8">
   <name>Production VLAN</name>
   <description>For hosting our Production Cloud Servers</description>
</editVlan>
```

JSON Request Sample

```
{
  "id": "0e56433f-d808-4669-821d-812769517ff8",
  "name": "Production VLAN",
  "description": "For hosting our Production Cloud Servers"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the VLAN to be edited.
name	No	Maximum length: 75 characters.
description	No	Maximum length: 255 characters.

4.4.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-02-
25T06:55:10.939-05:00/0593b9af-6b0c-4e5b-8e74-57b0e752f2bf">
    <operation>EDIT_VLAN</operation>
    <responseCode>OK</responseCode>
    <message>VLAN '<name>' was edited successfully</message>
</response>
```

JSON

```
{
    "operation": "EDIT_VLAN",
    "responseCode": "OK",
    "message": "VLAN '<name>' was edited successfully",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-02-25T08:40:03.842-05:00/86aff2a2-0abf-49b2-89ab-
```

```
57db6af9b463"
}
```

4.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	Invalid name: exceeds maximum length of 75 characters. Invalid description: exceeds maximum length of 255 characters. At least one of "name" or "description" must be provided.
NAME_NOT_UNIQUE	Another VLAN named '<name>' already exists in data center <datacenterId>.
RESOURCE_BUSY	Another operation is in progress on VLAN with Id <id>. Please try again later.
RESOURCE_LOCKED	VLAN with id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	VLAN <id> not found. *The value passed for <id> does not correspond to a valid VLAN belonging to {org-id}. See **List VLANs**.*

4.5. Delete VLAN

4.5.1. Description

Deletes a VLAN belonging to the organization identified by {org-id}.

4.5.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/deleteVlan`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Asynchronous. For status refer to **Get VLAN** and **List VLANs**.
Roles	"primary administrator" and "network"

XML Request Sample

```
<deleteVlan id="0e56433f-d808-4669-821d-812769517ff8"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "0e56433f-d808-4669-821d-812769517ff8"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the VLAN to be deleted.

4.5.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response requestId="NA9/2014-04-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e"
 xmlns="urn:didata.com:api:cloud:types">
  <operation>DELETE_VLAN</operation>
  <responseCode>IN_PROGRESS</responseCode>
  <message>Request to Delete VLAN (Id:0e56433f-d808-4669-821d-812769517ff8)
has been accepted and is being processed</message>
</response>
```

JSON

```
{
    "operation": "DELETE_VLAN",
    "responseCode": "IN_PROGRESS",
    "message": "Request to Delete VLAN (Id:0e56433f-d808-4669-821d-
812769517ff8) has been accepted and is being processed",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-02-25T10:18:05.795-05:00/8c09329d-8aa9-4a05-b664-
3e88649faab7"
}
```

4.5.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
HAS_DEPENDENCY	Cannot delete the VLAN <id> because there is at least one NIC defined within the VLAN space. All such NICs must be removed before the VLAN can be deleted. *The response will include error elements describing up to 10 such dependencies; with the following properties:* • *nicServerId* • *nicServerName* • *nicId* • *nicIpv4Address* • *nicIpv6Address* Cannot delete the VLAN <id> because there is at least one NAT Rule or Node defined that falls into the VLAN address space. All such NAT rules or Nodes must be removed before deleting the VLAN. *The response will include error elements describing up to 10 such dependencies; with the following properties:* • *natRuleId* • *natRuleExternalIp* • *natRuleInternalIp* • *nodeId* • *nodeName* • *nodeIpAddress*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
RESOURCE_BUSY	Another operation is in progress on VLAN with id <id>. Please try again later. Another operation is in progress on Network Domain <networkDomainId>. Please try again later.
RESOURCE_LOCKED	VLAN with id <id> is locked. Please contact support. Network Domain with id <networkDomainId> is locked. Please contact support.
RESOURCE_NOT_FOUND	VLAN <id> not found. *The value passed for <id> does not correspond to a valid VLAN belonging to {org-id}. See **List VLANs**.*

4.6. Expand VLAN

4.6.1. Description

Increases the scale of a VLAN owned by {org-id} from one CIDR boundary to another. The expansion will not be permitted if the proposed IP space overlaps with an already deployed VLAN's IP space.

Note that the proposed IP space cannot include any IP address(es) already in use by a Virtual Listener within the same Network Domain.

4.6.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/expandVlan`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Asynchronous. For status refer to **Get VLAN** and **List VLANs**.
Roles	"primary administrator" and "network"

XML Request Sample

```
<expandVlan id="9add4863-e74d-4478-badd-3ed46057ee08"
xmlns="urn:didata.com:api:cloud:types">
  <privateIpv4PrefixSize>23</urn:privateIpv4PrefixSize>
</expandVlan>
```

JSON Request Sample

```
{
    "id": "9add4863-e74d-4478-badd-3ed46057ee08",
    "privateIpv4PrefixSize": 23
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the VLAN to be expanded.
privateIpv4PrefixSize	Yes	An Integer between 16 and 23, which represents the size of the VLAN after expansion and must be consistent with the privateIpv4BaseAddress of the VLAN.

4.6.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-02-
13T05:58:00.905-05:00/04a82e12-2d51-420e-84f5-bf2630c5a5e5">
    <operation>EXPAND_VLAN</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to expand VLAN 'Production VLAN' has been accepted and is
being processed.</message>
</response>
```

JSON

```
{
  "operation": "EXPAND_VLAN",
  "responseCode": "IN_PROGRESS",
  "message": "Request to expand VLAN 'Production VLAN' has been accepted and
is being processed.",
```

```
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-08-11T09:55:00.614-04:00/52cd35e6-ab9b-47eb-b1cb-
cff4f7d52efd"
}
```

4.6.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	id must be provided. privateIpv4PrefixSize must be provided. privateIpv4PrefixSize must be between 16-23. privateIpv4PrefixSize <privateIpv4PrefixSize> is incompatible with VLAN network address <ipv4BaseAddress>. Acceptable expansion prefix sizes for this address are: <comma separated list of acceptable sizes>. **OR** privateIpv4PrefixSize <privateIpv4PrefixSize> is incompatible with VLAN network address <ipv4BaseAddress>. The selected VLAN cannot be expanded. privateIpv4PrefixSize must be greater than the existing VLAN size (<original size>).
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
IP_ADDRESS_NOT_UNIQUE	The requested VLAN space <privateIpv4BaseAddress >/<privateIpv4PrefixSize > overlaps with space used by an existing VLAN <VLAN name> (<existing privateIpv4BaseAddress >/<existing privateIpv4PrefixSize >) in the same Network Domain. There is a NAT Rule/Node defined on the broadcast address (<x.y.z>.255) of the expanded VLAN range. It must be removed before expanding the VLAN to this size. *Response will include error elements with NAT Rule / Node identifying information.* Virtual Listener(s) ({virtual-listener-name(s)}) is listening on an address/addresses in the expanded VLAN subnet. You must remove any listed Virtual Listener before expanding the VLAN to this size. *Note that the response will include error elements describing up to 10 such dependencies.*
RESOURCE_BUSY	Another operation is in progress on VLAN with Id <id>. Please try again later. Another operation is in progress on Network Domain <networkDomainId>. Please try again later.
RESOURCE_LOCKED	Network Domain with id <networkDomainId> is locked. Please contact support. VLAN with id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	VLAN <id> not found. *The value passed for <id> does not correspond to a valid VLAN belonging to {org-id}. See **List VLANs**.*

5. Network API – IP Address Management

5.1. Add Public IPv4 Address Block

5.1.1. Description

Adds a block of public IPv4 addresses to the designated Network Domain.

5.1.2. Request Details

URL	https://<Cloud API URL>/caas/2.1/{org-id}/network/addPublicIpBlock
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous.
Roles	"primary administrator" and "network"

XML Request Sample

```
<addPublicIpBlock xmlns="urn:didata.com:api:cloud:types">
  <networkDomainId>484174a2-ae74-4658-9e56-50fc90e086cf</networkDomainId>
</addPublicIpBlock>
```

JSON Request Sample

```
{
    "networkDomainId": "484174a2-ae74-4658-9e56-50fc90e086cf"
}
```

Request Properties

Field	Required	Type and Constraints
networkDomainId	Yes	The "id" of the Network Domain belonging to {org-id} with which the Public IPv4 Address Block will be associated.

5.1.3. Response Details

The response includes an "*info*" property named "*publicIpBlockId*". The value of this property contains the unique identifier of the Public IPv4 Address Block which has been added to the Network Domain. The *publicIpBlockId* can be used in conjunction with **Get Public IPv4 Address Block** or **List Public IPv4 Address Block** to retrieve detailed information about the block.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response requestId="na/2014-04-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e"
 xmlns="urn:didata.com:api:cloud:types">
  <operation>ADD_PUBLIC_IP_BLOCK</operation>
  <responseCode>OK</responseCode>
  <message>Public IPv4 Address Block has been added successfully to Network
Domain <networkDomainId>.</message>
  <info name="publicIpBlockId" value="4487241a-f0ca-11e3-9315-d4bed9b167ba"/>
</response>
```

JSON

```
{
    "operation": "ADD_PUBLIC_IP_BLOCK",
    "responseCode": "OK",
    "message": "Public IPv4 Address Block has been added successfully to
```

```
Network Domain <networkDomainId>.",
    "info": [
        {
            "name": "publicIpBlockId",
            "value": "4487241a-f0ca-11e3-9315-d4bed9b167ba"
        }
    ],
    "warning": [],
    "error": [],
    "requestId": "NA9/2014-04-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e"
}
```

5.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
EXCEEDS_LIMIT*	No more Public IPv4 Address Blocks may be added because the maximum permitted ratio of Public IPv4 Addresses to Servers has already been reached.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	networkDomainId must be provided.
OPERATION_NOT_SUPPORTED	The API 2.1 "Add Public IP Block" function currently only supports MCP 2.0 data centers. If you need to add a Public IP Block to a Network in an MCP 1.0 data center, please use the "Reserve Public IP Block with Size" function under the 0.9 API.
OUT_OF_RESOURCES	No unassigned Public IPv4 Address Blocks currently available. Please try again. *Continual looping over this response is highly unlikely to succeed and should not be implemented. An approach such a short cycle (e.g. 3 or 4 iterations) of a second or two pause, followed by a retry attempt, is recommended. If the final attempt is unsuccessful then open a Support case.*
RESOURCE_BUSY	Another operation is in progress on Network Domain with id <networkDomainId>. Please try again later.
RESOURCE_LOCKED	Network Domain with id <networkDomainId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Network Domain <networkDomainId> not found. *The value passed for networkDomainId does not correspond to a valid Network Domain belonging to {org-id}. See **List Network Domains**.*

*If you receive an EXCEEDS_LIMIT response please refer to this Community article:

https://community.opsourcecloud.net/View.jsp?procId=1c328731f0cce9e588849ca0fa93e14a

5.2. List Public IPv4 Address Blocks

5.2.1. Description

Lists all of the Public IPv4 Address Blocks belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get Public IPv4 Address Block** than **List IPv4 Address Blocks** (filtered by *id*) if you wish to retrieve *state* status or details of a single block.

5.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/publicIpBlock?`
	
	`Filter Required Parameters:`
	` networkDomainId=`
	
	`Paging/Ordering Optional Parameters:`
	`  [&pageSize=]`
	`  [&pageNumber=]`
	`  [&orderBy=]`
	
	`Filter Optional Parameters:`
	`  [&id=]`
	`  [&datacenterId=]`
	`  [&baseIp=]`
	`  [&size=]`
	`  [&state=]`
	`  [&createTime=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	All roles

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual VLAN. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
networkDomainId	uuid (String)	Identifies an individual Network Domain. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes
baseIp	String	An IPv4 address in dot notation, corresponding to the *lowest* IPv4 address in the block to be matched. *baseIp=165.180.12.1*	Yes	Yes
size	Integer	Limits the blocks returned to those of	Yes	Yes

		the specified size. Note that the majority of datacenters provide blocks containing 2 IPv4 addresses. *size=2*		
state	String	Identifies VLANs by their *state*. Case insensitive. The initial possible set of values for *vlan.state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.	Yes	Yes
createTime	Date	Identifies the date of creation of the Public IP4v Address Block. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes

5.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<publicIpBlocks
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="3"
totalCount="3" pageSize="250">
    <publicIpBlock id="cacc028a-7f12-11e4-a91c-0030487e0302" location="NA9">
        <networkDomainId>802abc9f-45a7-4efb-9d5a-
810082368708</networkDomainId>
        <baseIp>165.180.12.12</baseIp>
        <size>2</size>
        <createTime>2014-12-15T16:35:07.000Z</createTime>
        <state>NORMAL</state>
    </publicIpBlock>
    <publicIpBlock id="bef128f8-844b-11e4-a91c-0030487e0302" location="NA9">
        <networkDomainId>802abc9f-45a7-4efb-9d5a-
810082368708</networkDomainId>
        <baseIp>165.180.12.14</baseIp>
        <size>2</size>
        <createTime>2015-02-26T14:19:30.000Z</createTime>
        <state>NORMAL</state>
    </publicIpBlock>
</publicIpBlocks>
```

JSON

```
{
    "publicIpBlock": [
        {
            "networkDomainId": "802abc9f-45a7-4efb-9d5a-810082368708",
            "baseIp": "165.180.12.12",
            "size": 2,
            "createTime": 1418661307000,
```

```
            "state": "NORMAL",
            "id": "cacc028a-7f12-11e4-a91c-0030487e0302",
            "datacenterId": "NA9"
        },
        {
            "networkDomainId": "802abc9f-45a7-4efb-9d5a-810082368708",
            "baseIp": "165.180.12.14",
            "size": 2,
            "createTime": 1424960370000,
            "state": "NORMAL",
            "id": "bef128f8-844b-11e4-a91c-0030487e0302",
            "datacenterId": "NA9"
        }
    ],
    "pageNumber": 1,
    "pageCount": 3,
    "totalCount": 3,
    "pageSize": 250
}
```


5.2.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

5.3. Get Public IPv4 Address Block

5.3.1. Description

Returns details of a single Public IPv4 Address Block belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get Public IPv4 Address Block** than **List Public IPv4 Address Block** (filtered by *id*) if you wish to retrieve *state* status or details of a single block.

5.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/publicIpBlock/{public-ipv4-address-block-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	All roles

5.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<publicIpBlock
    xmlns="urn:didata.com:api:cloud:types" id="cacc028a-7f12-11e4-a91c-0030487e0302" datacenterId="NA9">
    <networkDomainId>802abc9f-45a7-4efb-9d5a-810082368708</networkDomainId>
    <baseIp>165.180.12.12</baseIp>
    <size>2</size>
    <createTime>2014-12-15T16:35:07.000Z</createTime>
    <state>NORMAL</state>
</publicIpBlock>
```

JSON

```
{
    "networkDomainId": "802abc9f-45a7-4efb-9d5a-810082368708",
    "baseIp": "165.180.12.12",
    "size": 2,
    "createTime": "2014-12-15T16:35:07.000Z",
    "state": "NORMAL",
    "id": "cacc028a-7f12-11e4-a91c-0030487e0302",
    "datacenterId": "NA9"
}
```

5.3.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Public IPv4 Address Block <ipBlockId> not found. *The value passed for {public-ipv4-address-block-id} does not correspond to a valid Public IPv4 Address Block belonging to {org-id}. See **List Public IPv4 Address Blocks**.*

5.4. Remove Public IPv4 Address Block

5.4.1. Description

Removes a Public IPv4 Address Block from a Network Domain belonging to the organization identified by {org-id}.

5.4.2. Request Details

URL	https://<Cloud API URL>/caas/2.1/{org-id}/network/removePublicIpBlock
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<removePublicIpBlock xmlns="urn:didata.com:api:cloud:types" id="bef4334a-844b-11e4-a91c-0030487e0302" />
```

JSON Request Sample

```
{
  "id": "bef4334a-844b-11e4-a91c-0030487e0302"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Public IPv4 Address Block to be removed.

5.4.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-02-27T09:31:34.239-05:00/3e762559-c7f5-4d2b-abab-63306d406aa1">
    <operation>REMOVE_PUBLIC_IP_BLOCK</operation>
    <responseCode>OK</responseCode>
    <message>Public Ip Block bef4334a-844b-11e4-a91c-0030487e0302 has been removed successfully</message>
</response>
```

JSON

```
{
    "operation": "REMOVE_PUBLIC_IP_BLOCK",
    "responseCode": "OK",
    "message": "Public Ip Block bef4334a-844b-11e4-a91c-0030487e0302 has been removed successfully",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2014-04-14T13:37:20/62f06368-c3fb-11e3-b29c-001517c4643e"
}
```

5.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
HAS_DEPENDENCY	It is not possible to remove this Public IPv4 Address Block because there are associated VIPs or NATs.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	id must be provided.
OPERATION_NOT_SUPPORTED	The API 2.1 "Remove Public IPv4 Address Block" function currently only supports MCP 2.0 data centers. If you need to remove a IP Block from a Network in an MCP 1.0 data center, please use the "Release Public IP Block" function under the 0.9 API.
RESOURCE_BUSY	Another operation is in progress on IPv4 Address Block with id <ipBlockId>. Please try again later. Another operation is in progress on Network Domain with id <networkDomainId>. Please try again later.
RESOURCE_LOCKED	IPv4 Address Block with id <ipBlockId> is locked. Please contact support. Network Domain with id <networkDomainId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Public IPv4 Address Block <ipBlockId> not found. *The value passed for <id> does not correspond to a valid Public IPv4 Address Block belonging to {org-id}. See **List Public IPv4 Address Blocks**.*

5.5. List Reserved Public IPv4 Addresses

5.5.1. Description

Lists the Public IPv4 addresses which have been reserved for use on any NAT rules or VIPs belonging to the organization identified by {org-id}.

5.5.2. Request Details

URL	https://<Cloud API URL>/caas/2.1/{org-id}/network/reservedPublicIpv4Address? Filter *Required Parameters*. **One** of the following must be provided: networkId= **or** networkDomainId= Paging/Ordering Optional Parameters: [&pageSize=] [&pageNumber=] [&orderBy=] Filter Optional Parameters: [&ipBlockId=] [&ipAddress=]
Type	HTTP GET
Supported for	MCP 1.0 and MCP 2.0
Processing	Synchronous
Roles	All roles

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
networkId	uuid (String)	Identifies Cloud Networks (MCP 1.0). Supports the LIKE operator. *networkId=b2457b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
networkDomainId	uuid (String)	Identifies Network Domains (MCP 2.0). Supports the LIKE operator. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
ipBlockId	uuid (String)	Identifies Public IPv4 Address Blocks. Supports the LIKE operator. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
ipAddress	String	Identifies an individual Public IPv4 address in dot notation. *ipAddress=165.180.12.1*	Yes	Yes

5.5.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<reservedPublicIpv4Addresses
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="1"
totalCount="1" pageSize="250">
    <ip datacenterId="NA9" ipBlockId="cacc028a-7f12-11e4-a91c-0030487e0302"
networkDomainId="802abc9f-45a7-4efb-9d5a-810082368708">165.180.12.12</ip>
</reservedPublicIpv4Addresses>
```

JSON

```
{
    "ip": [
        {
            "value": "165.180.12.12",
            "datacenterId": "NA9",
            "ipBlockId": "cacc028a-7f12-11e4-a91c-0030487e0302",
            "networkDomainId": "802abc9f-45a7-4efb-9d5a-810082368708"
        }
    ],
    "pageNumber": 1,
    "pageCount": 1,
    "totalCount": 1,
    "pageSize": 250
}
```

5.5.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

5.6. List Reserved Private IPv4 Addresses

5.6.1. Description

Lists the Private IPv4 addresses which have been reserved for use by the organization identified by {org-id}. The response can be filtered by VLAN and Cloud Network to return just the IPv4 addresses reserved for the VLAN and/or Cloud Network identifiers provided.

5.6.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/reservedPrivateIpv4Address[?]`
	
	`Paging/Ordering Optional Parameters:`
	`[&pageSize=]`
	`[&pageNumber=]`
	`[&orderBy=]`
	
	`Filter Optional Parameters:`
	`[&networkId=]`
	`[&vlanId=]`
	`[&ipAddress=]`

Type	HTTP GET
Supported for	MCP 1.0 and MCP 2.0
Processing	Synchronous
Roles	All roles

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
networkId	uuid (String)	Identifies Cloud Networks (MCP 1.0). *networkId=b2457b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
vlanId	uuid (String)	Identifies VLANs (MCP 2.0). *vlanId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
ipAddress	String	Identifies an individual Private IPv4 address in dot notation. *ipAddress=165.180.12.1*	Yes	No

5.6.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<reservedPrivateIpv4Addresses
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="86"
totalCount="86" pageSize="250">
    <ipv4 datacenterId="NA9" vlanId="5d1d62c4-0627-4dc9-83a3-
985fbd82ff29">10.0.0.11</ipv4>
    <ipv4 datacenterId="NA9" vlanId="5d1d62c4-0627-4dc9-83a3-
985fbd82ff29">10.0.0.12</ipv4>
    <ipv4 datacenterId="NA5" networkId="50a8b326-2cbe-11e1-b5d6-
0030487e0302">10.157.101.12</ipv4>
    <ipv4 datacenterId="NA5" networkId="50a8b326-2cbe-11e1-b5d6-
```

```
0030487e0302">10.157.101.13</ipv4>
</reservedPrivateIpv4Addresses>
```

JSON

```
{
    "ipv4": [
        {
            "value": "10.0.0.11",
            "datacenterId": "NA9",
            "vlanId": "5d1d62c4-0627-4dc9-83a3-985fbd82ff29"
        },
        {
            "value": "10.0.0.12",
            "datacenterId": "NA9",
            "vlanId": "5d1d62c4-0627-4dc9-83a3-985fbd82ff29"
        },
        {
            "value": "10.157.101.12",
            "datacenterId": "NA5",
            "networkId": "50a8b326-2cbe-11e1-b5d6-0030487e0302"
        },
        {
            "value": "10.157.101.13",
            "datacenterId": "NA5",
            "networkId": "50a8b326-2cbe-11e1-b5d6-0030487e0302"
        }
    ],
    "pageNumber": 1,
    "pageCount": 86,
    "totalCount": 86,
    "pageSize": 250
}
```

5.6.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

6. Network API – Firewall Policy

6.1. List Firewall Rules

6.1.1. Description

Lists all of the Firewall Rules on a particular Network Domain at an MCP 2.0 data center, belonging to the organization identified by the {org-id} parameter. Various filter values can be used to narrow the list returned as detailed below.

Note that it is more efficient to use **Get Firewall Rule** than **List Firewall Rules** (filtered by *id*) if you wish to retrieve *state* status or details of a single Firewall Rule.

All Network Domains have a number of default rules designed to protect your resources until you wish to begin exposing particular ports and protocols. The number of default rules can change over time. Each default rule is clearly marked as such in two ways.

For integration use a ruleType field is provided, which will contain either DEFAULT_RULE if the rule is a default rule or CLIENT_RULE if the rule is one you have created.

For quick human reference, default rules also have a prefix to their name: "CCDEFAULT". It is *not* possible to create CLIENT_RULE rules which begin with "CCDEFAULT".

Refer to the Community for further details on Firewall Rule management:

https://community.opsourcecloud.net/View.jsp?procId=5048dec35c1d0153cf58401e91e3587f

6.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/firewallRule[?]`
	
	`Paging/Ordering Optional Parameters:`
	`[&pageSize=]`
	`[&pageNumber=]`
	`[&orderBy=]`
	
	`Filter Optional Parameters:`
	`[&id=]`
	`[&networkDomainId=]`
	`[&name=]`
	`[&createTime=]`
	`[&state=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual Firewall Rule. *id=a3457b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
networkDomainId	uuid (String)	Identifies an individual Network Domain. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No

name	String	Identifies Firewall Rules by their *name*. *name=MailRule* Supports the use of the LIKE comparator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	Yes
createTime	Date	Identifies the date of creation of Firewall Rules. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes
state	String	Identifies Firewall Rules by their *state*. Case insensitive. The initial possible set of values for *vlan.state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.	Yes	No

6.1.3. Response Details

Note that the sample responses below are truncated for brevity. Each sample displays a single default Firewall Rule as well the paging information. The elements returned for a given Firewall Rule can differ based on the definition of the rule. Refer to the XSD for details of the complete potential response structure.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<firewallRules
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="4"
totalCount="4" pageSize="50">
    <firewallRule id="1aa3d0ce-d95d-4296-8338-9717e0d37ff9" datacenterId="NA9"
ruleType="DEFAULT_RULE">
        <networkDomainId>484174a2-ae74-4658-9e56-
50fc90e086cf</networkDomainId>
        <name>CCDEFAULT.BlockOutboundMailIPv6Secure</name>
        <action>DROP</action>
        <ipVersion>IPV6</ipVersion>
        <protocol>TCP</protocol>
        <source>
            <ip address="ANY"/>
        </source>
        <destination>
            <ip address="ANY"/>
            <port begin="587"/>
        </destination>
        <enabled>true</enabled>
```

```
        <state>NORMAL</state>
    </firewallRule>
</firewallRules>
```

JSON

```
{
    "firewallRule": [
        {
            "networkDomainId": "484174a2-ae74-4658-9e56-50fc90e086cf",
            "name": "CCDEFAULT.BlockOutboundMailIPv6Secure",
            "action": "DROP",
            "ipVersion": "IPV6",
            "protocol": "TCP",
            "source": {
                "ip": {
                    "address": "ANY"
                }
            },
            "destination": {
                "ip": {
                    "address": "ANY"
                },
                "port": {
                    "begin": 587
                }
            },
            "enabled": true,
            "state": "NORMAL",
            "id": "1aa3d0ce-d95d-4296-8338-9717e0d37ff9",
            "datacenterId": "NA9",
            "ruleType": "DEFAULT_RULE"
        }
    ],
    "pageNumber": 1,
    "pageCount": 4,
    "totalCount": 4,
    "pageSize": 50
}
```

6.1.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

6.2. Create Firewall Rule

6.2.1. Description

This function creates a new CLIENT_RULE Firewall Rule on a Network Domain in an MCP 2.0 data center location.

Refer to the community for more information about creating Firewall Rules:

https://community.opsourcecloud.net/View.jsp?procId=20ee207afeef3937382cead8d9822773

6.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/createFirewallRule`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<createFirewallRule xmlns="urn:didata.com:api:cloud:types">
    <networkDomainId>484174a2-ae74-4658-9e56-50fc90e086cf</networkDomainId>
    <name>My.Rule.2</name>
    <action>ACCEPT_DECISIVELY</action>
    <ipVersion>IPV4</ipVersion>
    <protocol>TCP</protocol>
    <source>
      <ip address="10.2.0.0" prefixSize="16"/>
      <port begin="8888" end="8899"/>
    </source>
    <destination>
      <ip address="10.0.4.115"/>
    </destination>
    <enabled>true</enabled>
    <placement position="AFTER" relativeToRule="My.Rule"/>
</createFirewallRule>
```

JSON Request Sample

```
{
    "networkDomainId": "484174a2-ae74-4658-9e56-50fc90e086cf",
    "name": "My.Rule",
    "action": "ACCEPT_DECISIVELY",
    "ipVersion": "IPV4",
    "protocol": "TCP",
    "source": {
      "ip": {
        "address": "10.2.0.0",
        "prefixSize": "16"
      },
      "port": {
        "begin": "8866",
        "end": "8877"
      }
    },
    "destination": {
      "ip": { "address": "10.0.4.114" }
    },
    "enabled": "true",
    "placement": {
        "position": "FIRST"
    }
  }
}
```

Request Properties

Field	Required	Type and Constraints
networkDomainId	Yes	The “id” of a Network Domain belonging to {org-id} within the same MCP 2.0 data center.
name	Yes	Maximum length: 75 characters.
action	Yes	One of ACCEPT_DECISIVELY or DROP.
ipVersion	Yes	One of IPV4 or IPV6.
protocol	Yes	One of IP, ICMP, TCP or UDP.
source.ip.address source.ip.prefixSize	Yes	The *source.ip* fields are used to limit the range of sources of network traffic allowed by the Firewall Rule. source*.ip.address* can have three possible values: • "ANY", in which case prefixSize is not used and source traffic is acceptable from any IP address. • A valid IPv4 or IPv6 address depending on the *ipVersion* value selected, in which case traffic will be permitted from that specific IP address. • A valid IPv4 or IPv6 network address depending on the *ipVersion* value selected, with a prefixSize to define a range of addresses from which traffic will be permitted. • Note that it is **NOT** possible to specify *ANY* for both the *source.ip.address* and *destination.ip.address* if *ipVersion* is set to IPv6.
source.port.begin source.port.end	No	The *source.port* fields are used to define a single or range of source ports in conjunction with the *source.ip* values above. Note that a port range can only be specified on either the *source* or *destination* - not both. *source.port* fields are only expected if TCP or UDP is selected as the *protocol*. To define a single port, supply the *source.port.begin* field by itself. To define a range of ports, supply both the *source.port.begin* and *source.port.end* fields.
destination.ip.address destination.ip.prefixSize	Yes	The *destination.ip* fields are used to limit the range of destination for network traffic allowed by the Firewall Rule. *destination.ip.address* can have three possible values: • "ANY", in which case *prefixSize* is not used and source traffic is acceptable from any IP address. • A valid IPv4 or IPv6 address depending on the *ipVersion* value selected, in which case traffic will be permitted from that specific IP address. • A valid IPv4 or IPv6 network address depending on the *ipVersion* value selected, with a prefixSize to define a range of addresses from which traffic will be permitted. • Note that it is **NOT** possible to specify *ANY* for both the *source.ip.address* and *destination.ip.address* if *ipVersion* is set to IPv6.

destination.port.begin destination.port.end	No	The *destination.port* fields are used to define a single or range of destination ports in conjunction with the *destination.ip* values above. Note that a port range can only be specified on either the *source* or *destination* - not both. *destination.port* fields are only expected if TCP or UDP is selected as the *protocol*. To define a single port, supply the *destination.port.begin* field by itself. To define a range of ports, supply both the *destination.port.begin* and *destination.port.end* fields.
enabled	Yes	Defines whether or not this rule is enabled after being added to the Firewall Policy. Must be "true" or "false".
placement.position	Yes	One of FIRST, LAST, BEFORE or AFTER
placement.relativeToRule	No	If placement is set to BEFORE or AFTER, a *relativeToRule* field must be provided. The value of this should contain the name of an existing CLIENT_RULE on the Firewall Policy for the same Network Domain.

6.2.3. Response Details

The response includes an "*info*" property named "*firewallRuleId*". The value of this property contains the unique identifier of the VLAN being deployed and can be used in conjunction with **Get VLAN** and or **List VLANs** to retrieve the status of this asynchronous operation.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-03-
06T02:28:56.679-05:00/1003070b-dea9-41bd-9f0c-5c24086589b8">
    <operation>CREATE_FIREWALL_RULE</operation>
    <responseCode>OK</responseCode>
    <message>Request create Firewall Rule 'My.Rule.2' successful</message>
    <info name="firewallRuleId" value="d0a20f59-77b9-4f28-a63b-e58496b73a6c"/>
</response>
```

JSON

```
{
    "operation": "CREATE_FIREWALL_RULE",
    "responseCode": "OK",
    "message": "Request create Firewall Rule 'My.Rule' successful",
    "info": [
        {
            "name": "firewallRuleId",
            "value": "dc545f3e-823c-4500-93c9-8d7f576311de"
        }
    ],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-05T13:46:34.848-05:00/f8fdef24-8a12-45ea-a831-
d5463212ef6a"
}
```

6.2.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
EXCEEDS_LIMIT	You have reached the maximum number of 64 Firewall Rules specifying port range, which may be present in a Network Domain.
EXCEEDS_LIMIT	A Firewall Policy may only have a maximum of 1000 Firewall Rules.
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	Invalid *name*. Name is missing. Invalid *name*. Name includes spaces. Invalid *name*. Name includes invalid characters. Invalid *name*. Name starts with a number or '.'. Invalid *name*. Name exceeds maximum length of 75 characters. Invalid *name*. Name starts with "CCDEFAULT".
INVALID_INPUT_DATA	*action* can only be ACCEPT_DECISIVELY or DROP.
INVALID_INPUT_DATA	*ipVersion* can only be IPV4 or IPV6.
INVALID_INPUT_DATA	*protocol* can only be IP, ICMP, TCP or UDP.
INVALID_INPUT_DATA	Invalid *source*/*destination* IPv4 address {ip}. Must be a valid IPv4 in dot-decimal notation (x.x.x.x). Invalid *source*/*destination* IPv6 address {ip}. Must be a valid IPv6 address in compressed or extended format. prefixSize in *source*/*destination* is inconsistent with the IP address {ip}. You cannot specify ANY for both *source* and *destination* address for *ipVersion* IPV6.
INVALID_INPUT_DATA	You may specify a *port* range in *source* or *destination* but not in both.
INVALID_INPUT_DATA	You may not specify a *source* or *destination* port for IP or ICMP protocols.
INVALID_INPUT_DATA	*prefixSize* for {ip} IPv4 address has to be a number between 0-32. *prefixSize* for {ip} IPv6 address has to be a number between 0-128.
INVALID_INPUT_DATA	The maximum number of ports permitted within a range is 1024.
INVALID_INPUT_DATA	ICMP code and type must each be an integer between 0-255
INVALID_INPUT_DATA	ICMP properties can only be set when *protocol* is ICMP
NAME_NOT_UNIQUE	Another Firewall Rule named '<name>' already exists on Network Domain '<networkDomainName>'.
RESOURCE_BUSY	Another operation is in progress on Network Domain with id {id}. Please try again later. Another Firewall Rule operation is in progress on Network Domain with id {id}. Please try again later.
RESOURCE_LOCKED	Network Domain with id {id} is locked. Please contact support. Network Domain with id {id} is locked. Please contact support.
RESOURCE_NOT_FOUND	Cannot find *placement* Firewall Rule with name {name}. Cannot find Network Domain <id>.

6.3. Get Firewall Rule

6.3.1. Description

Returns details of a single Firewall Rule belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get Firewall Rule** than **List Firewall Rules** (filtered by *id*) if you wish to retrieve *state* status or details of a single Firewall Rule.

6.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/firewallRule/{firewall-rule-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

6.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<firewallRule
    xmlns="urn:didata.com:api:cloud:types" id="412f0e28-20f5-44ce-beb7-
385fb758a82c" datacenterId="NA9" ruleType="DEFAULT_RULE">
    <networkDomainId>484174a2-ae74-4658-9e56-50fc90e086cf</networkDomainId>
    <name>CCDEFAULT.BlockOutboundMailIPv4</name>
    <action>DROP</action>
    <ipVersion>IPV4</ipVersion>
    <protocol>TCP</protocol>
    <source>
        <ip address="ANY"/>
    </source>
    <destination>
        <ip address="ANY"/>
        <port begin="25"/>
    </destination>
    <enabled>true</enabled>
    <state>NORMAL</state>
</firewallRule>
```

JSON

```
{
    "networkDomainId": "484174a2-ae74-4658-9e56-50fc90e086cf",
    "name": "CCDEFAULT.BlockOutboundMailIPv4",
    "action": "DROP",
    "ipVersion": "IPV4",
    "protocol": "TCP",
    "source": {
        "ip": {
            "address": "ANY"
        }
    },
    "destination": {
        "ip": {
            "address": "ANY"
        },
        "port": {
            "begin": 25
        }
    },
    "enabled": true,
    "state": "NORMAL",
    "id": "412f0e28-20f5-44ce-beb7-385fb758a82c",
```

```
    "datacenterId": "NA9",
    "ruleType": "DEFAULT_RULE"
}
```

6.3.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Firewall Rule <firewall-rule-id> not found. *The value passed for {firewall-rule-id} does not correspond to a Firewall Rule belonging to {org-id}. See **List Firewall Rules**.*

6.4. Edit Firewall Rule

6.4.1. Description

This function allows the caller to enable or disable any of their own (CLIENT_RULE) Firewall Rules or any default (DEFAULT_RULE) Firewall Rules belonging to the organization identified by {org-id}.

Note that unverified organizations cannot use this function. Please contact your account representative for verification.

6.4.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/editFirewallRule`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<editFirewallRule id="1aa3d0ce-d95d-4296-8338-9717e0d37ff9"
xmlns="urn:didata.com:api:cloud:types">
    <enabled>true</enabled>
</editFirewallRule>
```

JSON Request Sample

```
{
    "id": "1aa3d0ce-d95d-4296-8338-9717e0d37ff9",
    "enabled": false
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Firewall Rule to be edited.
enabled	Yes	Must be "true" or "false". Toggles the enablement state of the Firewall Rule. Note that only a verified Organization can change Firewall Rule enablement.

6.4.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-03-
06T03:34:21.951-05:00/d7e37b41-ceff-4a8c-992d-a0220b8ebd76">
    <operation>EDIT_FIREWALL_RULE</operation>
    <responseCode>OK</responseCode>
    <message>Firewall Rule with id 1aa3d0ce-d95d-4296-8338-9717e0d37ff9 has
been edited</message>
</response>
```

JSON

```
{
    "operation": "EDIT_FIREWALL_RULE",
    "responseCode": "OK",
    "message": "Firewall Rule with id 1aa3d0ce-d95d-4296-8338-9717e0d37ff9 has
```

```
been edited",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-06T03:35:44.588-05:00/a51df6f7-0511-440a-8e59-
c939eb338d9a"
}
```

6.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	Firewall Rule <id> is required. <enabled> can only be "true" or "false".
RESOURCE_BUSY	Operation in progress on placement Firewall Rule <name> with id <id>. Please try again later. Another Firewall Rule operation is in progress on Network Domain with id {id}. Please try again later.
RESOURCE_LOCKED	Firewall Rule with id {id} is locked. Please contact support. Network Domain with id {id} is locked for Firewall Rule operations. Please contact support.
RESOURCE_NOT_FOUND	Cannot find Firewall Rule with Id <id>. *The value passed for <id> does not correspond to a valid Firewall Rule belonging to {org-id}. See **List Firewall Rules**.*

6.5. Delete Firewall Rule

6.5.1. Description

Deletes a Firewall Rule belonging to the organization identified by {org-id}.

Note you cannot delete default (DEFAULT_RULE) Firewall Rules. They can be disabled using the **Edit Firewall Rule** function.

6.5.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/deleteFirewallRule`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<deleteFirewallRule id="84e34850-595d-436e-a885-7cd37edb24a4"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
    "id": "84e34850-595d-436e-a885-7cd37edb24a4"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the CLIENT_TYPE Firewall Rule to be deleted.

6.5.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response requestId="NA9/2015-03-06T04:49:42.721-05:00/f69f1106-0df7-4614-
b517-844ce29013e4"
 xmlns="urn:didata.com:api:cloud:types">
  <operation>DELETE_FIREWALL_RULE</operation>
  <responseCode>OK</responseCode>
  <message>Firewall Rule (Id:84e34850-595d-436e-a885-7cd37edb24a4) has been
deleted</message>
</response>
```

JSON

```
{
    "operation": "DELETE_FIREWALL_RULE",
    "responseCode": "OK",
    "message": "Firewall Rule (Id:84e34850-595d-436e-a885-7cd37edb24a4) has
been deleted",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-06T04:49:42.721-05:00/f69f1106-0df7-4614-b517-
844ce29013e4"
}
```

6.5.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	id must be provided.
OPERATION_NOT_SUPPORTED	Only CLIENT_RULE Firewall Rules may be deleted.
RESOURCE_BUSY	Operation in progress on Firewall Rule with id {id}. Please try again later. Another Firewall Rule operation is in progress on Network Domain with id {id}. Please try again later.
RESOURCE_LOCKED	Firewall Rule with id {id} is locked. Please contact support. Network Domain with id {id} is locked for Firewall Rule operations. Please contact support.
RESOURCE_NOT_FOUND	Firewall Rule <id> not found. *The value passed for <id> does not correspond to a valid Firewall Rule belonging to {org-id}. See **List Firewall Rules**.*

7. Network API – Network Address Translation (NAT) Management

7.1. Create NAT Rule

7.1.1. Description

This function creates a NAT Rule on a Network Domain in an MCP 2.0 data center location.

Note that as part of NAT Rule creation, a Node is created implicitly. The Node can be managed independently (see **Edit Node**) if the Network Domain is an Advanced type Network Domain. See **Edit Network Domain** for a validation restriction related to the Node underlying a NAT Rule.

7.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/createNatRule`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<createNatRule xmlns="urn:didata.com:api:cloud:types">
    <networkDomainId>802abc9f-45a7-4efb-9d5a-810082368708</networkDomainId>
    <internalIp>10.0.0.15</internalIp>
    <externalIp>165.180.12.13</externalIp>
</createNatRule>
```

JSON Request Sample

```
{
    "networkDomainId": "802abc9f-45a7-4efb-9d5a-810082368708",
    "internalIp" : "10.0.0.16",
    "externalIp" : "165.180.12.17"
}
```

Request Properties

Field	Required	Type and Constraints
networkDomainId	Yes	The “id” of a Network Domain belonging to {org-id}.
internalIp	Yes	Cannot be in use by an existing Virtual Listener on the Network Domain. If the internalIp is within the address space of a VLAN deployed on the same Network Domain then internalIp cannot be one of the system reserved IP addresses within the space. The system reserved IP addresses are the first six (.0 - .5), eighth (.7) and broadcast address (the last .255 in the VLAN). If the internalIp is not within the address space of a VLAN deployed on the same Network Domain then any RFC1918 dot-decimal IPv4 address is acceptable.
externalIp	No	Dot-decimal IPv4 address. For example: “165.180.12.12”. If not provided on the request, an unreserved Public IPv4 address from the Public IPv4 Address Blocks belonging Network Domain will be automatically assigned.

		See **List Public IPv4 Address Blocks** See **List Reserved Public IPv4 Addresses**

7.1.3. Response Details

The response includes an "*info*" property named "*natRuleId*". The value of this property contains the unique identifier of the newly created NAT Rule and can be used in conjunction with **Get NAT Rule** and/or **List NAT Rule** to retrieve details about the NAT Rule.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-03-
06T05:53:37.334-05:00/fb842265-fa06-4ca9-a2af-a8a6815735fc">
    <operation>CREATE_NAT_RULE</operation>
    <responseCode>OK</responseCode>
    <message>NAT Rule with Id d31c2db0-be6b-4d50-8744-9a7a534b5fba has been
created.</message>
    <info name="natRuleId" value="d31c2db0-be6b-4d50-8744-9a7a534b5fba"/>
</response>
```

JSON

```
{
    "operation": "CREATE_NAT_RULE",
    "responseCode": "OK",
    "message": "NAT Rule with Id c95a30a1-6780-487d-b24f-5857ddb9c749 has been
created.",
    "info": [
        {
            "name": "natRuleId",
            "value": "c95a30a1-6780-487d-b24f-5857ddb9c749"
        }
    ],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-06T06:20:43.841-05:00/200922b6-ea8b-4209-89cd-
caabd547edb5"
}
```

7.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	<networkDomainId> must be provided. <internalIp> must be provided. internalIp <internalIp> is not a valid IPv4 address. externalIp <externalIp> is not a valid IPv4 address.
IP_ADDRESS_NOT_UNIQUE	internalIp <internalIp> is already in use in the Network Domain <id>. internalIp <internalIp> is already in use by a Virtual Listener(s). *The response will include error elements describing the dependency, with the following properties:* • *virtualListenerId*

	• *virtualListenerName* externalIp <ip> is already in use in the Network Domain <id>.
IP_ADDRESS_OUT_OF_RANGE	IPv4 address <internalIp> is outside the permitted range (<6th address in subnet> , and <8th address in subnet> to <broadcast address - 1> inclusive). *See note above about system reserved space.* externalIp <ip> does not exist within any of the Public IPv4 Blocks currently assigned to Network Domain <id>.
NO_IP_ADDRESS_AVAILABLE	None of the Public IPv4 Blocks currently assigned to Network Domain with Id <id> have any IP addresses available for use as an External IPv4 Address. See **Add Public IPv4 Address Block.**
OPERATION_NOT_SUPPORTED	Creating a NAT Rule on a Cloud Network is not supported by this API version. Please use the 0.9 API.
RESOURCE_BUSY	Another operation is in progress on Network Domain <id>.
RESOURCE_LOCKED	Network Domain <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Cannot find Network Domain with Id <networkDomainId>. *The value passed for networkDomainId does not correspond to a valid Network Domain belonging to {org-id}. See **List Network Domains**.*

7.2. List NAT Rules

7.2.1. Description

Lists all of the NAT Rules on a particular Network Domain at an MCP 2.0 data center, belonging to the organization identified by the {org-id} parameter. Various filter values can be used to narrow the list returned as detailed below.

Note that it is more efficient to use **Get NAT Rule** than **List NAT Rules** (filtered by *id*) if you wish to retrieve details for a single NAT.

7.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/natRule?`
	
	`Filter Required Parameters:`
	`  networkDomainId=`
	
	`Filter Optional Parameters:`
	`  [&id=]`
	`  [&state=]`
	`  [&createTime=]`
	`  [&internalIp=]`
	`  [&externalIp=]`
	`  [&nodeId=]`
	
	`Paging/Ordering Optional Parameters:`
	`  [&pageSize=]`
	`  [&pageNumber=]`
	`  [&orderBy=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
networkDomainId	uuid (String)	**Required**. Identifies an individual Network Domain. Supports the LIKE operator. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
id	uuid (String)	Identifies an individual NAT Rule. Supports the LIKE operator. *id=j6ghb7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
state	String	Identifies NAT Rules by their *state*. Case insensitive. The initial possible set of values for *state* are: "NORMAL", "PENDING_ADD",	Yes	Yes

		"PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.		
createTime	Date	Identifies the date of creation of NAT Rules. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes
internalIp	String	Identifies internal IPv4 address addresses. *internalIp=10.1.1.15*	Yes	Yes
externalIp	String	Identifies external IPv4 addresses. *externalIp=165.180.12.17*	Yes	Yes
nodeId	uuid (String)	Identifies a Node (See **List Nodes**) and can be used to determine if a Node belongs to a NAT. *nodeId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes

7.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<natRules
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="2"
totalCount="2" pageSize="250">
    <natRule id="2187a636-7ebb-49a1-a2ff-5d617f496dce" datacenterId="NA9">
        <networkDomainId>484174a2-ae74-4658-9e56-
50fc90e086cf</networkDomainId>
        <internalIp>10.0.0.15</internalIp>
        <externalIp>165.180.12.18</externalIp>
        <createTime>2015-03-06T13:43:45.000Z</createTime>
        <state>NORMAL</state>
    </natRule>
    <natRule id="2169a38e-5692-497e-a22a-701a838a6539" datacenterId="NA9">
        <networkDomainId>484174a2-ae74-4658-9e56-
50fc90e086cf</networkDomainId>
        <internalIp>10.0.0.16</internalIp>
        <externalIp>165.180.12.19</externalIp>
        <createTime>2015-03-06T13:45:10.000Z</createTime>
        <state>NORMAL</state>
    </natRule>
</natRules>
```

JSON

```
{
    "natRule": [
        {
```

```
            "networkDomainId": "484174a2-ae74-4658-9e56-50fc90e086cf",
            "internalIp": "10.0.0.15",
            "externalIp": "165.180.12.18",
            "createTime": "2015-03-06T13:43:45.000Z",
            "state": "NORMAL",
            "id": "2187a636-7ebb-49a1-a2ff-5d617f496dce",
            "datacenterId": "NA9"
        },
        {
            "networkDomainId": "484174a2-ae74-4658-9e56-50fc90e086cf",
            "internalIp": "10.0.0.16",
            "externalIp": "165.180.12.19",
            "createTime": "2015-03-06T13:45:10.000Z",
            "state": "NORMAL",
            "id": "2169a38e-5692-497e-a22a-701a838a6539",
            "datacenterId": "NA9"
        }
    ],
    "pageNumber": 1,
    "pageCount": 2,
    "totalCount": 2,
    "pageSize": 250
}
```


7.2.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

Response Code	Message
RESOURCE_NOT_FOUND	NAT Rule <id> not found. *The value passed for {id} does not correspond to a NAT Rule belonging to {org-id}. See* ***List NAT Rules*** unfiltered.

7.3. Get NAT Rule

7.3.1. Description

Returns details of a single NAT Rule belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get NAT Rule** than **List NAT Rules** (filtered by *id*) if you wish to retrieve *state* status or details for a single NAT Rule.

7.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/natRule/{nat-rule-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

7.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<natRule
    xmlns="urn:didata.com:api:cloud:types" id="2169a38e-5692-497e-a22a-
701a838a6539" datacenterId="NA9">
    <networkDomainId>484174a2-ae74-4658-9e56-50fc90e086cf</networkDomainId>
    <internalIp>10.0.0.16</internalIp>
    <externalIp>165.180.12.19</externalIp>
    <createTime>2015-03-06T13:45:10.000Z</createTime>
    <state>NORMAL</state>
</natRule>
```

JSON

```
{
    "networkDomainId": "484174a2-ae74-4658-9e56-50fc90e086cf",
    "internalIp": "10.0.0.16",
    "externalIp": "165.180.12.19",
    "createTime": "2015-03-06T13:45:10.000Z",
    "state": "NORMAL",
    "id": "2169a38e-5692-497e-a22a-701a838a6539",
    "datacenterId": "NA9"
}
```

7.3.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Cannot find NAT Rule {nat-rule-id}. *The value passed for {nat-rule-id} does not correspond to a valid NAT Rule belonging to {org-id}. See **List NAT Rules**.*

7.4. Delete NAT Rule

7.4.1. Description

Deletes a NAT Rule belonging to the organization identified by {org-id}.

7.4.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/network/deleteNatRule`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<deleteNatRule id="2187a636-7ebb-49a1-a2ff-5d617f496dce"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "2187a636-7ebb-49a1-a2ff-5d617f496dce"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the NAT Rule to be deleted.

7.4.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-03-
06T12:24:57.820-05:00/5974fe37-bac5-4345-8a88-4c93a874bf4a">
    <operation>DELETE_NAT_RULE</operation>
    <responseCode>OK</responseCode>
    <message>NAT Rule with Id 2187a636-7ebb-49a1-a2ff-5d617f496dce has been
deleted.</message>
</response>
```

JSON

```
{
    "operation": "DELETE_NAT_RULE",
    "responseCode": "OK",
    "message": "NAT Rule with Id 2169a38e-5692-497e-a22a-701a838a6539 has been
deleted.",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-06T12:25:39.042-05:00/cc04a610-bdcb-477f-af3d-
1215c7e57b70"
}
```

7.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	NAT Rule id must be provided.
RESOURCE_BUSY	Another operation is in progress on NAT Rule Id <id>.
RESOURCE_LOCKED	NAT Rule with id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Cannot find NAT Rule < id>. *The value passed for <id> does not correspond to a valid NAT Rule belonging to {org-id}. See **List NAT Rules**.*

8. Network API - VIPs - Node Management

A Node is a low level component of a VIP and a NAT in an MCP 2.0 data center.

A Node is added to a Pool (see **Create Pool**) as a Pool Member (see **Add Pool Member**), where it can receive designated network traffic from a Virtual Listener (see **Create Virtual Listener**).

Node Management is restricted to *Advanced* type Network Domains.

8.1. Create Node

8.1.1. Description

This function creates a Node on a Network Domain in an MCP 2.0 data center location.

The properties of a Nodes are described in detail in the Community article How to Create a Node on a Network Domain.

8.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/createNode`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous.
Roles	"primary administrator" and "network"

XML Request Sample

```
<createNode xmlns="urn:didata.com:api:cloud:types">
    <networkDomainId>553f26b6-2a73-42c3-a78b-6116f11291d0</networkDomainId>
    <name>myProductionNode.1</name>
    <description>Production Server 1</description>
    <ipv4Address>10.5.2.14</ipv4Address>
    <status>ENABLED</status>
    <healthMonitorId>0168b83a-d487-11e4-811f-005056806999</healthMonitorId>
    <connectionLimit>20000</connectionLimit>
    <connectionRateLimit>2000</connectionRateLimit>
</createNode>
```

JSON Request Sample

```
{
    "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
    "name": "myProductionNode.1",
    "description": "Production Server 1",
    "ipv4Address": "10.5.2.14",
    "status": "ENABLED",
    "healthMonitorId": "0168b83a-d487-11e4-811f-005056806999",
    "connectionLimit": "20000",
    "connectionRateLimit": "2000"
}
```

Request Properties

Field	Required	Type and Constraints
networkDomainId	Yes	The “id” of a Network Domain belonging to {org-id} within the same MCP 2.0 data center.
name	Yes	Must be alphanumeric with the following exceptions permitted "_" (underscore) and "." (full stop / period). Cannot begin with a "." or a number.

		Cannot contain spaces. Minimum length: 1 character. Maximum length: 75 characters.
description	No	Maximum length: 255 characters.
ipv4Address **or** ipv6Address	Yes	Cannot be in use by another Node on the Network Domain. **ipv4Address:** Cannot be in use by an existing Virtual Listener on the Network Domain. If the ipv4Address is within the address space of a VLAN deployed on the same Network Domain then internalIp cannot be one of the system reserved IP addresses within the space. The system reserved IP addresses are the first six (.0 - .5), eighth (.7) and broadcast address (the last .255 in the VLAN). If the ipv4Address is not within the address space of a VLAN deployed on the same Network Domain then any RFC1918 dot-decimal IPv4 address is acceptable. **ipv6Address:** Must be a valid IPv6 address in compressed or extended format. Site-local unicast addresses are not permitted. Link-local unicast addresses are not permitted. The first quad cannot be all zeros.
status	Yes	Must be one of ENABLED, DISABLED or FORCED_OFFLINE.
healthMonitorId	No	UUID corresponding to a *nodeCompatible* Health Monitor returned by the **List Default Health Monitors** API function.
connectionLimit	Yes	An integer in the range 1 - MAX_NODE_CONNECTION_LIMIT. See **List Data Centers** for the MAX_NODE_CONNECTION_LIMIT property.
connectionRateLimit	Yes	An integer in the range 1 - MAX_NODE_CONNECTION_RATE_LIMIT. See **List Data Centers** for the MAX_NODE_CONNECTION_RATE_LIMIT property.

8.1.3. Response Details

The response includes an “*info*” property named “*nodeId*”. The value of this property contains the unique identifier of the new Node and can be used in conjunction with **Get Node** and or **List Nodes** to retrieve details about the Node later. The UUID of each Node is also relevant in the context of Pools with the **Add Pool Member** API function.

An "*info*" property "*name*" is also included, carrying the value of the *name* property on the Create Node request, which can be useful for display purposes.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response requestId="na/2015-04-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e" xmlns="urn:didata.com:api:cloud:types">
    <operation>CREATE_NODE</operation>
    <responseCode>OK</responseCode>
    <message>Node 'myProductionNode.1' has been created.</message>
    <info name="nodeId" value="9e6b496d-5261-4542-91aa-b50c7f569c54"/>
    <info name="name" value="myProductionNode.1"/>
</response>
```

JSON

```
{
    "requestId": "na/2015-04-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e",
    "operation": "CREATE_NODE",
    "responseCode": "OK",
    "message": "Node 'myProductionNode.1' has been created.",
    "info": [{
      "name": "nodeId",
      "value": "9e6b496d-5261-4542-91aa-b50c7f569c54"
    },
    {
      "name": "name",
      "value": "myProductionNode.1"
    }]
}
```

8.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	The Health Monitor <healthMonitorName> (<healthMonitorId>) is not supported for Nodes. *See **List Default Health Monitors** (nodeCompatible).*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INCOMPATIBLE_OPERATION	Operation not available in Network Domain <networkDomainType>.
INVALID_INPUT_DATA	name must be provided. Invalid name: name must not include spaces. Invalid name: name can only include alphanumeric characters, '_' and '.'. Invalid name: name must not start with a number or '.'. Invalid name: exceeds maximum length of 75 characters. Invalid name: name must be provided and must be at least 1 character in length. Invalid description: exceeds maximum length of 255 characters. Invalid IPv4 address <ipv4Address>. Must be a valid IPv4 in dot-decimal notation (x.x.x.x). Invalid IPv6 address <ipv6Address>. Must be a valid IPv6 address in compressed or extended format. Only one of IPv4 or IPv6 address may be specified. connectionLimit must be an integer in the range 1-100000 inclusive. connectionRateLimit must be an integer in the range 1-4000 inclusive. status must be one of ENABLED, DISABLED, FORCED_OFFLINE.

IP_ADDRESS_NOT_UNIQUE	IP Address <ipv4Address/ipv6Address> is already in use in the Network Domain with Id <id>. IP address <ipv4Address> is already in use by a Virtual Listener(s). *The response will include error elements describing the dependency, with the following properties:* • *virtualListenerId* • *virtualListenerName*
IP_ADDRESS_OUT_OF_RANGE	IPv4 address <ipv4Address> is outside the permitted range (<6th address in subnet> , and <8th address in subnet> to <broadcast address - 1> inclusive). *See note above about system reserved space.* Invalid IPv6 address <ip>. Site-local unicast addresses are not allowed. Invalid IPv6 address<ip>. Link-local unicast addresses are not allowed. Invalid IPv6 address <ip>. First quad cannot be all zeros.
NAME_NOT_UNIQUE	Another Node named '<name>' already exists in network domain <networkDomainId>.
RESOURCE_BUSY	Another operation is in progress on Network Domain with id <networkDomainId>. Please try again later.
RESOURCE_LOCKED	Network Domain with id <networkDomainId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Network Domain <networkDomainId> not found. *See **List Network Domains**.* Health Monitor <healthMonitorId> not found. *See **List Default Health Monitors** (nodeCompatible).*

8.2. List Nodes

8.2.1. Description

Lists all of the Nodes on a particular Network Domain at an MCP 2.0 data center, belonging to the organization identified by the {org-id} parameter. Various filter values can be used to narrow the list returned as detailed below.

Note that it is more efficient to use **Get Node** than **List Nodes** (filtered by *id*) if you wish to retrieve the details of a single Node.

8.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/node[?]`
	
	`Paging/Ordering Optional Parameters:`
	`[&pageSize=]`
	`[&pageNumber=]`
	`[&orderBy=]`
	
	`Filter Optional Parameters:`
	`[&id=]`
	`[&networkDomainId=]`
	`[&datacenterId=]`
	`[&name=]`
	`[&state=]`
	`[&createTime=]`
	`[&ipv4Address=]`
	`[&ipv6Address=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual Node. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
networkDomainId	uuid (String)	Identifies an individual Network Domain. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes
name	String	Identifies Nodes by their *name*. *name=Production.Node.1* Supports the use of the LIKE comparator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	Yes

state	String	Identifies Nodes by their *state*. Case insensitive. The initial possible set of values for *node.state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.	Yes	No
createTime	Date	Identifies the date of creation of Nodes. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes
ipv4Address	String	Identifies Nodes by their specific ipv4Address. *ipv4Address=10.10.10.15*	Yes	No
ipv6Address	String	Identifies Nodes by their specific ipv6Address. *ipv6Address=2607:f480:1111:1384:2662:abd9:66f3:e302*	Yes	No

8.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<nodes xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="1"
totalCount="2" pageSize="250">
    <node id="34de6ed6-46a4-4dae-a753-2f8d3840c6f9" datacenterId="NA9">
        <networkDomainId>553f26b6-2a73-42c3-a78b-
6116f11291d0</networkDomainId>
        <name>ProductionNode.1</name>
        <description>Production Server 1</description>
        <ipv4Address>10.10.10.101</ipv4Address>
        <state>NORMAL</state>
        <status>ENABLED</status>
        <healthMonitor id="0168b83a-d487-11e4-811f-005056806999" name="ICMP"/>
        <connectionLimit>10000</connectionLimit>
        <connectionRateLimit>2000</connectionRateLimit>
        <createTime>2015-05-29T13:56:13.000Z</createTime>
    </node>
    <node id="78d2437e-36e1-4dd5-9bfd-582a85d207ad" datacenterId="NA9">
        <networkDomainId>553f26b6-2a73-42c3-a78b-
6116f11291d0</networkDomainId>
        <name>ProductionNode.2</name>
        <description>Production Server 2</description>
        <ipv4Address>10.5.2.15</ipv4Address>
        <state>NORMAL</state>
        <status>ENABLED</status>
        <connectionLimit>10000</connectionLimit>
        <connectionRateLimit>2000</connectionRateLimit>
        <createTime>2015-05-29T11:46:55.000Z</createTime>
    </node>
</nodes>
```

JSON

```
{
```

```
  "node": [
    {
      "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
      "name": "ProductionNode.1",
      "description": "Production Server 1",
      "ipv4Address": "10.10.10.123",
      "state": "NORMAL",
      "status": "ENABLED",
      "connectionLimit": 10000,
      "connectionRateLimit": 1000,
      "createTime": "2015-05-22T14:31:10.000Z",
      "id": "6586aaa8-efe3-48ae-a2c1-3c7ad7024cd3",
      "datacenterId": "NA9"
    },
    {
      "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
      "name": "ProductionNode.2",
      "description": "Production Server 2",
      "ipv4Address": "10.10.10.101",
      "state": "NORMAL",
      "status": "ENABLED",
      "healthMonitor": {
        "id": "0168b83a-d487-11e4-811f-005056806999",
        "name": "ICMP"
      },
      "connectionLimit": 10000,
      "connectionRateLimit": 2000,
      "createTime": "2015-05-27T13:56:13.000Z",
      "id": "34de6ed6-46a4-4dae-a753-2f8d3840c6f9",
      "datacenterId": "NA9"
    }
  ],
  "pageNumber": 1,
  "pageCount": 2,
  "totalCount": 2,
  "pageSize": 250
}
```


8.2.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

8.3. Get Node

8.3.1. Description

Returns details of a single Node belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get Node** than **List Nodes** (filtered by *id*) if you wish to retrieve *state* status or details of a single Node.

8.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/node/{node-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

8.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<node xmlns="urn:didata.com:api:cloud:types" id="34de6ed6-46a4-4dae-a753-
2f8d3840c6f9" datacenterId="NA9">
    <networkDomainId>553f26b6-2a73-42c3-a78b-6116f11291d0</networkDomainId>
    <name>ProductionNode.2</name>
    <description>Production Server 2</description>
    <ipv4Address>10.10.10.101</ipv4Address>
    <state>NORMAL</state>
    <status>ENABLED</status>
    <healthMonitor id="0168b83a-d487-11e4-811f-005056806999" name="ICMP"/>
    <connectionLimit>10000</connectionLimit>
    <connectionRateLimit>2000</connectionRateLimit>
    <createTime>2015-05-27T13:56:13.000Z</createTime>
</node>
```

JSON

```
{
  "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
  "name": "ProductionNode.2",
  "description": "Production Server 2",
  "ipv4Address": "10.10.10.101",
  "state": "NORMAL",
  "status": "ENABLED",
  "healthMonitor": {
    "id": "0168b83a-d487-11e4-811f-005056806999",
    "name": "ICMP"
  },
  "connectionLimit": 10000,
  "connectionRateLimit": 2000,
  "createTime": "2015-05-27T13:56:13.000Z",
  "id": "34de6ed6-46a4-4dae-a753-2f8d3840c6f9",
  "datacenterId": "NA9"
}
```

8.3.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Node {node-id} not found.

	*The value passed for {node-id} does not correspond to a valid Node belonging to {org-id}. See **List Nodes**.*

8.4. Edit Node

8.4.1. Description

Updates the mutable properties of a Node belonging to the organization identified by {org-id}.

8.4.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/editNode`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

Note the use of "nil" in the samples below. In both cases, the payload is deleting the contents of *description* and *healthMonitorId* properties. See **Considerations for Edit API Functions** for further information on updates to optional fields.

XML Request Sample

```
<editNode id="34de6ed6-46a4-4dae-a753-2f8d3840c6f9"
xmlns="urn:didata.com:api:cloud:types">
  <description xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"/>
  <status>ENABLED</status>
  <healthMonitorId xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"/>
  <connectionLimit>15000</connectionLimit>
  <connectionRateLimit>2600</connectionRateLimit>
</editNode>
```

JSON Request Sample

```
{
    "id": "34de6ed6-46a4-4dae-a753-2f8d3840c6f9",
    "description": {
      "nil": "true"
    },
    "status": "ENABLED",
    "healthMonitorId": {
      "nil": "true"
    },
    "connectionLimit": "20000",
    "connectionRateLimit": "3500"
}
```

Request Properties

Field	Required	Type and Constraints
description	No	Maximum length: 255 characters.
status	No	Must be one of ENABLED, DISABLED or FORCED_OFFLINE.
healthMonitorId	No	UUID corresponding to a *nodeCompatible* Health Monitor returned by the **List Default Health Monitors** API function.
connectionLimit	No	An integer in the range 1 - MAX_NODE_CONNECTION_LIMIT. See **List Data Centers** for the MAX_NODE_CONNECTION_LIMIT property.
connectionRateLimit	No	An integer in the range 1 - MAX_NODE_CONNECTION_RATE_LIMIT.

		See **List Data Centers** for the MAX_NODE_CONNECTION_RATE_LIMIT property.

8.4.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-05-
29T11:57:29.793-04:00/9f7191df-de3a-4cf3-a0a1-6321e88b5af5">
    <operation>EDIT_NODE</operation>
    <responseCode>OK</responseCode>
    <message>Node 'ProductionNode.2' has been edited successfully.</message>
    <info name="nodeId" value="34de6ed6-46a4-4dae-a753-2f8d3840c6f9"/>
</response>
```

JSON

```
{
  "operation": "EDIT_NODE",
  "responseCode": "OK",
  "message": "Node 'ProductionNode.2' has been edited successfully.",
  "info": [
    {
      "name": "nodeId",
      "value": "34de6ed6-46a4-4dae-a753-2f8d3840c6f9"
    }
  ],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-05-29T11:55:54.006-04:00/eeb00057-c6ab-4606-a960-
dcf43c4c8bea"
}
```

8.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	The Health Monitor <healthMonitorName> (<healthMonitorId>) is not supported for Nodes. *See **List Default Health Monitors** (nodeCompatible).*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INCOMPATIBLE_OPERATION	Operation not available in Network Domain <networkDomainType>.
INVALID_INPUT_DATA	Invalid description: exceeds maximum length of 255 characters. connectionLimit must be an integer in the range 1-100000 inclusive. connectionRateLimit must be an integer in the range 1-4000 inclusive. status must be one of ENABLED, DISABLED, FORCED_OFFLINE.
RESOURCE_BUSY	Another operation is in progress on Network Domain with id <networkDomainId>. Please try again later.
RESOURCE_LOCKED	Network Domain with id <networkDomainId>

	is locked. Please contact support.
RESOURCE_NOT_FOUND	Network Domain <networkDomainId> not found. *See **List Network Domains**.* Health Monitor <healthMonitorId> not found. *See **List Default Health Monitors** (nodeCompatible).*

8.5. Delete Node

8.5.1. Description

Deletes a Node belonging to the organization identified by {org-id}.

8.5.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/deleteNode`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous.
Roles	"primary administrator" and "network"

XML Request Sample

```
<deleteNode id="34de6ed6-46a4-4dae-a753-2f8d3840c6f9"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "34de6ed6-46a4-4dae-a753-2f8d3840c6f9"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Node to be deleted.

8.5.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response requestId="NA9/2015-05-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e"
 xmlns="urn:didata.com:api:cloud:types">
  <operation>DELETE_NODE</operation>
  <responseCode>OK</responseCode>
  <message>Node (id:34de6ed6-46a4-4dae-a753-2f8d3840c6f9) has been
deleted.</message>
</response>
```

JSON

```
{
  "operation": "DELETE_NODE",
  "responseCode": "OK",
  "message": "Node (Id:34de6ed6-46a4-4dae-a753-2f8d3840c6f9) has been
deleted",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-05-29T12:10:56.433-04:00/cfbb1c31-31ac-4f7c-a635-
1d7419668e1a"
}
```

8.5.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
HAS_DEPENDENCY	Node {nodeName} cannot be deleted. It is a member of Pool(s): poolName0 ({poolId0}), poolName1 ({poold1}). Node {nodeName} cannot be deleted. It is associated with NAT Rule: {natRuleExternalIp} ({natRuleId}). *A Node cannot be deleted if it is still a Pool Member of a Pool or if it is associated with a NAT Rule. See **Remove Pool Member** and **Delete NAT Rule** respectively.*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
RESOURCE_BUSY	Another operation is in progress on Node with Id <id>. Please try again later.
RESOURCE_LOCKED	Node with id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Node <id> not found. *The value passed for <id> does not correspond to a valid Node belonging to {org-id}. See **List Nodes**.*

9. Network API - VIPs - Pool Management

A Pool is a fundamental component of a VIP used to group Nodes so that they can be addressed through a Virtual Listener according to the rules defined by the Virtual Listener and Pool.

A Pool can be added to a Virtual Listener, as either a *pool* or *clientClonePool*, where *pool* is the primary recipient of the traffic and the *clientClonePool* receives a clone of the *same* traffic for additional processing such as intrusion detection.

Pool Management is restricted to *Advanced* type Network Domains.

For further information refer to:

- Network API - VIPs - Node Management
- Network API - VIPs – Virtual Listener Management

9.1. Create Pool

9.1.1. Description

This function creates a Pool on a Network Domain in an MCP 2.0 data center location.

Up to 100 Pools can be created on a single Network Domain.

The properties of a Pool are described in detail in the Community article: How to Create a Pool on a Network Domain.

9.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/createPool`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous.
Roles	"primary administrator" and "network"

XML Request Sample

```
<createPool xmlns="urn:didata.com:api:cloud:types">
  <networkDomainId>553f26b6-2a73-42c3-a78b-6116f11291d0</networkDomainId>
  <name>myProductionPool.1</name>
  <description>Pool for load balancing production application
servers.</description>
  <loadBalanceMethod>ROUND_ROBIN</loadBalanceMethod>
  <healthMonitorId>e33a97fc-ff8b-4808-97ea-50b05624fd11</healthMonitorId>
  <healthMonitorId>e44a97fc-ff8z-4809-97ea-50b05624fd12</healthMonitorId>
  <serviceDownAction>NONE</serviceDownAction>
  <slowRampTime>10</slowRampTime>
</createPool>
```

JSON Request Sample

```
{
  "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
  "name": "myDevelopmentPool.1",
  "description": "Pool for load balancing development application servers.",
  "loadBalanceMethod": "ROUND_ROBIN",
  "healthMonitorId": [
        "01683574-d487-11e4-811f-005056806999",
        "0168546c-d487-11e4-811f-005056806999"
        ],
  "serviceDownAction": "RESELECT",
  "slowRampTime": 10
}
```

Request Properties

Field	Required	Type and Constraints
networkDomainId	Yes	The “id” of the Network Domain belonging to {org-id}, which the Pool will belong to.
name	Yes	Must be alphanumeric with the following exceptions permitted "_" (underscore) and "." (full stop / period). Cannot begin with a "." or a number. Cannot contain spaces. Minimum length: 1 character. Maximum length: 75 characters.
description	No	Maximum length: 255 characters.
loadBalanceMethod	Yes	Must be one of: • ROUND_ROBIN • LEAST_CONNECTIONS_MEMBER • LEAST_CONNECTIONS_NODE • OBSERVED_MEMBER • OBSERVED_NODE • PREDICTIVE_MEMBER • PREDICTIVE_NODE.
healthMonitorId	No	UUID corresponding to a *poolCompatible* Health Monitor returned by the **List Default Health Monitors** API function. Up to two Health Monitors may be specified for a single pool. Note that Health Monitors are only supported for Pools with Pool Members on specific ports. If you want to add Pool Members with the "Any" port option, Health Monitors cannot be set.
serviceDownAction	Yes	Must be one of NONE, DROP or RESELECT.
slowRampTime	Yes	An integer in the range 1-120 (seconds).

9.1.3. Response Details

The response includes an “*info*” property named “*poolId*”. The value of this property contains the unique identifier of the new Pool and can be used in conjunction with **Get Pool** and or **List Pools** to retrieve details about the Pool later. The UUID of each Pool is also relevant in the context of Virtual Listeners with the **Create Virtual Listener** and **Edit Virtual Listener** API functions, for the *pool and clientClonePool* properties.

An "*info*" property "*name*" is also included, carrying the value of the *name* property on the Create Pool request, which can be useful for display purposes.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response requestId="na/2015-04-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e" xmlns="urn:didata.com:api:cloud:types">
    <operation>CREATE_POOL</operation>
    <responseCode>OK</responseCode>
    <message>Pool 'myProductionPool.1' has been created.</message>
    <info name="poolId" value="9e6b496d-5261-4542-91aa-b50c7f569c54"/>
    <info name="name" value="myProductionPool.1"/>
</response>
```

JSON

```
{
    "requestId": "na/2015-04-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e",
    "operation": "CREATE_POOL",
    "responseCode": "OK",
    "message": "Pool 'myDevelopmentPool.1' has been created.",
    "info": [{
      "name": "poolId",
      "value": "4d360b1f-bc2c-4ab7-9884-1f03ba2768f7"
    },
    {
      "name": "name",
      "value": " myDevelopmentPool.1"
    }]
}
```

9.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	The Health Monitor <healthMonitorName> (<healthMonitorId>) is not supported for Pools. *See **List Default Health Monitors** (poolCompatible**).***
EXCEEDS_LIMIT	The maximum permitted number (100) of Pools that can be created on a Network Domain has already been reached.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INCOMPATIBLE_OPERATION	Operation not available in Network Domain <networkDomainType>.
INVALID_INPUT_DATA	name must be provided. Invalid name: name must not include spaces. Invalid name: name can only include alphanumeric characters, '_' and '.'. Invalid name: name must not start with a number or '.'. Invalid name: exceeds maximum length of 75 characters. Invalid name: name must be provided and must be at least 1 character in length. Invalid description: exceeds maximum length of 255 characters. The loadBalanceMethod must be one of: (see values in *Request Properties* above). Health Monitor <healthMonitorId> supplied multiple times. slowRampTime must be an integer in the range <RANGE> inclusive. The serviceDownAction must be one of: (see values in *Request Properties* above).
NAME_NOT_UNIQUE	Another Pool named '<name>' already exists in network domain <networkDomainId>.

RESOURCE_BUSY	Another operation is in progress on Network Domain with id <networkDomainId>. Please try again later.
RESOURCE_LOCKED	Network Domain with id <networkDomainId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Network Domain <networkDomainId> not found. *See **List Network Domains**.* Health Monitor <healthMonitorId> not found. *See **List Default Health Monitors** (poolCompatible).*

9.2. List Pools

9.2.1. Description

Lists all of the Pools on a particular Network Domain at an MCP 2.0 data center, belonging to the organization identified by the {org-id} parameter. Various filter values can be used to narrow the list returned as detailed below.

Note that it is more efficient to use **Get Pool** than **List Pools** (filtered by *id*) if you wish to retrieve the details of a single Pool.

9.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/pool[?]`
	
	`Paging/Ordering Optional Parameters:`
	`[&pageSize=]`
	`[&pageNumber=]`
	`[&orderBy=]`
	
	`Filter Optional Parameters:`
	`[&id=]`
	`[&networkDomainId=]`
	`[&datacenterId=]`
	`[&name=]`
	`[&state=]`
	`[&createTime=]`
	`[&loadBalanceMethod=]`
	`[&slowRampTime=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual Pool. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
networkDomainId	uuid (String)	Identifies an individual Network Domain. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes
name	String	Identifies Pools by their *name*. *name=Production.Pool.1* Supports the use of the LIKE comparator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	Yes

state	String	Identifies Pools by their *state*. Case insensitive. The initial possible set of values for *node.state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.	Yes	No
createTime	Date	Identifies the date of creation of Pools. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes
loadBalanceMethod	String	Filters the list to Pools with the supplied loadBalanceMethod(s). *loadBalanceMethod=ROUND_ROBIN*	Yes	No
serviceDownAction	String	Filters the list to Pools with the supplied serviceDownAction(s). *serviceDownAction=DROP*	Yes	No
slowRampTime	Integer	Filters the list to Pools with the supplied slowRampTime(s). Supports MIN, MAX, LT and GT. *slowRampTime.GT=10* Refer to further samples in **Paging and Filtering for List API Functions.**	Yes	No

9.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<pools
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="2"
totalCount="2" pageSize="250">
     <pool id="4d360b1f-bc2c-4ab7-9884-1f03ba2768f7" datacenterId="NA9">
        <networkDomainId>553f26b6-2a73-42c3-a78b-
6116f11291d0</networkDomainId>
        <name>myDevelopmentPool.1</name>
        <description>Pool for load balancing development application
servers.</description>
        <loadBalanceMethod>ROUND_ROBIN</loadBalanceMethod>
        <healthMonitor id="01683574-d487-11e4-811f-005056806999"
name="CCDEFAULT.Http"/>
        <healthMonitor id="0168546c-d487-11e4-811f-005056806999"
name="CCDEFAULT.Https"/>
        <serviceDownAction>RESELECT</serviceDownAction>
```

```
        <slowRampTime>10</slowRampTime>
        <state>NORMAL</state>
        <createTime>2015-06-04T09:15:07.000Z</createTime>
    </pool>
    <pool id="afb1fb1a-eab9-43f4-95c2-36a4cdda6cb8" datacenterId="NA9">
        <networkDomainId>553f26b6-2a73-42c3-a78b-
6116f11291d0</networkDomainId>
        <name>myProductionPool.1</name>
        <description>Pool for load balancing production application
servers.</description>
        <loadBalanceMethod>ROUND_ROBIN</loadBalanceMethod>
        <healthMonitor id="01683574-d487-11e4-811f-005056806999"
name="CCDEFAULT.Http"/>
        <healthMonitor id="0168546c-d487-11e4-811f-005056806999"
name="CCDEFAULT.Https"/>
        <serviceDownAction>NONE</serviceDownAction>
        <slowRampTime>10</slowRampTime>
        <state>NORMAL</state>
        <createTime>2015-06-03T14:11:17.000Z</createTime>
    </pool>
</pools>
```

JSON

```
{
  "pool": [
    {
      "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
      "name": "myDevelopmentPool.1",
      "description": "Pool for load balancing development application
servers.",
      "loadBalanceMethod": "ROUND_ROBIN",
      "healthMonitor": [
        {
          "id": "01683574-d487-11e4-811f-005056806999",
          "name": "CCDEFAULT.Http"
        },
        {
          "id": "0168546c-d487-11e4-811f-005056806999",
          "name": "CCDEFAULT.Https"
        }
      ],
      "serviceDownAction": "RESELECT",
      "slowRampTime": 10,
      "state": "NORMAL",
      "createTime": "2015-06-04T09:15:07.000Z",
      "id": "4d360b1f-bc2c-4ab7-9884-1f03ba2768f7",
      "datacenterId": "NA9"
    },
    {
      "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
      "name": "myProductionPool.1",
      "description": "Pool for load balancing production application
servers.",
      "loadBalanceMethod": "ROUND_ROBIN",
      "healthMonitor": [
        {
          "id": "01683574-d487-11e4-811f-005056806999",
          "name": "CCDEFAULT.Http"
        },
        {
          "id": "0168546c-d487-11e4-811f-005056806999",
          "name": "CCDEFAULT.Https"
        }
      ],
      "serviceDownAction": "NONE",
      "slowRampTime": 10,
      "state": "NORMAL",
```

```
      "createTime": "2015-06-03T14:11:17.000Z",
      "id": "afb1fb1a-eab9-43f4-95c2-36a4cdda6cb8",
      "datacenterId": "NA9"
    }
  ],
  "pageNumber": 1,
  "pageCount": 2,
  "totalCount": 2,
  "pageSize": 250
}
```


9.2.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

9.3. Get Pool

9.3.1. Description

Returns details of a single Pool belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get Pool** than **List Pools** (filtered by *id*) if you wish to retrieve *state* status or details of a single Pool.

9.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/pool/{pool-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

9.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<pool
    xmlns="urn:didata.com:api:cloud:types" id="4d360b1f-bc2c-4ab7-9884-
1f03ba2768f7" datacenterId="NA9">
    <networkDomainId>553f26b6-2a73-42c3-a78b-6116f11291d0</networkDomainId>
    <name>myDevelopmentPool.1</name>
    <description>Pool for load balancing development application
servers.</description>
    <loadBalanceMethod>ROUND_ROBIN</loadBalanceMethod>
    <healthMonitor id="01683574-d487-11e4-811f-005056806999"
name="CCDEFAULT.Http"/>
    <healthMonitor id="0168546c-d487-11e4-811f-005056806999"
name="CCDEFAULT.Https"/>
    <serviceDownAction>RESELECT</serviceDownAction>
    <slowRampTime>10</slowRampTime>
    <state>NORMAL</state>
    <createTime>2015-06-04T09:15:07.000Z</createTime>
</pool>
```

JSON

```
{
  "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
  "name": "myDevelopmentPool.1",
  "description": "Pool for load balancing development application servers.",
  "loadBalanceMethod": "ROUND_ROBIN",
  "healthMonitor": [
    {
      "id": "01683574-d487-11e4-811f-005056806999",
      "name": "CCDEFAULT.Http"
    },
    {
      "id": "0168546c-d487-11e4-811f-005056806999",
      "name": "CCDEFAULT.Https"
    }
  ],
  "serviceDownAction": "RESELECT",
  "slowRampTime": 10,
  "state": "NORMAL",
  "createTime": "2015-06-04T09:15:07.000Z",
  "id": "4d360b1f-bc2c-4ab7-9884-1f03ba2768f7",
  "datacenterId": "NA9"
}
```

9.3.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Pool {pool-id} not found. *The value passed for {pool-id} does not correspond to a valid Pool belonging to {org-id}. See **List Pools**.*

9.4. Edit Pool

9.4.1. Description

Updates the mutable properties of a Pool belonging to the organization identified by {org-id}.

9.4.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/editPool`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

Note the use of "nil" in the samples below. In both cases, the payload is deleting the contents of *description* and *healthMonitorId* properties. See **Considerations for Edit API Functions** for further information on updates to optional fields.

XML Request Sample

```
<editPool id="4d360b1f-bc2c-4ab7-9884-1f03ba2768f7"
xmlns="urn:didata.com:api:cloud:types">
  <description xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"/>
  <loadBalanceMethod>ROUND_ROBIN</loadBalanceMethod>
  <healthMonitorId xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"/>
  <serviceDownAction>DROP</serviceDownAction>
  <slowRampTime>10</slowRampTime>
</editPool>
```

JSON Request Sample

```
{
  "id": "4d360b1f-bc2c-4ab7-9884-1f03ba2768f7",
  "description": {
    "nil": "true"
  },
  "loadBalanceMethod": "ROUND_ROBIN",
  "healthMonitorId": {
     "nil": "true"
  },
  "serviceDownAction": "DROP",
  "slowRampTime": 10
}
```

Request Properties

Field	Required	Type and Constraints
description	No	Maximum length: 255 characters.
loadBalanceMethod	No	Must be one: • ROUND_ROBIN • LEAST_CONNECTIONS_MEMBER • LEAST_CONNECTIONS_NODE • OBSERVED_MEMBER • OBSERVED_NODE • PREDICTIVE_MEMBER

		• PREDICTIVE_NODE.
healthMonitorId	No	UUID corresponding to a *poolCompatible* Health Monitor returned by the **List Default Health Monitors** API function. Note that when editing a Pool, the presence of any Pool Members with the "Any" port option will block the ability to set Health Monitors on the Pool. Health Monitors are only supported for Pools with Pool Members on specific ports.
serviceDownAction	No	Must be one of: NONE, DROP and RESELECT.
slowRampTime	No	An integer in the range 1-120 (seconds).

9.4.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-05-
29T11:57:29.793-04:00/9f7191df-de3a-4cf3-a0a1-6321e88b5af5">
    <operation>EDIT_POOL</operation>
    <responseCode>OK</responseCode>
    <message>Pool 'ProductionPool.2' has been updated.</message>
    <info name="name" value="ProductionPool.2"/>
    <info name="poolId" value="4d360b1f-bc2c-4ab7-9884-1f03ba2768f7"/>
</response>
```

JSON

```
{
  "operation": "EDIT_POOL",
  "responseCode": "OK",
  "message": "Pool 'ProductionPool.2' has been updated.",
  "info": [
    {
      "name": "poolId",
      "value": "4d360b1f-bc2c-4ab7-9884-1f03ba2768f7"
    },
    {
      "name": "name",
      "value": "ProductionPool.2"
    }
  ],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-05-29T11:55:54.006-04:00/eeb00057-c6ab-4606-a960-
dcf43c4c8bea"
}
```

9.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	The Health Monitor <healthMonitorName> (<healthMonitorId>) is not supported for Pools. *See **List Default Health Monitors** (poolCompatible).* Health Monitors are not supported unless all Pool Members in the pool have a specific port set.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INCOMPATIBLE_OPERATION	Operation not available in Network Domain

	<networkDomainType>.
INVALID_INPUT_DATA	id must be provided. At least one of description, loadBalanceMethod, healthMonitorId, serviceDownAction, or slowRampTime must be supplied. Invalid description: exceeds maximum length of 255 characters. loadBalanceMethod must be one of: (see values in *Request Properties* above). slowRampTime must be an integer in the range 1-120 inclusive. serviceDownAction must be one of: (see values in *Request Properties* above). Health Monitor <healthMonitorId> supplied multiple times. You may not supply more than 2 healthMonitorId values. In order to remove all Health Monitors from a Pool, you must supply a single healthMonitorId with nil="'true". A second healthMonitorId value must not be supplied in this case. See the sample payloads above and refer to the general ***Considerations for Edit API Functions*** section.
RESOURCE_BUSY	Another operation is in progress on Pool with id <id>. Please try again later.
RESOURCE_LOCKED	Pool with id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Pool with id <id> not found. See ***List Pools***. Health Monitor <healthMonitorId> not found. *See* ***List Default Health Monitors*** *(poolCompatible).*

9.5. Delete Pool

9.5.1. Description

Deletes a Pool belonging to the organization identified by {org-id}.

If there are Pool Members associated with the Pool, they will be removed by this action. However the underlying Nodes are NOT modified in any way.

A Pool cannot be deleted if it is associated with a Virtual Listener. See ***Edit Virtual Listener*** and ***Delete Virtual Listener***.

9.5.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/deletePool`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous.
Roles	"primary administrator" and "network"

XML Request Sample

```
<deletePool id="4d360b1f-bc2c-4ab7-9884-1f03ba2768f7"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "4d360b1f-bc2c-4ab7-9884-1f03ba2768f7"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Pool to be deleted.

9.5.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response requestId="NA9/2015-05-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e"
 xmlns="urn:didata.com:api:cloud:types">
  <operation>DELETE_POOL</operation>
  <responseCode>OK</responseCode>
  <message>Pool (id:4d360b1f-bc2c-4ab7-9884-1f03ba2768f7) has been
deleted.</message>
</response>
```

JSON

```
{
  "operation": "DELETE_POOL",
  "responseCode": "OK",
  "message": "Pool (Id:4d360b1f-bc2c-4ab7-9884-1f03ba2768f7) has been
deleted",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-05-29T12:10:56.433-04:00/cfbb1c31-31ac-4f7c-a635-
1d7419668e1a"
```

```
}
```

9.5.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
HAS_DEPENDENCY	Pool with id <id> cannot be deleted because it is associated with one or more Virtual Listeners. See ***List Virtual Listeners*** *(filter by poolId)*.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	Pool id is required.
RESOURCE_BUSY	Another operation is in progress on Pool with Id <id>. Please try again later. Operation in progress on one or more Pool Members of Pool with id {poolId}. Please try again later. See ***List Pool Members*** *(filter by poolId)*.
RESOURCE_LOCKED	Pool with id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Cannot find Pool with id: <id>. *The value passed for <id> does not correspond to a valid Pool belonging to {org-id}. See* ***List Pools****.*

9.6. Add Pool Member

9.6.1. Description

Adds a Node belonging to the organization identified by {org-id}, combined with Port information to the identified Pool as a new Pool Member. The Pool Member can respond to traffic directed to the Pool by a Virtual Listener, as determined by its *status* property.

9.6.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/addPoolMember`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous.
Roles	"primary administrator" and "network"

XML Request Sample

```
<addPoolMember xmlns="urn:didata.com:api:cloud:types">
    <poolId>6f2f5d7b-cdd9-4d84-8ad7-999b64a87978</poolId>
    <nodeId>3c207269-e75e-11e4-811f-005056806999</nodeId>
    <port>9888</port>
    <status>ENABLED</status>
</addPoolMember>
```

JSON Request Sample

```
{
  "poolId": "6f2f5d7b-cdd9-4d84-8ad7-999b64a87978",
  "nodeId": "3c207269-e75e-11e4-811f-005056806999",
  "port": 9888,
  "status": "ENABLED"
}
```

Request Properties

Field	Required	Type and Constraints
poolId	Yes	Identifies the Pool which the new Pool Member will be added to.
nodeId	Yes	Identifies a Node on the same Network Domain as Pool <poolId> which will receive traffic through the new Pool Member.
status	Yes	Identifies the enablement status of the Pool Member when added. Must be one of: • ENABLED • DISABLED • FORCED_OFFLINE.
port	No	Identifies the port on which the Pool Member (and hence the Node) will receive traffic. If no port property is provided "Any Port" is assumed as the port behavior. If present on an XML request, must be positioned between *nodeId* and *status*. Note that this property is influenced by the presence of Health Monitors on the target Pool (*poolId*). If the Pool has *any* Health Monitors then each Pool Member **must** have a Port specified.

9.6.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-06-
09T07:02:49.563-04:00/40653bb1-cf55-4ba8-ba00-121c19c50a54">
    <operation>ADD_POOL_MEMBER</operation>
    <responseCode>OK</responseCode>
    <message>Pool Member '10.0.3.13:9888' has been added.</message>
    <info name="poolMemberId" value="3dd806a2-c2c8-4c0c-9a4f-5219ea9266c0"/>
    <info name="nodeName" value="10.0.3.13"/>
</response>
```

JSON

```
{
  "operation": "ADD_POOL_MEMBER",
  "responseCode": "OK",
  "message": "Pool Member '10.0.3.13:9888' has been added.",
  "info": [
    {
      "name": "poolMemberId",
      "value": "b977578b-a827-4172-b285-030c3ba15daa"
    },
    {
      "name": "nodeName",
      "value": "10.0.3.13"
    }
  ],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-06-09T06:43:28.824-04:00/bd05f7fa-a019-47db-85fd-
44d77edf0302"
}
```

9.6.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	Node <nodeId> has already been added to the Pool <poolId> with the port <port>. Node <nodeId> has already been added to the Pool <poolId> with ANY port. A Pool Member cannot be added with ANY port if the Pool has one or more Health Monitors.
EXCEEDS_LIMIT	A Pool may only have a maximum of <maximum> Pool Members.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	poolId must be provided. nodeId must be provided. status must be provided. port <port> is outside the permitted range (1 to 65535). Status must be one of ENABLED, DISABLED, FORCED_OFFLINE.

RESOURCE_BUSY	Another operation is in progress on Pool with id <poolId>. Please try again later. Another operation is in progress on Node with id <nodeId>. Please try again later.
RESOURCE_LOCKED	Pool with id <poolId> is locked. Please contact support. Node with id <nodeId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Pool <poolId> not found. *See **List Pools**.* Node <nodeId> not found. *See **List Nodes**.* Node <nodeId> and Pool <poolId> do not belong to the same Network Domain.

9.7. List Pool Members

9.7.1. Description

Lists the Pool Members on a particular Network Domain at an MCP 2.0 data center, belonging to the organization identified by the {org-id} parameter. Various filter values can be used to narrow the list returned as detailed below.

poolId is probably the most useful filter for this API function, to return all of the Pool Members associated with a single Pool.

Note that it is more efficient to use **Get Pool Member** than **List Pool Members** (filtered by *id*) if you wish to retrieve the details of a single Pool Member.

9.7.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/poolMember[?]`
	
	`Paging/Ordering Optional Parameters:`
	`[&pageSize=]`
	`[&pageNumber=]`
	`[&orderBy=]`
	
	`Filter Optional Parameters:`
	`[&id=]`
	`[&networkDomainId=]`
	`[&datacenterId=]`
	`[&poolId=]`
	`[&poolName=]`
	`[&nodeId=]`
	`[&nodeName=]`
	`[&nodeIp=]`
	`[&nodeStatus=]`
	`[&port=]`
	`[&status=]`
	`[&state=]`
	`[&createTime=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual Pool Member. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
networkDomainId	uuid (String)	Identifies an individual Network Domain. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes

poolId	uuid (String)	Identifies an individual Pool. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
poolName	String	Identifies Pools by their *name*. *name=Production.Pool.1* *S*upports the use of the LIKE comparator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	Yes
nodeId	uuid (String)	Identifies an individual Node. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
nodeName	String	Identifies Nodes by their *name*. *name=Production.Node.1* *S*upports the use of the LIKE comparator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	Yes
nodeIp	uuid (String)	Identifies Nodes by their *ipv4Address* **or** by their *ipv6Address*. *nodeIp=10.0.0.3* *nodeIp= 2607:f480:1111:1388:3c32:69f:a4f0:cf1a* See ***Create Node*** and ***List Nodes***.	Yes	No
nodeStatus	String	Identifies Nodes by their *status*. *nodeStatus=ENABLED* See ***Create Node*** for the full set of values.	Yes	No
port	Integer	Identifies Pool Members by their *port* value. *port=9888* Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	No
status	String	Identifies Pool Members by their *status*. status=ENABLED See ***Add Pool Member*** for the full set of values.	Yes	No
state	String	Identifies Pool Members by their *state*. Case insensitive. The initial possible set of values for *node.state* are: "NORMAL", "PENDING_ADD",	Yes	No

		"PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.		
createTime	Date	Identifies the date of creation of Pool Members. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes

9.7.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<poolMembers
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="2"
totalCount="2" pageSize="250">
    <poolMember id="3dd806a2-c2c8-4c0c-9a4f-5219ea9266c0" datacenterId="NA9">
        <networkDomainId>553f26b6-2a73-42c3-a78b-
6116f11291d0</networkDomainId>
        <pool id="6f2f5d7b-cdd9-4d84-8ad7-999b64a87978"
name="myDevelopmentPool.1"/>
        <node id="3c207269-e75e-11e4-811f-005056806999" name="10.0.3.13"
ipAddress="10.0.3.13" status="ENABLED"/>
        <port>9889</port>
        <status>ENABLED</status>
        <state>NORMAL</state>
        <createTime>2015-06-09T11:02:50.000Z</createTime>
    </poolMember>
    <poolMember id="b977578b-a827-4172-b285-030c3ba15daa" datacenterId="NA9">
        <networkDomainId>553f26b6-2a73-42c3-a78b-
6116f11291d0</networkDomainId>
        <pool id="6f2f5d7b-cdd9-4d84-8ad7-999b64a87978"
name="myDevelopmentPool.1"/>
        <node id="3c207269-e75e-11e4-811f-005056806999" name="10.0.3.13"
ipAddress="10.0.3.13" status="ENABLED"/>
        <port>9888</port>
        <status>ENABLED</status>
        <state>NORMAL</state>
        <createTime>2015-06-09T10:43:29.000Z</createTime>
    </poolMember>
</poolMembers>
```

JSON

```
{
  "poolMember": [
    {
      "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
      "pool": {
        "id": "6f2f5d7b-cdd9-4d84-8ad7-999b64a87978",
        "name": "myDevelopmentPool.1"
      },
      "node": {
        "id": "3c207269-e75e-11e4-811f-005056806999",
        "name": "10.0.3.13",
```

```
        "ipAddress": "10.0.3.13",
        "status": "ENABLED"
      },
      "port": 9889,
      "status": "ENABLED",
      "state": "NORMAL",
      "createTime": "2015-06-09T11:02:50.000Z",
      "id": "3dd806a2-c2c8-4c0c-9a4f-5219ea9266c0",
      "datacenterId": "NA9"
    },
    {
      "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
      "pool": {
        "id": "6f2f5d7b-cdd9-4d84-8ad7-999b64a87978",
        "name": "myDevelopmentPool.1"
      },
      "node": {
        "id": "3c207269-e75e-11e4-811f-005056806999",
        "name": "10.0.3.13",
        "ipAddress": "10.0.3.13",
        "status": "ENABLED"
      },
      "port": 9888,
      "status": "ENABLED",
      "state": "NORMAL",
      "createTime": "2015-06-09T10:43:29.000Z",
      "id": "b977578b-a827-4172-b285-030c3ba15daa",
      "datacenterId": "NA9"
    }
  ],
  "pageNumber": 1,
  "pageCount": 2,
  "totalCount": 2,
  "pageSize": 250
}
```


9.7.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

9.8. Get Pool Member

9.8.1. Description

Returns details of a single Pool Member belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get Pool Member** than **List Pool Members** (filtered by *id*) if you wish to retrieve *state* status or details of a single Pool Member.

9.8.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/poolMember/{pool-member-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

9.8.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<poolMember
    xmlns="urn:didata.com:api:cloud:types" id="3dd806a2-c2c8-4c0c-9a4f-
5219ea9266c0" datacenterId="NA9">
    <networkDomainId>553f26b6-2a73-42c3-a78b-6116f11291d0</networkDomainId>
    <pool id="6f2f5d7b-cdd9-4d84-8ad7-999b64a87978"
name="myDevelopmentPool.1"/>
    <node id="3c207269-e75e-11e4-811f-005056806999" name="10.0.3.13"
ipAddress="10.0.3.13" status="ENABLED"/>
    <port>9889</port>
    <status>ENABLED</status>
    <state>NORMAL</state>
    <createTime>2015-06-09T11:02:50.000Z</createTime>
</poolMember>
```

JSON

```
{
  "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
  "pool": {
    "id": "6f2f5d7b-cdd9-4d84-8ad7-999b64a87978",
    "name": "myDevelopmentPool.1"
  },
  "node": {
    "id": "3c207269-e75e-11e4-811f-005056806999",
    "name": "10.0.3.13",
    "ipAddress": "10.0.3.13",
    "status": "ENABLED"
  },
  "port": 9889,
  "status": "ENABLED",
  "state": "NORMAL",
  "createTime": "2015-06-09T11:02:50.000Z",
  "id": "3dd806a2-c2c8-4c0c-9a4f-5219ea9266c0",
  "datacenterId": "NA9"
}
```

9.8.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Pool Member {pool-member-id} not found. *The value passed for {pool-member-id} does not correspond to a valid Pool Member belonging to {org-id}. See **List Pool Members**.*

9.9. Edit Pool Member

9.9.1. Description

Updates the status of a Pool Member belonging to the organization identified by {org-id}.

9.9.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/editPoolMember`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<editPoolMember id="3dd806a2-c2c8-4c0c-9a4f-5219ea9266c0"
xmlns="urn:didata.com:api:cloud:types">
  <status>ENABLED</status>
</editPoolMember>
```

JSON Request Sample

```
{
    "id": "3dd806a2-c2c8-4c0c-9a4f-5219ea9266c0",
    "status": "ENABLED"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	UUID of the Pool Member being edited. See ***List Pool Members***.
status	Yes	Must be one of: • ENABLED • DISABLED • FORCED_OFFLINE.

9.9.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-06-
09T09:29:46.249-04:00/c128c717-b3b4-480a-ba93-e3fb2e32a9be">
    <operation>EDIT_POOL_MEMBER</operation>
    <responseCode>OK</responseCode>
    <message>Pool Member (10.0.3.13:9889) has been edited.</message>
    <info name="poolMemberId" value="3dd806a2-c2c8-4c0c-9a4f-5219ea9266c0"/>
</response>
```

JSON

```
{
  "operation": "EDIT_POOL_MEMBER",
  "responseCode": "OK",
  "message": "Pool Member (10.0.3.13:9889) has been edited.",
```

```
  "info": [
    {
      "name": "poolMemberId",
      "value": "3dd806a2-c2c8-4c0c-9a4f-5219ea9266c0"
    }
  ],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-06-09T09:27:45.282-04:00/bc799b11-7178-425c-8d8e-
906ab5af80f8"
}
```


9.9.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	id must be provided. status must be provided. status must be one of ENABLED, DISABLED, FORCED_OFFLINE
RESOURCE_BUSY	Another operation is in progress on Pool Member with id <poolMemberId>. Please try again later. Another operation is in progress on Pool with id <poolId>. Please try again later.
RESOURCE_LOCKED	Pool Member with id <poolMemberId> is locked. Please contact support. Pool with id <poolId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Pool Member <poolMemberId> not found. See ***List Pool Members***.

9.10. Remove Pool Member

9.10.1. Description

Removes a Pool Member belonging to the organization identified by {org-id}.

Note that this function does NOT remove the underlying Node.

9.10.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/removePoolMember`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous.
Roles	"primary administrator" and "network"

XML Request Sample

```
<removePoolMember id="34de6ed6-46a4-4dae-a753-2f8d3840c6f9"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "34de6ed6-46a4-4dae-a753-2f8d3840c6f9"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Pool Member to be removed.

9.10.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response requestId="NA9/2015-05-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e"
 xmlns="urn:didata.com:api:cloud:types">
  <operation>REMOVE_POOL_MEMBER</operation>
  <responseCode>OK</responseCode>
  <message>Pool Member (id:34de6ed6-46a4-4dae-a753-2f8d3840c6f9) has been
removed.</message>
</response>
```

JSON

```
{
  "operation": " REMOVE_POOL_MEMBER",
  "responseCode": "OK",
  "message": "Pool Member (Id:34de6ed6-46a4-4dae-a753-2f8d3840c6f9) has been
removed.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-05-29T12:10:56.433-04:00/cfbb1c31-31ac-4f7c-a635-
1d7419668e1a"
}
```

9.10.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	Pool Member id is required.
RESOURCE_BUSY	Another operation is in progress on Pool with id {poolId}. Please try again later. Another operation is in progress on Pool Member with id {poolMemberId}. Please try again later.
RESOURCE_LOCKED	Pool with id {poolId} is locked. Please contact support. Pool Member with id {poolMemberId} is locked. Please contact support.
RESOURCE_NOT_FOUND	Pool Member <id> not found. *The value passed for <id> does not correspond to a valid Pool Member belonging to {org-id}. See **List Pool Members**.*

10. Network API - VIPs – Virtual Listener Management

Virtual Listener is the top level component of a VIP.

Virtual Listener Management is restricted to *Advanced* type Network Domains.

A Virtual Listener is used to expose its underlying Pool(s) to external network traffic via its *listenerIpAddress* property.

Further in-depth information is available on the Community:

- Introduction to VIPs in MCP 2.0
- How to Manage Virtual Listeners on a Network Domain.

10.1. Create Virtual Listener

10.1.1. Description

This function creates a Virtual Listener on a Network Domain in an MCP 2.0 data center location.

A Virtual Listener can listen on a public or private IPv4 address, refer to the **Request Properties** below for details.

A Virtual Listener can be created without any Pools. This is to facilitate creation of the various entities required for a complete and functioning Virtual Listener in whatever order suits the integrator.

However, for the Virtual Listener to **function,** it will need at least one Node and at least one Pool, with the Node attached to it as a Pool Member. See **Create Node**, **Create Pool** and **Add Pool Member**.

Addition of Persistence Profiles (stickiness) and iRules to the Virtual Listener is optional. If you want to include either, it is important to be aware that the available Persistence Profiles and iRules are governed by the combination of Virtual Listener *type* and *protocol.* In both cases, the respective API functions provide comprehensive compatibility information, defining which Persistence Profiles and which iRules can be use with each combination of *protocol* and *type*.

Refer to:

- **List Default Persistence Profiles**
- **List Default iRules**

As such it is important to note that *protocol* and *type* are immutable properties of a Virtual Listener and cannot be changed using the **Edit Virtual Listener** function.

An Optimization Profile (*optimizationProfile*) is mandatory for some combinations of *type* and *protocol*, however the combinations are sufficiently few for *optimizationProfile* to remain as an optional property – the inputs are documented in the **Request Properties** table below.

The properties of a Virtual Listener are described in detail in the Community article How to Create a Virtual Listener on a Network Domain.

10.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/createVirtualListener`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous.
Roles	"primary administrator" and "network"

XML Request Sample

```
<createVirtualListener xmlns="urn:didata.com:api:cloud:types">
    <networkDomainId>553f26b6-2a73-42c3-a78b-6116f11291d0</networkDomainId>
    <name>Production.Load.Balancer</name>
    <description>Used as the load balancer for the production
apps.</description>
    <type>STANDARD</type>
    <protocol>TCP</protocol>
```

```
    <listenerIpAddress>165.180.12.22</listenerIpAddress>
    <port>80</port>
    <enabled>true</enabled>
    <connectionLimit>25000</connectionLimit>
    <connectionRateLimit>2000</connectionRateLimit>
    <sourcePortPreservation>PRESERVE</sourcePortPreservation>
    <poolId>afb1fb1a-eab9-43f4-95c2-36a4cdda6cb8</poolId>
    <clientClonePoolId>033a97dc-ee9b-4808-97ea-
50b06624fd16</clientClonePoolId>
    <persistenceProfileId>a34ca25c-f3db-11e4-b010-
005056806999</persistenceProfileId>
    <fallbackPersistenceProfileId>033a97dc-ee9b-4808-97ea-
50b06624fd18</fallbackPersistenceProfileId>
    <optimizationProfile>TCP</optimizationProfile>
    <iruleId>2b20abd9-ffdc-11e4-b010-005056806999</iruleId>
</createVirtualListener>
```

JSON Request Sample

```
{
    "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
    "name": "Production.Load.Balancer",
    "description": "Used as the load balancer for the production
applications.",
    "type": "STANDARD",
    "protocol": "TCP",
    "listenerIpAddress": "165.180.12.22",
    "port": 80,
    "enabled": true,
    "connectionLimit": 25000,
    "connectionRateLimit": 2000,
    "sourcePortPreservation": "PRESERVE",
    "poolId": "afb1fb1a-eab9-43f4-95c2-36a4cdda6cb8",
    "clientClonePoolId": "033a97dc-ee9b-4808-97ea-50b06624fd16",
    "persistenceProfileId": "a34ca25c-f3db-11e4-b010-005056806999",
    "fallbackPersistenceProfileId": "6f2f5d7b-cdd9-4d84-8ad7-999b64a87978",
    "iruleId": [
      "2b20abd9-ffdc-11e4-b010-005056806999"
    ],
    "optimizationProfile": [
       "TCP"
     ]
}
```

Request Properties

Field	Required	Type and Constraints
networkDomainId	Yes	The *id* of a Network Domain belonging to {org-id}.
name	Yes	Must be alphanumeric with the following exceptions permitted "_" (underscore) and "." (full stop / period). Cannot begin with a "." or a number. Cannot contain spaces. Minimum length: 1 character. Maximum length: 75 characters.
description	No	Maximum length: 255 characters.
type	Yes	Must be one of: • STANDARD • PERFORMANCE_LAYER_4

protocol	Yes	The permitted range of values for *protocol* is governed by the choice of the *type* property. For *STANDARD type, protocol* must be one of: • ANY • TCP • UDP For *PERFORMANCE_LAYER_4 type, protocol* must be one of: • ANY • TCP • UDP • HTTP
listenerIpAddress	No*	Must be a valid IPv4 in dot-decimal notation (x.x.x.x). listenerIpAddress can be one of two types: 1. Public: listenerIpAddress is **optional* for creating a Public IP address Virtual Listener. If not supplied on a Create Virtual Listener request, the “Public” type below is assumed. The listenerIpAddress must be in the set of IP addresses comprised by the Public IPv4 Address Blocks on the Network Domain. If none is supplied, the next unused Public IPv4 Address will be selected automatically. See **List Public IPv4 Address Blocks**. 2. Private: listenerIpAddress is **required* to create a Private IP address Virtual Listener. The listenerIpAddress must be not already be in use by a Node on the Network Domain (see **List Nodes**) and cannot fall within the IP space of a VLAN deployed on the Network Domain (see **List VLANs**). For both types the listenerIpAddress cannot already be in use in a NAT Rule on the Network Domain. See **List NAT Rules**.
port	No	An integer in the range 1-65535. If *port* is not supplied it will be taken to mean "Any Port".
enabled	No	Must be one of *true* or *false*, where *true* indicates that the Virtual Listener will permit traffic to flow to its Pool and/or Client Clone Pool.
connectionLimit	Yes	An integer in the range 1 - MAX_VIRTUAL_LISTENER_CONNECTION_LIMIT. See **List Data Centers** for the MAX_VIRTUAL_LISTENER_CONNECTION_LIMIT property.
connectionRateLimit	Yes	An integer in the range 1 - MAX_VIRTUAL_LISTENER_CONNECTION_RATE_LIMIT. See **List Data Centers** for the MAX_VIRTUAL_LISTENER_CONNECTION_RATE_LIMIT property.
sourcePortPreservation	Yes	Must be one of: • PRESERVE • PRESERVE_STRICT • CHANGE

poolId	No	The *id* of a Pool on Network Domain, networkDomainId.
clientClonePoolId	No	The id of a Pool on Network Domain, networkDomainId. Cannot be the same Pool as identified by *poolId.*
persistenceProfileId	No	The *id* of a Persistence Profile compatible with the combination of *type* and *protocol* included. See **List Default Persistence Profiles**.
fallbackPersistenceProfileId	No	The *id* of a Persistence Profile compatible with the combination of *type* and *protocol* included. See **List Default Persistence Profiles**. Cannot be the same as *persistenceProfileId*. Cannot be included if a *persistenceProfileId* is not included.
optimizationProfile	No*	An optimizationProfile can only be included for certain *type* and *protocol* combinations: 1. *STANDARD/TCP: optimizationProfile is required for this combination and must be one of: TCP, LAN_OPT, WAN_OPT, MOBILE_OPT, TCP_LEGACY. 2. STANDARD/UDP: SMTP or SIP.
iruleId	No	A list of Rule *id* properties; each compatible with the combination of *type* and *protocol* included. See **List Default iRules**. Note that the order these are listed in the request payload, is the order in which they will be saved and determines the order of their execution.

10.1.3. Response Details

The response includes three "*info*" properties named:

- *virtualListenerId:* the value of this property contains the unique identifier of the new Virtual Listener and can be used in conjunction with **Get Virtual Listener** and/or **List Virtual Listeners** to retrieve details about the Virtual Listener.
- *name*: the value of the *name* property is informational and matches what was submitted in the request payload. This can be use for display and search purposes on **List Virtual Listeners.**
- *listenerIpAddress*: the value of this property matches the *listenerIpAddress* submitted in the request payload. This can be use for display and search purposes on **List Virtual Listeners.**

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-06-
11T05:45:22.833-04:00/0dc58e64-6e05-4de2-98d6-70bee28cba97">
    <operation>CREATE_VIRTUAL_LISTENER</operation>
    <responseCode>OK</responseCode>
    <message>Virtual Listener 'Production.Load.Balancer' has been created on
Public IP Address 165.180.12.22.</message>
    <info name="virtualListenerId" value="8334f461-0df0-42d5-97eb-
f4678eb26bea"/>
    <info name="name" value="Production.Load.Balancer"/>
    <info name="listenerIpAddress" value="165.180.12.22"/>
</response>
```

JSON

```
{
  "operation": "CREATE_VIRTUAL_LISTENER",
  "responseCode": "OK",
  "message": "Virtual Listener 'Production.Load.Balancer' has been created on
Public IP Address 165.180.12.22.",
```

```
  "info": [
    {
      "name": "virtualListenerId",
      "value": "43a445f1-9ac9-4f13-8b0d-a2d1fad231c3"
    },
    {
      "name": "name",
      "value": "Production.Load.Balancer"
    },
    {
      "name": "listenerIpAddress",
      "value": "165.180.12.22"
    }
  ],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-06-11T05:39:21.529-04:00/cbd08e3e-2151-4a7b-8569-
d4be191200e7"
}
```


10.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	protocol <protocol> may not be used with type <type>. A combination of a specific Port (<port>) with "Any" as the Protocol option can only be used with the Performance Layer 4 Virtual Listener Type. Persistence Profile '<persistence profile name>' (id: <persistenceProfileId) cannot be used as a Fallback Persistence Profile. See **List Default Persistence Profiles**. persistenceProfile <persistenceProfileId> may not be used with the combination of type <type> and protocol <protocol>. See **List Default Persistence Profiles**. iRule <iruleId> cannot be used with the combination of type <type> and protocol <protocol>. See **List Default iRules**. optimizationProfile is mandatory for the combination of type STANDARD and protocol TCP. Must be one of TCP, LAN_OPT, WAN_OPT, MOBILE_OPT or TCP_LEGACY. optimizationProfile is optional for the combination of type STANDARD and protocol UDP. Must be one of SMTP or SIP if supplied.
EXCEEDS_LIMIT	A Network Domain may only have a maximum of <max> Virtual Listeners.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INCOMPATIBLE_OPERATION	Operation not available for ESSENTIALS Network Domains.
INVALID_INPUT_DATA	<fieldName> must be provided. Invalid name: name must not include spaces. Invalid name: name can only include alphanumeric characters, '_' and '.'. Invalid name: name must not start with a number or '.'.

	Invalid name: exceeds maximum length of 75 characters. Invalid name: name must be provided and must be at least 1 character in length. Invalid description: exceeds maximum length of 255 characters. Invalid IP address <listenerIpAddress>. Must be a valid IPv4 in dot-decimal notation (x.x.x.x). port <port> is outside the permitted range (1 to 65535). connectionLimit <connectionLimit> is outside the permitted range. See **Request Properties** for range details. connectionRateLimit <connectionRateLimit> is outside the permitted range. See **Request Properties** for range details. type must be one of STANDARD, PERFORMANCE_LAYER_4. protocol must be one of ANY, TCP, UDP, HTTP. fallbackPersistenceProfileId may not be provided unless a persistenceProfileId is provided. fallbackPersistenceProfileId and persistenceProfileId cannot be the same. clientClonePoolId and poolId may not be the same. iruleId <iruleId> should not be provided more than once. Only one optimizationProfile may be provided.
IP_ADDRESS_NOT_UNIQUE	IP address <listenerIpAddress> on ANY port is already in use by Virtual Listener <virtualListenerId>. IP address <listenerIpAddress> on port <port> is already in use by Virtual Listener <virtualListenerId>. IP address <listenerIpAddress> is already is use by NAT <natId>. IP address <listenerIpAddress> is already in use by Node <nodeId>. IP address <listenerIpAddress> is already within the address space of VLAN <vlanName>, <vlanIp>/<vlanPrefix> (<vlanId>).
IP_ADDRESS_OUT_OF_RANGE	listenerIpAddress <listenerIpAddress> does not exist within any of the Public IPv4 Blocks currently assigned to Network Domain with Id <networkDomainId>.
NAME_NOT_UNIQUE	Another Virtual Listener named <name> already exists in Network Domain <networkDomainId>.
NO_IP_ADDRESS_AVAILABLE	None of the Public IPv4 Blocks currently assigned to Network Domain with Id <id> have any IP addresses available for use as an IPv4 Address.
RESOURCE_BUSY	Another operation is in progress on Network Domain with id <networkDomainId>. Please try again later. Another operation is in progress on Pool with id <poolId>. Please try again later. Another operation is in progress on IP Block with id

	<publicIpv4AddressBlockId>. Please try again later. *This corresponds to the Public IPv4 Address Block identified by the listenerIpAddress property. See* ***Get Public IPv4 Address Block***.
RESOURCE_LOCKED	Network Domain with id <networkDomainId> is locked. Please contact support. Pool with id <poolId> is locked. Please contact support. IP Block with id <publicIpv4AddressBlockId> is locked. Please contact support. *This corresponds to the Public IPv4 Address Block identified by the listenerIpAddress property. See* ***Get Public IPv4 Address Block***.
RESOURCE_NOT_FOUND	Network Domain <networkDomainId> not found. *See* ***List Network Domains***. Pool <poolId> not found. See **List Pools.** Persistence Profile <persitenceProfileId> not found. See **List Default Persistence Profiles**. irule <iruleId> not found. See **List Default iRules**.

10.2. List Virtual Listener

10.2.1. Description

Lists all of the Virtual Listeners at an MCP 2.0 data center, belonging to the organization identified by the {org-id} parameter. Various filter values can be used to narrow the list returned as detailed below.

Note that it is more efficient to use **Get Virtual Listener** than **List Virtual Listeners** (filtered by *id*) if you wish to retrieve the details of a single Virtual Listener.

10.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/virtualListener`
	
	`Paging/Ordering Optional Parameters:`
	`[&pageSize=]`
	`[&pageNumber=]`
	`[&orderBy=]`
	
	`Filter Optional Parameters:`
	`[&id=]`
	`[&networkDomainId=]`
	`[&datacenterId=]`
	`[&name=]`
	`[&enabled=]`
	`[&state=]`
	`[&createTime=]`
	`[&type=]`
	`[&protocol=]`
	`[&listenerIpAddress=]`
	`[&port=]`
	`[&poolId=]`
	`[&clientClonePoolId=]`
	`[&persistenceProfileId=]`
	`[&fallbackPersistenceProfileId=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual Virtual Listener. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
networkDomainId	uuid (String)	Identifies an individual Network Domain. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes

name	String	Identifies Virtual Listener by their *name*. *name=Production.VirtualListener.1* Supports the use of the LIKE comparator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	Yes
enabled	Boolean	Identifies Virtual Listeners by whether or not they are *enabled*. *enabled=false*	Yes	No
state	String	Identifies Virtual Listeners by their *state*. Case insensitive. The initial possible set of values for *virtualListener.state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.	Yes	No
createTime	Date	Identifies the date of creation of Virtual Listener. Supports MIN, MAX, LT and GT. *createTime.MIN=2015-05-31T00:00:00Z* Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes
type	String	Filters by Virtual Listener *type*. *type=STANDARD* See **Create Virtual Listener** for the full set of values.	Yes	No
protocol	String	Filters by Virtual Listener *protocol*. *protocol=UDP* See **Create Virtual Listener** for the full set of values.	Yes	No
listenerIpAddress	String	Filters by the listenerIpAddress. *listenerIpAddress=165.101.11.115*	Yes	No
port	Integer	Filters by the Virtual Listener *port*. Supports MIN, MAX, LT and GT.	Yes	No

		port=9888 Refer to samples in **Paging and Filtering for List API Functions.**		
poolId	uuid (String)	Filters by the Pool *id* property in the context of the Virtual Listener *poolId* property. See **List Pools**. *poolId=9a857b7c-37bd-11e2-a91c-0030487e0302.*	Yes	No
clientClonePoolId	uuid (String)	Filters by the Pool *id* property in the context of the Virtual Listener *clientClonePoolId* property. See **List Pools**. *clientClonePoolId=9a857b7c-37bd-11e2-a91c-0030487e0302.*	Yes	No
persistenceProfileId	uuid (String)	Filters by the Default Persistence Profile *id* property in the context of the Virtual Listener *persistenceProfileId* property. See **List Default Persistence Profiles**. *persistenceProfileId=9a857b7c-37bd-11e2-a91c-0030487e0302.*	Yes	No
fallbackPersistenceProfileId	uuid (String)	Filters by the Default Persistence Profile *id* property in the context of the Virtual Listener *fallbackPersistenceProfileId* property. See **List Default Persistence Profiles**. *persistenceProfileId=9a857b7c-37bd-11e2-a91c-0030487e0302.*	Yes	No

10.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<virtualListeners
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="5"
totalCount="5" pageSize="250">
    <virtualListener id="6115469d-a8bb-445b-bb23-d23b5283f2b9"
datacenterId="NA9">
        <networkDomainId>553f26b6-2a73-42c3-a78b-
6116f11291d0</networkDomainId>
        <name>myProduction.Virtual.Listener</name>
        <state>NORMAL</state>
        <description>Virtual Listener for load balancing our test
systems.</description>
        <createTime>2015-05-28T15:59:49.000Z</createTime>
        <type>PERFORMANCE_LAYER_4</type>
        <protocol>HTTP</protocol>
        <listenerIpAddress>165.180.12.22</listenerIpAddress>
        <port>8899</port>
        <enabled>true</enabled>
        <connectionLimit>10000</connectionLimit>
        <connectionRateLimit>400</connectionRateLimit>
        <sourcePortPreservation>PRESERVE</sourcePortPreservation>
        <pool id="afb1fb1a-eab9-43f4-95c2-36a4cdda6cb8"
name="myProductionPool.1">
```

```
            <loadBalanceMethod>ROUND_ROBIN</loadBalanceMethod>
            <serviceDownAction>NONE</serviceDownAction>
            <slowRampTime>10</slowRampTime>
            <healthMonitor id="01683574-d487-11e4-811f-005056806999"
name="CCDEFAULT.Http"/>
            <healthMonitor id="0168546c-d487-11e4-811f-005056806999"
name="CCDEFAULT.Https"/>
        </pool>
        <clientClonePool id="6f2f5d7b-cdd9-4d84-8ad7-999b64a87978"
name="myDevelopmentPool.1">
            <loadBalanceMethod>ROUND_ROBIN</loadBalanceMethod>
            <serviceDownAction>RESELECT</serviceDownAction>
            <slowRampTime>10</slowRampTime>
            <healthMonitor id="01683574-d487-11e4-811f-005056806999"
name="CCDEFAULT.Http"/>
            <healthMonitor id="0168546c-d487-11e4-811f-005056806999"
name="CCDEFAULT.Https"/>
        </clientClonePool>
        <persistenceProfile id="a34ca25c-f3db-11e4-b010-005056806999"
name="CCDEFAULT.DestinationAddress"/>
        <fallbackPersistenceProfile id="a34ca3f6-f3db-11e4-b010-005056806999"
name="CCDEFAULT.SourceAddress"/>
        <irule id="2b20abd9-ffdc-11e4-b010-005056806999"
name="CCDEFAULT.IpProtocolTimers"/>
        <irule id="2b20e790-ffdc-11e4-b010-005056806999"
name="CCDEFAULT.Ips"/>
    </virtualListener>
</virtualListeners>
```

JSON

```
{
  "virtualListener": [
    {
      "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
      "name": "myProduction.Virtual.Listener",
      "state": "NORMAL",
      "description": "Virtual Listener for load balancing our test systems.",
      "createTime": "2015-05-28T15:59:49.000Z",
      "type": "PERFORMANCE_LAYER_4",
      "protocol": "HTTP",
      "listenerIpAddress": "165.180.12.22",
      "port": 8899,
      "enabled": true,
      "connectionLimit": 10000,
      "connectionRateLimit": 400,
      "sourcePortPreservation": "PRESERVE",
      "pool": {
        "loadBalanceMethod": "ROUND_ROBIN",
        "serviceDownAction": "NONE",
        "slowRampTime": 10,
        "healthMonitor": [
          {
            "id": "01683574-d487-11e4-811f-005056806999",
            "name": "CCDEFAULT.Http"
          },
          {
            "id": "0168546c-d487-11e4-811f-005056806999",
            "name": "CCDEFAULT.Https"
          }
        ],
        "id": "afb1fb1a-eab9-43f4-95c2-36a4cdda6cb8",
        "name": "myProductionPool.1"
      },
      "clientClonePool": {
        "loadBalanceMethod": "ROUND_ROBIN",
        "serviceDownAction": "RESELECT",
        "slowRampTime": 10,
```

```
        "healthMonitor": [
          {
            "id": "01683574-d487-11e4-811f-005056806999",
            "name": "CCDEFAULT.Http"
          },
          {
            "id": "0168546c-d487-11e4-811f-005056806999",
            "name": "CCDEFAULT.Https"
          }
        ],
        "id": "6f2f5d7b-cdd9-4d84-8ad7-999b64a87978",
        "name": "myDevelopmentPool.1"
      },
      "persistenceProfile": {
        "id": "a34ca25c-f3db-11e4-b010-005056806999",
        "name": "CCDEFAULT.DestinationAddress"
      },
      "fallbackPersistenceProfile": {
        "id": "a34ca3f6-f3db-11e4-b010-005056806999",
        "name": "CCDEFAULT.SourceAddress"
      },
      "optimizationProfile": [],
      "id": "6115469d-a8bb-445b-bb23-d23b5283f2b9",
      "datacenterId": "NA9",
      "irule": [
        {
          "id": "2b20abd9-ffdc-11e4-b010-005056806999",
          "name": "CCDEFAULT.IpProtocolTimers"
        },
        {
          "id": "2b20e790-ffdc-11e4-b010-005056806999",
          "name": "CCDEFAULT.Ips"
        }
      ]
    }
  ],
  "pageNumber": 1,
  "pageCount": 1,
  "totalCount": 1,
  "pageSize": 250
}
```


10.2.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

10.3. Get Virtual Listener

10.3.1. Description

Returns details of a single Virtual Listener belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get Virtual Listener** than **List Virtual Listeners** (filtered by *id*) if you wish to retrieve *state* status or details of a single Virtual Listener.

10.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/virtualListener/{virtual-listener-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

10.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<virtualListener
    xmlns="urn:didata.com:api:cloud:types" id="6115469d-a8bb-445b-bb23-
d23b5283f2b9" datacenterId="NA9">
    <networkDomainId>553f26b6-2a73-42c3-a78b-6116f11291d0</networkDomainId>
    <name>myProduction.Virtual.Listener</name>
    <state>NORMAL</state>
    <description>Virtual Listener for load balancing our test
systems.</description>
    <createTime>2015-05-28T15:59:49.000Z</createTime>
    <type>PERFORMANCE_LAYER_4</type>
    <protocol>HTTP</protocol>
    <listenerIpAddress>165.180.12.22</listenerIpAddress>
    <port>8899</port>
    <enabled>true</enabled>
    <connectionLimit>10000</connectionLimit>
    <connectionRateLimit>400</connectionRateLimit>
    <sourcePortPreservation>PRESERVE</sourcePortPreservation>
    <pool id="afb1fb1a-eab9-43f4-95c2-36a4cdda6cb8" name="myProductionPool.1">
        <loadBalanceMethod>ROUND_ROBIN</loadBalanceMethod>
        <serviceDownAction>NONE</serviceDownAction>
        <slowRampTime>10</slowRampTime>
        <healthMonitor id="01683574-d487-11e4-811f-005056806999"
name="CCDEFAULT.Http"/>
        <healthMonitor id="0168546c-d487-11e4-811f-005056806999"
name="CCDEFAULT.Https"/>
    </pool>
    <clientClonePool id="6f2f5d7b-cdd9-4d84-8ad7-999b64a87978"
name="myDevelopmentPool.1">
        <loadBalanceMethod>ROUND_ROBIN</loadBalanceMethod>
        <serviceDownAction>RESELECT</serviceDownAction>
        <slowRampTime>10</slowRampTime>
        <healthMonitor id="01683574-d487-11e4-811f-005056806999"
name="CCDEFAULT.Http"/>
        <healthMonitor id="0168546c-d487-11e4-811f-005056806999"
name="CCDEFAULT.Https"/>
    </clientClonePool>
    <persistenceProfile id="a34ca25c-f3db-11e4-b010-005056806999"
name="CCDEFAULT.DestinationAddress"/>
    <fallbackPersistenceProfile id="a34ca3f6-f3db-11e4-b010-005056806999"
name="CCDEFAULT.SourceAddress"/>
    <irule id="2b20abd9-ffdc-11e4-b010-005056806999"
name="CCDEFAULT.IpProtocolTimers"/>
    <irule id="2b20e790-ffdc-11e4-b010-005056806999" name="CCDEFAULT.Ips"/>
```

```
</virtualListener>
```

JSON

```
{
  "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
  "name": "myProduction.Virtual.Listener",
  "state": "NORMAL",
  "description": "Virtual Listener for load balancing our test systems.",
  "createTime": "2015-05-28T15:59:49.000Z",
  "type": "PERFORMANCE_LAYER_4",
  "protocol": "HTTP",
  "listenerIpAddress": "165.180.12.22",
  "port": 8899,
  "enabled": true,
  "connectionLimit": 10000,
  "connectionRateLimit": 400,
  "sourcePortPreservation": "PRESERVE",
  "pool": {
    "loadBalanceMethod": "ROUND_ROBIN",
    "serviceDownAction": "NONE",
    "slowRampTime": 10,
    "healthMonitor": [
      {
        "id": "01683574-d487-11e4-811f-005056806999",
        "name": "CCDEFAULT.Http"
      },
      {
        "id": "0168546c-d487-11e4-811f-005056806999",
        "name": "CCDEFAULT.Https"
      }
    ],
    "id": "afb1fb1a-eab9-43f4-95c2-36a4cdda6cb8",
    "name": "myProductionPool.1"
  },
  "clientClonePool": {
    "loadBalanceMethod": "ROUND_ROBIN",
    "serviceDownAction": "RESELECT",
    "slowRampTime": 10,
    "healthMonitor": [
      {
        "id": "01683574-d487-11e4-811f-005056806999",
        "name": "CCDEFAULT.Http"
      },
      {
        "id": "0168546c-d487-11e4-811f-005056806999",
        "name": "CCDEFAULT.Https"
      }
    ],
    "id": "6f2f5d7b-cdd9-4d84-8ad7-999b64a87978",
    "name": "myDevelopmentPool.1"
  },
  "persistenceProfile": {
    "id": "a34ca25c-f3db-11e4-b010-005056806999",
    "name": "CCDEFAULT.DestinationAddress"
  },
  "fallbackPersistenceProfile": {
    "id": "a34ca3f6-f3db-11e4-b010-005056806999",
    "name": "CCDEFAULT.SourceAddress"
  },
  "optimizationProfile": [],
  "id": "6115469d-a8bb-445b-bb23-d23b5283f2b9",
  "datacenterId": "NA9",
  "irule": [
    {
      "id": "2b20abd9-ffdc-11e4-b010-005056806999",
      "name": "CCDEFAULT.IpProtocolTimers"
    },
```

```
    {
      "id": "2b20e790-ffdc-11e4-b010-005056806999",
      "name": "CCDEFAULT.Ips"
    }
  ]
}
```

10.3.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Node {virtual-listener-id} not found. *The value passed for {virtual-listener-id} does not correspond to a valid Virtual Listener belonging to {org-id}. See **List Virtual Listeners**.*

10.4. Edit Virtual Listener

10.4.1. Description

Updates the mutable properties of a Virtual Listener belonging to the organization identified by {org-id}.

10.4.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/editVirtualListener`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

See **Considerations for Edit API Functions** for further information on updates to optional fields.

XML Request Sample

```
<editVirtualListener id="6e42868b-e013-41c3-ac38-5f7b50d54808"
xmlns="urn:didata.com:api:cloud:types">
    <description>Used as the load balancer for the production
apps.</description>
    <enabled>true</enabled>
    <connectionLimit>25000</connectionLimit>
    <connectionRateLimit>2000</connectionRateLimit>
    <sourcePortPreservation>PRESERVE</sourcePortPreservation>
    <poolId>afb1fb1a-eab9-43f4-95c2-36a4cdda6cb8</poolId>
    <persistenceProfileId>a34ca25c-f3db-11e4-b010-
005056806999</persistenceProfileId>
    <optimizationProfile>TCP</optimizationProfile>
    <iruleId>2b20abd9-ffdc-11e4-b010-005056806999</iruleId>
</editVirtualListener>
```

JSON Request Sample

```
{
    "id": "6e42868b-e013-41c3-ac38-5f7b50d54808",
    "description": "Used as the load balancer for the production
applications.",
    "enabled": true,
    "connectionLimit": 25000,
    "connectionRateLimit": 2000,
    "sourcePortPreservation": "PRESERVE",
    "poolId": "afb1fb1a-eab9-43f4-95c2-36a4cdda6cb8",
    "persistenceProfileId": "a34ca25c-f3db-11e4-b010-005056806999",
    "iruleId": [
      "2b20abd9-ffdc-11e4-b010-005056806999"
    ],
    "optimizationProfile": [
       "TCP"
     ]
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Virtual Listener being edited.
description	No	Maximum length: 255 characters.
enabled	No	Must be one of *true* or *false*, where *true* indicates that the Virtual Listener will permit traffic to flow to its Pool and/or Client Clone Pool.

connectionLimit	No	An integer in the range 1 - MAX_VIRTUAL_LISTENER_CONNECTION_LIMIT. See **List Data Centers** for the MAX_VIRTUAL_LISTENER_CONNECTION_LIMIT property.
connectionRateLimit	No	An integer in the range 1 - MAX_VIRTUAL_LISTENER_CONNECTION_RATE_LIMIT. See **List Data Centers** for the MAX_VIRTUAL_LISTENER_CONNECTION_RATE_LIMIT property.
sourcePortPreservation	No	Must be one of: • PRESERVE • PRESERVE_STRICT • CHANGE
poolId	No	The *id* of a Pool on the same Network Domain. See **Get Virtual Listener** for *networkDomainId*.
clientClonePoolId	No	The id of a Pool on the same Network Domain. Cannot be the same as *poolId*. See **Get Virtual Listener** for *networkDomainId*.
persistenceProfileId	No	The *id* of a Persistence Profile compatible with the combination of *type* and *protocol* included. See **List Default Persistence Profiles**.
fallbackPersistenceProfileId	No	The *id* of a Persistence Profile compatible with the combination of *type* and *protocol* included. See **List Default Persistence Profiles**. Cannot be the same as *persistenceProfileId*. Cannot be included if a *persistenceProfileId* is not included.
optimizationProfile	No	An optimizationProfile can only be included for certain *type* and *protocol* combinations: 3. STANDARD/TCP: One of: TCP, LAN_OPT, WAN_OPT, MOBILE_OPT, TCP_LEGACY. 4. STANDARD/UDP: One of: SMTP or SIP.
iruleId	No	A list of iRule *id* properties; each compatible with the combination of *type* and *protocol* included. See **List Default iRules**. Note that the order these are listed in the request payload, is the order in which they will be saved and determines the order of their execution. Refer to **Considerations for Edit API Functions** for further information on editing such lists.

10.4.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-06-
12T04:41:35.010-04:00/ae9f8391-457c-494c-b058-99483b5fe7d0">
    <operation>EDIT_VIRTUAL_LISTENER</operation>
    <responseCode>OK</responseCode>
    <message>Virtual Listener 'Production.Load.Balancer' has been
updated.</message>
</response>
```

JSON

```
{
  "operation": "EDIT_VIRTUAL_LISTENER",
  "responseCode": "OK",
  "message": "Virtual Listener 'Production.Load.Balancer' has been updated.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-06-12T04:41:04.469-04:00/3730165e-b9c6-427d-a84c-
86e6a55b1eb1"
}
```

10.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	persistenceProfile <persistenceProfileId> may not be used with the combination of type <type> and protocol <protocol>. See **List Default Persistence Profiles**. Persistence Profile '<persistence profile name>' (id: <persistenceProfileId) cannot be used as a Fallback Persistence Profile. See **List Default Persistence Profiles**. iRule <iruleId> cannot be used with the combination of type <type> and protocol <protocol>. See **List Default iRules**. optimizationProfile is mandatory for the combination of type STANDARD and protocol TCP. Must be one of TCP, LAN_OPT, WAN_OPT, MOBILE_OPT or TCP_LEGACY. optimizationProfile is optional for the combination of type STANDARD and protocol UDP. Must be one of SMTP or SIP if supplied.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INCOMPATIBLE_OPERATION	Operation not available for ESSENTIALS Network Domains.
INVALID_INPUT_DATA	<id> must be provided. At least one of description, enabled, connectionLimit, connectionRateLimit, sourcePortPreservation, poolId, clientClonePoolId, persistenceProfileId, fallbackPersistenceProfileId, optimizationProfile or iruleId must be supplied. Invalid description: exceeds maximum length of 255 characters. connectionLimit <connectionLimit> is outside the permitted range. See **Request Properties** for range details. connectionRateLimit <connectionRateLimit> is outside the permitted range. See **Request Properties** for range details. fallbackPersistenceProfileId may not be provided unless a persistenceProfileId is provided or is already present. fallbackPersistenceProfileId and persistenceProfileId cannot be the same. clientClonePoolId and poolId may not be the same.

	iruleId <iruleId> should not be provided more than once. Only one optimizationProfile may be provided.
RESOURCE_BUSY	Another operation is in progress on Virtual Listener with id <id>. Please try again later. See **List Virtual Listeners**. Another operation is in progress on Pool with id <poolId>. Please try again later. See **List Pools**. Another operation is in progress on Network Domain with id <networkDomainId>. Please try again later. See **List Network Domains**.
RESOURCE_LOCKED	Virtual Listener with id <id> is locked. Please contact support. Pool with id <poolId> is locked. Please contact support. Network Domain with id <networkDomainId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Virtual Listener <id> not found. *See **List Virtual Listeners**.* Pool <poolId> not found. See **List Pools.** Persistence Profile <persitenceProfileId> not found. See **List Default Persistence Profiles**. iRule <iruleId> not found. See **List Default iRules**.

10.5. Delete Virtual Listener

10.5.1. Description

Deletes a Virtual Listener belonging to the organization identified by {org-id}.

Note that if there is a Pool or Client Clone Pool associated with a Virtual Listener, they are **not** deleted if the Virtual Listener is deleted. They remain unaffected.

10.5.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/deleteVirtualListener`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous.
Roles	"primary administrator" and "network"

XML Request Sample

```
<deleteVirtualListener id="6115469d-a8bb-445b-bb23-d23b5283f2b9"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "6115469d-a8bb-445b-bb23-d23b5283f2b9"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Virtual Listener to be deleted.

10.5.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response requestId="NA9/2015-05-14T13:37:20/62f06368-c3fb-11e3-b29c-
001517c4643e"
 xmlns="urn:didata.com:api:cloud:types">
  <operation>DELETE_VIRTUAL_LISTENER</operation>
  <responseCode>OK</responseCode>
  <message>Virtual Listener (id:6115469d-a8bb-445b-bb23-d23b5283f2b9) has been
deleted.</message>
</response>
```

JSON

```
{
  "operation": "DELETE_VIRTUAL_LISTENER",
  "responseCode": "OK",
  "message": "Virtual Listener (Id:6115469d-a8bb-445b-bb23-d23b5283f2b9) has
been deleted",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-05-29T12:10:56.433-04:00/cfbb1c31-31ac-4f7c-a635-
1d7419668e1a"
}
```

10.5.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	Virtual Listener id is required.
RESOURCE_BUSY	Another operation is in progress on Virtual Listener with Id <id>. Please try again later.
RESOURCE_LOCKED	Virtual Listener with id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Virtual Listener <id> not found. *The value passed for <id> does not correspond to a valid Virtual Listener belonging to {org-id}. See **List Virtual Listeners**.*

11. Network API - VIPs – Supporting Functions

VIPs Supporting Functions provide inputs to a number of VIPs "Create" functions and filters on VIPs "List" functions. Each such API function definition declares which *Supporting Function(s)* is relevant to it.

For example: **Create Node** declares that **List Default Health Monitors** (nodeCompatible) is of relevance for its optional *healthMonitorId* property.

11.1. List Default Health Monitors

11.1.1. Description

This function is of relevance to **Create Node**, **Edit Node**, **Create Pool** and **Edit Pool**.

Lists the default Health Monitors at an MCP 2.0 data center specific to a Network Domain (networkDomainId) belonging to the organization identified by the {org-id} parameter.

Other filter values can be used to narrow the list returned as detailed below.

Each Health Monitor declares whether or not it is "*nodeCompatible*" and/or "*poolCompatible*".

11.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/defaultHealthMonitor?`
	
	`Filter Required Parameters:`
	`  networkDomainId=`
	
	`Filter Optional Parameters:`
	`  [&id=]`
	`  [&datacenterId=]`
	`  [&name=]`
	`  [&enabled=]`
	`  [&state=]`
	
	`Paging/Ordering Optional Parameters:`
	`  [&pageSize=]`
	`  [&pageNumber=]`
	`  [&orderBy=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
networkDomainId	uuid (String)	**Required**. Identifies the Network Domain, which the Health Monitors correspond to. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
id	uuid (String)	Identifies an individual Health Monitor. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No

name	String	Identifies Health Monitors by their *name*. *name=CCDEFAULT.Https* Supports the use of the LIKE comparator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	Yes
nodeCompatible	Boolean	Filters by Health Monitors which can be specified on **Create Node** or **Edit Node**. *nodeCompatible=true*	Yes	Yes
poolCompatible	Boolean	Filters by Health Monitors which can be specified on **Create Pool** or **Edit Pool**. *poolCompatible=true*	Yes	Yes

11.1.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<defaultHealthMonitors
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="6"
totalCount="6" pageSize="250">
    <defaultHealthMonitor id="01683574-d487-11e4-811f-005056806999">
        <name>CCDEFAULT.Http</name>
        <nodeCompatible>false</nodeCompatible>
        <poolCompatible>true</poolCompatible>
    </defaultHealthMonitor>
    <defaultHealthMonitor id="0168546c-d487-11e4-811f-005056806999">
        <name>CCDEFAULT.Https</name>
        <nodeCompatible>false</nodeCompatible>
        <poolCompatible>true</poolCompatible>
    </defaultHealthMonitor>
    <defaultHealthMonitor id="0168b83a-d487-11e4-811f-005056806999">
        <name>CCDEFAULT.Icmp</name>
        <nodeCompatible>true</nodeCompatible>
        <poolCompatible>false</poolCompatible>
    </defaultHealthMonitor>
    <defaultHealthMonitor id="01686f4b-d487-11e4-811f-005056806999">
        <name>CCDEFAULT.Tcp</name>
        <nodeCompatible>false</nodeCompatible>
        <poolCompatible>true</poolCompatible>
    </defaultHealthMonitor>
    <defaultHealthMonitor id="0168a2f9-d487-11e4-811f-005056806999">
        <name>CCDEFAULT.TcpHalfOpen</name>
        <nodeCompatible>false</nodeCompatible>
        <poolCompatible>true</poolCompatible>
    </defaultHealthMonitor>
    <defaultHealthMonitor id="01688878-d487-11e4-811f-005056806999">
        <name>CCDEFAULT.Udp</name>
        <nodeCompatible>false</nodeCompatible>
        <poolCompatible>true</poolCompatible>
    </defaultHealthMonitor>
</defaultHealthMonitors>
```

JSON

```
{
  "defaultHealthMonitor": [
    {
```

```
      "name": "CCDEFAULT.Http",
      "nodeCompatible": false,
      "poolCompatible": true,
      "id": "01683574-d487-11e4-811f-005056806999"
    },
    {
      "name": "CCDEFAULT.Https",
      "nodeCompatible": false,
      "poolCompatible": true,
      "id": "0168546c-d487-11e4-811f-005056806999"
    },
    {
      "name": "CCDEFAULT.Icmp",
      "nodeCompatible": true,
      "poolCompatible": false,
      "id": "0168b83a-d487-11e4-811f-005056806999"
    },
    {
      "name": "CCDEFAULT.Tcp",
      "nodeCompatible": false,
      "poolCompatible": true,
      "id": "01686f4b-d487-11e4-811f-005056806999"
    },
    {
      "name": "CCDEFAULT.TcpHalfOpen",
      "nodeCompatible": false,
      "poolCompatible": true,
      "id": "0168a2f9-d487-11e4-811f-005056806999"
    },
    {
      "name": "CCDEFAULT.Udp",
      "nodeCompatible": false,
      "poolCompatible": true,
      "id": "01688878-d487-11e4-811f-005056806999"
    }
  ],
  "pageNumber": 1,
  "pageCount": 6,
  "totalCount": 6,
  "pageSize": 250
}
```


11.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	A networkDomainId filter must be provided.

11.2. List Default Persistence Profiles

11.2.1. Description

This API function is of relevance to **Create Virtual Listener** and **Edit Virtual Listener**.

Lists the default Persistence Profiles at an MCP 2.0 data center specific to a Network Domain (networkDomainId) belonging to the organization identified by the {org-id} parameter.

Other filter values can be used to narrow the list returned as detailed below.

Each Persistence Profile declares the combination of Virtual Listener *type* and *protocol* with which it is compatible and whether or not it is compatible as a Fallback Persistence Profile.

11.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/defaultPersistenceProfile?`
	
	`Filter Required Parameters:`
	`  networkDomainId=`
	
	`Filter Optional Parameters:`
	`  [&id=]`
	`  [&datacenterId=]`
	`  [&name=]`
	`  [&enabled=]`
	`  [&state=]`
	
	`Paging/Ordering Optional Parameters:`
	`  [&pageSize=]`
	`  [&pageNumber=]`
	`  [&orderBy=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
networkDomainId	uuid (String)	**Required**. Identifies the Network Domain, which the Persistence Profile correspond to. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
id	uuid (String)	Identifies an individual Persistence Profile. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
name	String	Identifies Persistence Profiles by their *name*. *name=CCDEFAULT.Cookie* Supports the use of the LIKE comparator defined in the ***Paging and Filtering for List API***	Yes	No

		Functions overview.		
fallbackCompatible	Boolean	Filters by Persistence Profiles which can be specified as Fallback Persistence Profiles on **Create Virtual Listener** and **Edit Virtual Listener** calls. *fallbackCompatible=true*	Yes	No
virtualListenerType	String	Filters by Virtual Listener compatibility type. See **Create Virtual Listener** for the set of valid values for Virtual Listener type. *virtualListenerType=STANDARD*	Yes	No
virtualListenerProtocol	String	Filters by Virtual Listener compatibility protocol. See **Create Virtual Listener** for the set of valid values for Virtual Listener protocol. *virtualListenerProtocol=TCP*	Yes	No

11.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<defaultPersistenceProfiles
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="4"
totalCount="4" pageSize="250">
    <defaultPersistenceProfile id="a34ca024-f3db-11e4-b010-005056806999"
fallbackCompatible="false">
        <name>CCDEFAULT.Cookie</name>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="HTTP"/>
    </defaultPersistenceProfile>
    <defaultPersistenceProfile id="a34ca25c-f3db-11e4-b010-005056806999"
fallbackCompatible="true">
        <name>CCDEFAULT.DestinationAddress</name>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="HTTP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="TCP"/>
        <virtualListenerCompatibility type="STANDARD" protocol="UDP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="UDP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="ANY"/>
        <virtualListenerCompatibility type="STANDARD" protocol="TCP"/>
        <virtualListenerCompatibility type="STANDARD" protocol="ANY"/>
    </defaultPersistenceProfile>
    <defaultPersistenceProfile id="a34ca4b7-f3db-11e4-b010-005056806999"
fallbackCompatible="false">
        <name>CCDEFAULT.Sip</name>
        <virtualListenerCompatibility type="STANDARD" protocol="UDP"/>
        <virtualListenerCompatibility type="STANDARD" protocol="TCP"/>
    </defaultPersistenceProfile>
    <defaultPersistenceProfile id="a34ca3f6-f3db-11e4-b010-005056806999"
fallbackCompatible="true">
        <name>CCDEFAULT.SourceAddress</name>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="HTTP"/>
```

```
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="TCP"/>
        <virtualListenerCompatibility type="STANDARD" protocol="UDP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="UDP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="ANY"/>
        <virtualListenerCompatibility type="STANDARD" protocol="TCP"/>
        <virtualListenerCompatibility type="STANDARD" protocol="ANY"/>
    </defaultPersistenceProfile>
</defaultPersistenceProfiles>
```

JSON

```
{
  "defaultPersistenceProfile": [
    {
      "name": "CCDEFAULT.Cookie",
      "virtualListenerCompatibility": [
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "HTTP"
        }
      ],
      "id": "a34ca024-f3db-11e4-b010-005056806999",
      "fallbackCompatible": false
    },
    {
      "name": "CCDEFAULT.DestinationAddress",
      "virtualListenerCompatibility": [
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "HTTP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "TCP"
        },
        {
          "type": "STANDARD",
          "protocol": "UDP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "UDP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "ANY"
        },
        {
          "type": "STANDARD",
          "protocol": "TCP"
        },
        {
          "type": "STANDARD",
          "protocol": "ANY"
        }
      ],
      "id": "a34ca25c-f3db-11e4-b010-005056806999",
      "fallbackCompatible": true
    },
    {
      "name": "CCDEFAULT.Sip",
      "virtualListenerCompatibility": [
        {
          "type": "STANDARD",
          "protocol": "UDP"
```

```
        },
        {
          "type": "STANDARD",
          "protocol": "TCP"
        }
      ],
      "id": "a34ca4b7-f3db-11e4-b010-005056806999",
      "fallbackCompatible": false
    },
    {
      "name": "CCDEFAULT.SourceAddress",
      "virtualListenerCompatibility": [
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "HTTP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "TCP"
        },
        {
          "type": "STANDARD",
          "protocol": "UDP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "UDP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "ANY"
        },
        {
          "type": "STANDARD",
          "protocol": "TCP"
        },
        {
          "type": "STANDARD",
          "protocol": "ANY"
        }
      ],
      "id": "a34ca3f6-f3db-11e4-b010-005056806999",
      "fallbackCompatible": true
    }
  ],
  "pageNumber": 1,
  "pageCount": 4,
  "totalCount": 4,
  "pageSize": 250
}
```


11.2.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	A networkDomainId filter must be provided.

11.3. List Default iRules

11.3.1. Description

This API function is of relevance to **Create Virtual Listener** and **Edit Virtual Listener**.

Lists the default iRules at an MCP 2.0 data center specific to a Network Domain (networkDomainId) belonging to the organization identified by the {org-id} parameter.

Other filter values can be used to narrow the list returned as detailed below.

Each iRule declares the combination of Virtual Listener *type* and *protocol* with which it is.

11.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/networkDomainVip/defaultIrule?`
	
	`Filter Required Parameters:`
	`  networkDomainId=`
	
	`Filter Optional Parameters:`
	`  [&id=]`
	`  [&datacenterId=]`
	`  [&name=]`
	`  [&enabled=]`
	`  [&state=]`
	
	`Paging/Ordering Optional Parameters:`
	`  [&pageSize=]`
	`  [&pageNumber=]`
	`  [&orderBy=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
networkDomainId	uuid (String)	**Required**. Identifies the Network Domain, which the iRule correspond to. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
id	uuid (String)	Identifies an individual iRule. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
name	String	Identifies iRules by their *name*. *name=CCDEFAULT.lps* *S*upports the use of the LIKE comparator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	No

virtualListenerType	String	Filters by Virtual Listener compatibility type. See **Create Virtual Listener** for the set of valid values for Virtual Listener type. *virtualListenerType=STANDARD*	Yes	No
virtualListenerProtocol	String	Filters by Virtual Listener compatibility protocol. See **Create Virtual Listener** for the set of valid values for Virtual Listener protocol. *virtualListenerProtocol=TCP*	Yes	No

11.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<defaultIrules
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="5"
totalCount="5" pageSize="250">
    <defaultIrule>
        <irule id="2b20cb2c-ffdc-11e4-b010-005056806999"
name="CCDEFAULT.HttpsRedirect"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="HTTP"/>
    </defaultIrule>
    <defaultIrule>
        <irule id="2b20abd9-ffdc-11e4-b010-005056806999"
name="CCDEFAULT.IpProtocolTimers"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="HTTP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="TCP"/>
        <virtualListenerCompatibility type="STANDARD" protocol="UDP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="UDP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="ANY"/>
        <virtualListenerCompatibility type="STANDARD" protocol="TCP"/>
        <virtualListenerCompatibility type="STANDARD" protocol="ANY"/>
    </defaultIrule>
    <defaultIrule>
        <irule id="2b20e790-ffdc-11e4-b010-005056806999"
name="CCDEFAULT.Ips"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="HTTP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="TCP"/>
        <virtualListenerCompatibility type="STANDARD" protocol="UDP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="UDP"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="ANY"/>
        <virtualListenerCompatibility type="STANDARD" protocol="TCP"/>
        <virtualListenerCompatibility type="STANDARD" protocol="ANY"/>
    </defaultIrule>
    <defaultIrule>
        <irule id="2b210846-ffdc-11e4-b010-005056806999"
name="CCDEFAULT.IpsHttp"/>
        <virtualListenerCompatibility type="PERFORMANCE_LAYER_4"
protocol="HTTP"/>
    </defaultIrule>
```

```
</defaultIrules>
```

JSON

```
{
  "defaultIrule": [
    {
      "virtualListenerCompatibility": [
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "HTTP"
        }
      ],
      "irule": {
        "id": "2b20cb2c-ffdc-11e4-b010-005056806999",
        "name": "CCDEFAULT.HttpsRedirect"
      }
    },
    {
      "virtualListenerCompatibility": [
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "HTTP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "TCP"
        },
        {
          "type": "STANDARD",
          "protocol": "UDP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "UDP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "ANY"
        },
        {
          "type": "STANDARD",
          "protocol": "TCP"
        },
        {
          "type": "STANDARD",
          "protocol": "ANY"
        }
      ],
      "irule": {
        "id": "2b20abd9-ffdc-11e4-b010-005056806999",
        "name": "CCDEFAULT.IpProtocolTimers"
      }
    },
    {
      "virtualListenerCompatibility": [
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "HTTP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "TCP"
        },
        {
          "type": "STANDARD",
          "protocol": "UDP"
        },
```

```
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "UDP"
        },
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "ANY"
        },
        {
          "type": "STANDARD",
          "protocol": "TCP"
        },
        {
          "type": "STANDARD",
          "protocol": "ANY"
        }
      ],
      "irule": {
        "id": "2b20e790-ffdc-11e4-b010-005056806999",
        "name": "CCDEFAULT.Ips"
      }
    },
    {
      "virtualListenerCompatibility": [
        {
          "type": "PERFORMANCE_LAYER_4",
          "protocol": "HTTP"
        }
      ],
      "irule": {
        "id": "2b210846-ffdc-11e4-b010-005056806999",
        "name": "CCDEFAULT.IpsHttp"
      }
    }
  ],
  "pageNumber": 1,
  "pageCount": 5,
  "totalCount": 5,
  "pageSize": 250
}
```


11.3.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	A networkDomainId filter must be provided.

12. Network API – VLAN Security Groups

This set of APIs is used to manage VLAN Security Groups at MCP 2.0 data centers.

VLAN Security Groups are created a the VLAN level on **Advanced** type Network Domains.

A NICs that is connected to a given VLAN can be added to a Security Group, which will have the following effects in terms of both private IPv4 and IPv6 traffic within the VLAN:

- NICs in the same Security Group can communicate with each other.
- NICs in a different Security Group can NOT communicate with each other.
- If a NIC is not in a Security Group, it can communicate with any NIC on the VLAN including those in any Security Group as well as others not in a Security Group.

Note: this functionality applies only to traffic within the VLAN. Traffic with any other IPv4 / IPv6 address outside the VLAN is unaffected and is governed by Firewall Rules.

12.1. Create Security Group

12.1.1. Description

This function creates a Security Group for a given VLAN deployed to a Network Domain at an MCP 2.0 data center.

You can have one or more Security Groups on a given VLAN

12.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/securityGroup/createSecurityGroup`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<createSecurityGroup xmlns="urn:didata.com:api:cloud:types">
  <vlanId>bc529e20-dc6f-42ba-be20-0ffe44d1993f</vlanId>
  <name>SecurityGroupA</name>
  <description>Security group for a VLAN.</description>
</createSecurityGroup>
```

JSON Request Sample

```
{
    "vlanId": "bc529e20-dc6f-42ba-be20-0ffe44d1993f",
    "name": "SecurityGroupB",
    "description": "Security group for a VLAN."
}
```

Request Properties

Field	Required	Type and Constraints
vlanId	Yes	The UUID of a VLAN belonging to {org-id}.
name	Yes	Must be between 1 and 24 characters. Cannot contain spaces.
description	No	Must be 255 characters or less.

12.1.3. Response Details

The response includes an "*info*" property named "*securityGroupId*". The value of this property contains the unique identifier of the newly created Security Group and can be used in conjunction with **List Security Groups** to retrieve details for the Security Group.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response xmlns="urn:didata.com:api:cloud:types" requestId="na9/2015-11-
27T10:39:27.510-05:00/87b67ffb-c44f-423b-88e4-d3dc56206c36">
  <operation>CREATE_SECURITY_GROUP</operation>
  <responseCode>OK</responseCode>
  <message>Security group 'SecurityGroupA' has been created.</message>
  <info name="securityGroupId" value="327a46c7-8f81-4397-9859-11374a06192a"/>
</response>
```

JSON

```
{
  "operation": "CREATE_SECURITY_GROUP",
  "responseCode": "OK",
  "message": "Security group 'SecurityGroupB' has been created.",
  "info": [
    {
      "name": "securityGroupId",
      "value": "816100b3-90c8-4f03-b2ee-835b472493b7"
    }
  ],
  "warning": [],
  "error": [],
  "requestId": "na9/2015-11-27T10:41:37.787-05:00/c8bf7019-1c86-4cce-9e3a-
959513cea284"
}
```

12.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
EXCEEDS_LIMIT	A VLAN may only have a maximum of n security groups. A data center may only have a maximum of n Security Groups.
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> hypervisor is in a maintenance state.
INVALID_INPUT_DATA	Invalid name. Name is missing. Invalid name. Name includes spaces. Invalid name. Name must be between 1 and 24 characters. Invalid description: Description must be 255 characters or less.
INCOMPATIBLE_OPERATION	Security Groups are only supported for VLANs on Advanced Network Domains.
NAME_NOT_UNIQUE	Security Group name must be unique within the data center.
OPERATION_NOT_SUPPORTED	VLAN Security Groups are not available in data center <datacenterId>.
RESOURCE_BUSY	Another operation is in progress on VLAN with Id {id}. Please try again later.

RESOURCE_LOCKED	VLAN with Id {id} is locked. Please contact support.
RESOURCE_NOT_FOUND	VLAN <vlanId> not found. *The value passed for vlanId does not correspond to a valid VLAN belonging to {org-id}. See **List VLANs**.*

12.2. List Security Groups

12.2.1. Description

Lists all of the Security Groups on a particular VLAN, belonging to the organization identified by the {org-id} parameter.

Various filter values can be used to narrow the list returned as detailed below.

12.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/securityGroup/securityGroup?`
	
	`Filter Required Parameters:`
	`  vlanId= or`
	`  serverId=`
	
	`Filter Optional Parameters:`
	`  [&id=]`
	`  [&state=]`
	`  [&name=]`
	`  [&createTime=]`
	
	`Paging/Ordering Optional Parameters:`
	`  [&pageSize=]`
	`  [&pageNumber=]`
	`  [&orderBy=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
vlanId **or** serverId	uuid (String)	**Required**. Identifies an individual VLAN or Cloud Server (i.e. a set of NICs). *vlanId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
id	uuid (String)	Identifies an individual Security Group. *id=j6ghb7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
state	String	Identifies Security Groups by their *state*. Case insensitive. The initial possible set of values for *state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD",	Yes	Yes

		"FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time. *state=NORMAL*		
name	String	Identifies a partial or whole Security Group name. Support the LIKE operator. *name=MySecurityGroup*	Yes	Yes
createTime	Date	Identifies the date of creation of NAT Rules. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes

12.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<securityGroups xmlns="urn:didata.com:api:cloud:types" pageNumber="1"
pageCount="2" totalCount="2" pageSize="300">
  <securityGroup id="327a46c7-8f81-4397-9859-11374a06192a">
    <name>SecurityGroupA</name>
    <description>My security group</description>
    <state>NORMAL</state>
    <vlanId>bc529e20-dc6f-42ba-be20-0ffe44d1993f</vlanId>
    <createTime>2015-11-27T15:39:27.000Z</createTime>
  </securityGroup>
  <securityGroup id="816100b3-90c8-4f03-b2ee-835b472493b7">
    <name>SecurityGroupB</name>
    <description>Another security group</description>
    <state>NORMAL</state>
    <vlanId>bc529e20-dc6f-42ba-be20-0ffe44d1993f</vlanId>
    <createTime>2015-11-27T15:41:37.000Z</createTime>
  </securityGroup>
</securityGroups>
```

JSON

```
{
  "securityGroup": [
    {
      "name": "SecurityGroupA",
      "description": "My security group",
      "state": "NORMAL",
      "nic": [],
      "vlanId": "bc529e20-dc6f-42ba-be20-0ffe44d1993f",
      "createTime": "2015-11-27T15:39:27.000Z",
      "id": "327a46c7-8f81-4397-9859-11374a06192a"
    },
    {
      "name": "SecurityGroupB",
      "description": "Another security group",
      "state": "NORMAL",
      "nic": [],
      "vlanId": "bc529e20-dc6f-42ba-be20-0ffe44d1993f",
      "createTime": "2015-11-27T15:41:37.000Z",
      "id": "816100b3-90c8-4f03-b2ee-835b472493b7"
```

```
    }
  ],
  "pageNumber": 1,
  "pageCount": 2,
  "totalCount": 2,
  "pageSize": 300
}
```


12.2.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

Response Code	Message
INVALID_INPUT_DATA	At least one of vlanId or serverId must be supplied.

12.3. Add NIC to Security Group

12.3.1. Description

Adds a Server NIC corresponding to a particular VLAN to a Security Group created for the same VLAN, belonging to the organization identified by {org-id}.

12.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/securityGroup/addNicToSecurityGroup`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<addNicToSecurityGroup securityGroupId="327a46c7-8f81-4397-9859-11374a06192a"
xmlns:="urn:didata.com:api:cloud:types">
    <nicId>4015d3e6-0880-4a39-9db2-3d05375b92ea</nicId>
</addNicToSecurityGroup>
```

JSON Request Sample

```
{
    "securityGroupId": "327a46c7-8f81-4397-9859-11374a06192a",
    "nicId": "4015d3e6-0880-4a39-9db2-3d05375b92ea"
}
```

Request Properties

Field	Required	Type and Constraints
securityGroupId	Yes	Identifies the Security Group to which the Server NIC will be added. *See **List Security Groups.***
nicId	Yes	Identifies the Server NIC being added to the Security Group. The NIC must correspond to the same VLAN to which the Security Group identified by *securityGroupId* was added.

12.3.3. Response Details

XML

```
<response xmlns="urn:didata.com:api:cloud:types" requestId="na9/2015-11-
27T11:41:55.192-05:00/87abf64c-3047-431f-8fb8-a8c16287a1a5">
  <operation>ADD_NIC_TO_SECURITY_GROUP</operation>
  <responseCode>OK</responseCode>
  <message>Server NIC with Id 4015d3e6-0880-4a39-9db2-3d05375b92ea was
successfully added to Security Group with Id 327a46c7-8f81-4397-9859-
11374a06192a.</message>
</response>
```

JSON

```
{
  "operation": "ADD_NIC_TO_SECURITY_GROUP",
  "responseCode": "OK",
  "message": "Server NIC with Id 4015d3e6-0880-4a39-9db2-3d05375b92ea was
successfully added to Security Group with Id 327a46c7-8f81-4397-9859-
```

```
11374a06192a.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "na9/2015-11-27T12:02:47.348-05:00/feb377a9-e80e-46d0-b511-
7393df171bd2"
}
```

12.3.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
EXCEEDS_LIMIT	A Security Group may have a maximum of *n* NICs.
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> hypervisor is in a maintenance state.
INVALID_INPUT_DATA	securityGroupId must be provided. nicId must be provided.
NOT_UNIQUE	NIC <nicId> is already a member of Security Group <securityGroupId>.
RESOURCE_BUSY	Another operation is in progress on Security Group with id <securityGroupId>. Please try again later. Another operation is in progress on NIC with id <nicId>. Please try again later.
RESOURCE_LOCKED	Security Group with Id <securityGroupId> is locked. Please contact support. NIC with Id <nicId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Security Group <securityGroupId> not found. NIC <nicId> not found. See **List Servers**. NIC <nicId> is not in the VLAN associated with this Security Group. *Refer to the filter options on **List Security Groups**, **List VLANs** and **List Servers** to ensure the three cases above are avoided.*

12.4. Remove NIC from Security Group

12.4.1. Description

Removes a Server NIC from a VLAN Security Group owned by the organization identified by {org-id}.

12.4.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/securityGroup/removeNicFromSecurityGroup`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<removeNicFromSecurityGroup securityGroupId="327a46c7-8f81-4397-9859-
11374a06192a" xmlns="urn:didata.com:api:cloud:types">
    <nicId>4015d3e6-0880-4a39-9db2-3d05375b92ea</nicId>
</removeNicFromSecurityGroup>
```

JSON Request Sample

```
{
    "securityGroupId": "327a46c7-8f81-4397-9859-11374a06192a",
    "nicId": "4015d3e6-0880-4a39-9db2-3d05375b92ea"
}
```

Request Properties

Field	Required	Type and Constraints
securityGroupId	Yes	Identifies the Security Group from which the Server NIC will be removed. *See **List Security Groups.***
nicId	Yes	Identifies the Server NIC being removed from the Security Group.

12.4.3. Response Details

XML

```
<response xmlns="urn:didata.com:api:cloud:types" requestId="na9/2015-11-
27T11:58:30.691-05:00/df430185-37b3-4fc2-8701-ee7a18000c14">
  <operation>REMOVE_NIC_FROM_SECURITY_GROUP</operation>
  <responseCode>OK</responseCode>
  <message>Server NIC with Id 4015d3e6-0880-4a39-9db2-3d05375b92ea was
successfully removed from Security Group with Id 327a46c7-8f81-4397-9859-
11374a06192a.</message>
</response>
```

JSON

```
{
  "operation": "REMOVE_NIC_FROM_SECURITY_GROUP",
  "responseCode": "OK",
  "message": "Server NIC with Id 4015d3e6-0880-4a39-9db2-3d05375b92ea was
successfully removed from Security Group with Id 327a46c7-8f81-4397-9859-
11374a06192a.",
  "info": [],
  "warning": [],
  "error": [],
```

```
  "requestId": "na9/2015-11-27T12:04:50.017-05:00/70aefc4f-b7e2-4461-a75f-
8ccc7fc33ead"
}
```

12.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> hypervisor is in a maintenance state.
INVALID_INPUT_DATA	securityGroupId must be provided. nicId must be provided.
RESOURCE_BUSY	Another operation is in progress on Security Group with id <securityGroupId>. Please try again later. Another operation is in progress on NIC with id <nicId>. Please try again later.
RESOURCE_LOCKED	Security Group with Id <securityGroupId> is locked. Please contact support. NIC with Id <nicId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Security Group <securityGroupId> not found. NIC <nicId> not found. See **List Servers**. NIC <nicId> is not in the VLAN associated with this Security Group. *Refer to the filter options on **List Security Groups**, **List VLANs** and **List Servers** to ensure the three cases above are avoided.*

12.5. Edit Security Group

12.5.1. Description

Edits the modifiable properties of a Security Group belonging to the organization identified by {org-id}.

12.5.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/securityGroup/editSecurityGroup`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<editSecurityGroup id="327a46c7-8f81-4397-9859-11374a06192a"
xmlns="urn:didata.com:api:cloud:types">
  <name>SecurityGroupC</name>
  <description>This is an updated Security Group</description>
</editSecurityGroup>
```

JSON Request Sample

```
{
    "id": "816100b3-90c8-4f03-b2ee-835b472493b7",
    "name": "SecurityGroupA",
    "description": "This is an updated Security Group"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Security Group to be edited.
name	No*	
description	No*	

*Note that at least one of *name* or *description* must be provided.

12.5.3. Response Details

XML

```
<response xmlns="urn:didata.com:api:cloud:types" requestId="na9/2015-11-
27T12:13:25.705-05:00/19d4d63c-52eb-40bd-b3bb-e7f994225e23">
  <operation>EDIT_SECURITY_GROUP</operation>
  <responseCode>OK</responseCode>
  <message>Security Group with Id 327a46c7-8f81-4397-9859-11374a06192a was
successfully edited.</message>
</response>
```

JSON

```
{
  "operation": "EDIT_SECURITY_GROUP",
  "responseCode": "OK",
  "message": "Security Group with Id 816100b3-90c8-4f03-b2ee-835b472493b7 was
successfully edited.",
  "info": [],
```

```
  "warning": [],
  "error": [],
  "requestId": "na9/2015-11-27T12:17:11.904-05:00/4cee5c1a-6e04-48c7-9e73-
e8542ef9a62e"
}
```

12.5.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> hypervisor is in a maintenance state.
INVALID_INPUT_DATA	id must be provided. Invalid name. Name includes spaces. Invalid name. Name must be between 1 and 24 characters. Invalid description: Description must be 255 characters or less. At least one of name or description must be supplied.
NAME_NOT_UNIQUE	Security Group name must be unique within the data center.
RESOURCE_BUSY	Another operation is in progress on Security Group with id <id>. Please try again later.
RESOURCE_LOCKED	Security Group with Id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Security Group <id> not found. *The value passed for id does not correspond to a valid Security Group belonging to {org-id}. See **List Security Groups**.*

12.6. Delete Security Group

12.6.1. Description

Deletes a Security Group belonging to the organization identified by {org-id}.

12.6.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/securityGroup/deleteSecurityGroup`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "network"

XML Request Sample

```
<deleteSecurityGroup id="327a46c7-8f81-4397-9859-11374a06192a"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
    "id": "816100b3-90c8-4f03-b2ee-835b472493b7"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Security Group to be deleted.

12.6.3. Response Details

XML

```
<response xmlns="urn:didata.com:api:cloud:types" requestId="na9/2015-11-
27T12:31:19.535-05:00/97186c51-4131-4046-8236-351a28648509">
  <operation>DELETE_SECURITY_GROUP</operation>
  <responseCode>OK</responseCode>
  <message>Security Group (Id: 327a46c7-8f81-4397-9859-11374a06192a) has been
deleted.</message>
</response>
```

JSON

```
{
  "operation": "DELETE_SECURITY_GROUP",
  "responseCode": "OK",
  "message": "Security Group (Id: 816100b3-90c8-4f03-b2ee-835b472493b7) has
been deleted.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "na9/2015-11-27T12:32:20.544-05:00/fbf05b47-4d79-4c13-85d3-
e0db1c5d8df4"
}
```

12.6.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
HAS_DEPENDENCY	There is at least one NIC associated with the VLAN Security Group. All such NICs must be removed before the VLAN Security Group can be deleted. *See **List Security Groups** and **Remove NIC from Security Group**.*
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> hypervisor is in a maintenance state.
INVALID_INPUT_DATA	Security Group id is required.
RESOURCE_BUSY	Another operation is in progress on Security Group with Id<id>. Please try again later.
RESOURCE_LOCKED	Security Group with Id <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Security Group <id> not found. *The value passed for <id> does not correspond to a valid Security Group belonging to {org-id}. See **List Security Groups**.*

13. Server API – Server Management

13.1. Deploy Server

13.1.1. Description

This function deploys a new Cloud Server.

In **MCP 1.0** data centers the function deploys the Server to a Cloud Network using the single required NIC to associate it with the Cloud Network.

In **MCP 2.0** data centers the function deploys the Server on a Network Domain with a Primary required NIC to associate it with a VLAN on the Network Domain, and a number of optional additional NICs. All NICs must relate to VLANs deployed on the same Network Domain.

13.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/deployServer`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

We recommend referring to the XSD for this function as it uses the Schema "choice" element to separate different properties depending on the nature of the request. Viewing the XSD will also help clarify the difference for JSON users.

For an **MCP 1.0** request, a *network* element is required. *network* can contain either a Cloud Network ID (network*Id*) **OR** a Private IPv4 address (*privateIpv4*) from the target Cloud Network to which the server will be deployed.

For an **MCP 2.0** request, a *networkInfo* element is required. *networkInfo* identifies the Network Domain to which the Server will be deployed. It contains a *primaryNic* element defining the required NIC for the Server and optional *additionalNic* elements defining any additional VLAN connections for the Server. Each NIC must contain either a VLAN ID (*vlanId*) **OR** a Private IPv4 address (*privateIpv4*) from the target VLAN which the NIC will associate the Server with.

Note that the sample requests below are specific to a source Server Image (identified by *imageId*) which has a particular disk structure.

MCP 1.0 Request Samples

XML Request Sample (using networkId)

```
<deployServer xmlns="urn:didata.com:api:cloud:types">
  <name>Production FTPS Server</name>
  <description>This is the main FTPS Server</description>
  <imageId>e0b48a16-f5c3-4045-bfd0-a1a7232a2710</imageId>
  <start>true</start>
  <administratorPassword>P$$ssWwrrdGoDd!</administratorPassword>
  <cpu speed="HIGHPERFORMANCE" count="4" coresPerSocket="2"/>
  <memoryGb>4</memoryGb>
  <network>
    <networkId>c550be0e-65c1-11e4-811f-005056806999</networkId>
  </network>
  <primaryDns>10.20.255.12</primaryDns>
  <secondaryDns>10.20.255.13</secondaryDns>
  <disk scsiId="0" speed="STANDARD"/>
  <microsoftTimeZone>040</microsoftTimeZone>
</deployServer>
```

JSON Request Sample (using privateIpv4)

```
{
  "imageId": "8241f16d-3cda-4030-a9fe-6233cc94990b",
  "start": true,
  "disk": [
    {
```

```
      "speed": "STANDARD",
      "scsiId": "0"
    }
  ],
  "administratorPassword": "P$$ssWwrrdGoDd!",
  "cpu": {
    "count": 4,
    "coresPerSocket": 1,
    "speed": "STANDARD"
  },
  "memoryGb": 4,
  "description": "This is the main FTPS Server",
  "network": {
    "privateIpv4": "10.160.117.45"
  },
  "primaryDns": "10.160.117.125",
  "secondaryDns": "10.160.117.125",
  "name": "Production FTPS Server (Local DNS)"
}
```

MCP 2.0 Request Samples

XML Request Sample (using mixture of vlanId and privateIpv4)

```
<deployServer xmlns="urn:didata.com:api:cloud:types">
  <name>Production Server</name>
  <description>Main application server</description>
  <imageId>18591fcd-c354-4f92-a222-d168176340be</imageId>
  <start>false</start>
  <administratorPassword>P$$ssWwrrdGoDd!</administratorPassword>
  <cpu speed="HIGHPERFORMANCE" count="4" coresPerSocket="1"/>
  <memoryGb>2</memoryGb>
  <networkInfo networkDomainId="553f26b6-2a73-42c3-a78b-6116f11291d0">
    <primaryNic>
      <privateIpv4>10.0.4.14</privateIpv4>
    </primaryNic>
    <additionalNic>
      <privateIpv4>172.16.0.14</privateIpv4>
    </additionalNic>
    <additionalNic>
      <vlanId>e0b4d43c-c648-11e4-b33a-72802a5322b2</vlanId>
    </additionalNic>
  </networkInfo>
  <primaryDns>10.20.255.12</primaryDns>
  <secondaryDns>10.20.255.13</secondaryDns>
  <disk scsiId="0" speed="STANDARD"/>
  <disk scsiId="2" speed="HIGHPERFORMANCE"/>
  <microsoftTimeZone>040</microsoftTimeZone>
</deployServer>
```

JSON Request Sample (using mixture of vlanId and privateIpv4)

```
{
  "name":"Production FTPS Server",
  "description":"This is the main FTPS Server",
  "imageId":"02250336-de2b-4e99-ab96-78511b7f8f4b",
  "start":true,
  "administratorPassword":"P$$ssWwrrdGoDd!",
  "cpu":{
   "count":4,
   "coresPerSocket":1,
   "speed":"STANDARD"
  },
  "memoryGb":4,
  "primaryDns":"10.20.255.12",
  "secondaryDns":"10.20.255.13",
  "networkInfo": {
    "networkDomainId":"484174a2-ae74-4658-9e56-50fc90e086cf",
    "primaryNic" : {"vlanId":"0e56433f-d808-4669-821d-812769517ff8"},
```

```
    "additionalNic" : [
       {"privateIpv4" : "172.16.0.14"},
       {"vlanId":"e0b4d43c-c648-11e4-b33a-72802a5322b2"}
     ]
  },
  "disk" : [{
   "scsiId" :"0" ,
   "speed" :"STANDARD"
  },{
   "scsiId" :"1" ,
   "speed" :"HIGHPERFORMANCE"
  }],
  "microsoftTimeZone":"035"
}
```

Request Properties

Fields are marked as MCP 1.0 or MCP 2.0 specific where relevant.

Field	Required	Type and Constraints
name	Yes	Minimum length: 1 character Maximum length: 75 characters.
description	No	Maximum length: 255 characters.
imageId	Yes	UUID of the Server Image being used as the target for the new Server deployment. See **List OS Images** and **List Customer Images**.
start	Yes	Boolean. "true" indicates that the Server will be started following deployment.
administratorPassword	n/a	Required if a Customer Image is not being used as *imageId*. Required for Windows Customer Images except for Windows 2003. Should **not be provided** if *imageId* represents a Linux Customer Images or Windows 2003 Customer Image.
cpu.speed	No	String. Required if *cpu* is specified. Determines the CPU Speed to be used across all of the CPUs on the Server. If the source Server Image is an OS Image and CPU Speed is **not supplied**, then the value used will be the *default for the Data Center. If the source Server Image is a Customer Image and CPU Speed is **not supplied**, then the value used will be the value defined on the Customer Image if it is available at the data center. See **List Data Centers** for the set of CPU Speeds which are valid for the data center, and to reference which is the *default.
cpu.count	No	Integer. Required if *cpu* is specified. Overrides the number of CPUs specified on the source Server Image. Note the relationship between this value and the *coresPerSocket* value. See **List Data Centers** for the *MAX_CPU_COUNT* property which definies the maximum allowed for this element.
cpu.coresPerSocket	No	Integer. Required if *cpu* is specified and must be an integer factor of CPU *count*. The default and recommended value for cores per socket is

		1. Cores per socket is an advanced setting, which can adversely affect performance if adjusted without understanding what it is doing. Please refer to the following article: VMware vSphere Blog: Does corespersocket Affect Performance?
memoryGb	No	Can be used to override the memory value inherited from the source Server Image (*imageId*).
network ***or*** *networkInfo*	Yes	MCP 1.0: network is required. MCP 2.0: networkInfo is required.
networkInfo.networkDomainId	n/a	*networkInfo* defines all networking connectivity associated with the Server. The *networkDomainId* field is mandatory for MCP 2.0 data center server deployments and is the UUID of a Network Domain belonging to {org-id} within the same MCP 2.0 data center. The *networkDomainId* field can be omitted for MCP 1.0 server deployments.
networkInfo.primaryNic: *vlanId* or *privateIpv4*	Yes	A single Primary NIC is required for every Server. It defines the VLAN (MCP 2.0) or Cloud Network (MCP 1.0), which is permanently associated with the Server. The networkInfo.primaryNic section can contain one of two fields. Either a *vlanId* containing a valid UUID for a VLAN on the same Network Domain as networkDomainId **OR** A *privateIpv4* containing an RFC1918 dot-decimal IPv4 address, which must not be in the system reserved space for the VLAN. The system reserved IP addresses are the first six (.0 - .5), eighth (.7) and broadcast address (the last .255 in the VLAN). For example: “10.0.4.15”.
networkInfo.additionalNic: *vlanId* or *privateIpv4*	No	Multiple Additional NICs can be provided for a Server during deployment or added after deployment using the **Add NIC** function. Each *networkInfo.additionalNic* section can contain one of two fields. Either a *vlanId* containing a valid UUID for a VLAN on the same Network Domain as networkDomainId **OR** A *privateIpv4* containing an RFC1918 dot-decimal IPv4 address, which must not be in the system reserved space for the VLAN. The system reserved IP addresses are the first six (.0 - .5), eighth (.7) and broadcast address (the last .255 in the VLAN). For example: “10.0.4.15”. The maximum number of additional NICs that can be added is governed by the Data Center hypervisor type, 9 for VMware data centers and 7 for Hyper-V Private MCP data centers.
network.*networkId* or network.*privateIpv4*or	Yes	For an MCP 1.0 deployment one of these properties must be provided within the *network* element. *networkId* identifies a

		specific Cloud Network by its ID and will result in the next available private IP address being assigned to the newly deployed Server. *privateIpv4* identifies a specific, available IPv4 address on a Cloud Network at the relevant datacenter (identified from the source image).
disk.scsiId	Yes	Optional disk elements can be used to define the disk speed that each disk on the Server – inherited from the source Server Image - will be deployed to. It is not necessary to include a disk element for every disk; only those that you wish to set a disk speed value for. Each disk is identified using its SCSI ID, which can be retrieved from the 0.9 API **List Server Images (with Parameters)** function. Note that scsiId 7 cannot be used. Up to 13 disks can be present in addition to the required OS disk on SCSI ID 0.
disk.speed	Yes	The speed of storage on to which the disk will be deployed. The available speeds can vary from data center to data center. They are enumerated on the **List Data Centers** API response. See the additional notes below for further information about disk speeds and server deployment.
microsoftTimeZone	No	String. For use with Microsoft Windows source Server Images (*imageId*) only. For the exact value to use please refer to the table of time zone indexes in the following Microsoft Technet documentation: [GuiUnattended] (Unattended Installation) If none is supplied, the default time zone for the data center's geographic region will be used.

Disk Speed

The Deploy Server function includes the option to define speeds for the various disks attached to the source Server Image - including the operating system disk on SCSI 0 - which will be created on the resulting Server.

A speed can be supplied for *any*, *some* or *none* of the disks associated with a given source Server Image using the <disk> field shown in the sample payloads above. The speed supplied must be valid for the data center in which the server is being deployed and will result in the disk being persisted on the appropriate type of storage when the server is deployed. See **List Data Centers**.

Please refer to the related Community article prior to using disk speed options available on this API. The article explains the concept of Tiered Storage in the context of CloudControl and what it means for your Cloud Servers' storage:

Introduction to Cloud Server Disks and Disk Speeds (Tiered Storage)

If a ***disk*** field is not supplied for a particular disk on the source Server Image there are a couple of possible outcomes:

- If the source Server Image is a Customer Image and was cloned from a Server on which disk speeds have already been configured, the speeds of those disks will have been retained as meta data. The retained speed will be used when creating the disk on the new server. Note that if a speed defined on the Server Image is not available at the target data center a synchronous API response will be returned explaining the problem.
- If the source image is a stock OS Server Image or has no custom speed information for a given disk, the default disk speed for the data center will be used for the disk.

13.1.3. Response Details

The response includes an "*info*" property named "*serverId*". The value of this property contains the unique identifier of the VLAN being deployed and can be used in conjunction with **Get Server** and or **List Servers** to retrieve the status of this asynchronous operation.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-03-
08T10:43:34.168-04:00/7c4ea967-1723-4a06-80e2-fcdf50f3fa82">
    <operation>DEPLOY</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to deploy Server 'Production FTPS Server' has been
accepted and is being processed.</message>
    <info name="serverId" value="28ba239f-c635-4a64-9d58-b6579ee86aa1"/>
</response>
```

JSON

```
{
    "operation": "DEPLOY_SERVER",
    "responseCode": "IN_PROGRESS",
    "message": "Request to deploy Server 'Production FTPS Server' has been
accepted and is being processed.",
    "info": [
        {
            "name": "serverId",
            "value": "7b62aae5-bdbe-4595-b58d-c78f95db2a7f"
        }
    ],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-08T11:06:08.620-04:00/44b71b17-9a68-4f3f-b628-
eedfc301e699"
}
```

13.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	The maximum number of additional NICs allowed on a Server in a <datacenter.hypervisor.type> data center is <maxValue>. This data center does not support the operating system <operatingSystemType> on Image <imageId>.
DISK_SPEED_NOT_AVAILABLE	Disk Speed <speed> is not available in data center <datacenterId>.
EXCEEDS_LIMIT	An unverified Organization may deploy a maximum of <maxValue> Servers in a calendar day. *Please contact your account representative for verification.*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
INVALID_INPUT_DATA	administratorPassword must be at least 8 characters in length and cannot include '<' or '>'. administratorPassword is mandatory when the imageId corresponds to any OS Image.

	administratorPassword is mandatory when the imageId corresponds to a Windows Customer Image excluding Windows Server 2003. administratorPassword must not be provided if the imageId corresponds to a Linux Customer Image or Windows 2003 Customer Image. Invalid privateIpv4 <privateIpv4>. Must be a valid IPv4 in dot-decimal notation (x.x.x.x). privateIpv4 must be a dot-decimal representation of an IPv4 /24 address in the RFC 1918 space. A Server cannot have more than 1 NIC associated with the same VLAN <vlanId>. Cannot deploy Server with Software Labels in the "Stopped" state (the *start* field must be set to true).
IP_ADDRESS_NOT_UNIQUE	privateIpv4 <privateIpv4> is already reserved. See **List Reserved Private IPv4 Addresses**.
IP_ADDRESS_OUT_OF_RANGE	privateIpv4 <privateIpv4> is outside the permitted range (<VLAN specific>.8 to <VLAN specific>.254). *See note above about system reserved space.* (MCP 2.0) Private IP w.x.y.z is outside the permitted range (w.x.y.11 to w.x.y.254). (MCP 1.0)
NO_IP_ADDRESS_AVAILABLE	There are no remaining unreserved IPv4 addresses within the permitted range of VLAN/Network <vlanId | networkId >.
RESOURCE_BUSY	Another operation is in progress on Server Image with id <imageId>. Please try again later. Another operation is in progress on Network Domain <networkDomainId>. Please try again later. (MCP 2.0) Another operation is in progress on VLAN <vlanId>. Please try again later. (MCP 2.0) Another operation is in progress on Network <networkId>. Please try again later. (MCP 1.0)
RESOURCE_LOCKED	Server Image with id <serverImageId> is locked. Please contact support. Network Domain <networkDomainId> is locked. Please contact support. (MCP 2.0) VLAN <vlanId> is locked. Please contact support. (MCP 2.0) Network <networkId> is locked. Please contact support. (MCP 1.0)
RESOURCE_NOT_FOUND	Server Image <serverImageId> not found. Network Domain <networkDomainId> not found (MCP 2.0). VLAN <vlanId> is not deployed on Network Domain <networkDomainId> (MCP 2.0). VLAN not found for Network Domain <networkDomainId> and IPv4 Address <privateIpv4> (MCP 2.0). Network not found for {org-id} and IPv4 Address <privateIpv4> (MCP 1.0). Could not find Network <networkId> (MCP 1.0). Disk <scsiId> not found in the Server Image

	<serverImageId>.

13.2. List Servers

13.2.1. Description

Lists all of the Servers at the geographic region belonging to the organization identified by the {org-id} parameter. Various filter values can be used to narrow the list returned as detailed below.

It is more efficient to use **Get Server** than **List Servers** (filtered by *id*) if you wish to retrieve *state* status or details of a single Server.

Of particular interest for narrowing the list of servers are the *datacenterId*, *networkDomainId* and *vlanId* filters described on more detail below.

13.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/server[?]`
	
	`Paging/Ordering Optional Parameters:`
	`  [&pageSize=]`
	`  [&pageNumber=]`
	`  [&orderBy=]`
	
	`Filter Optional Parameters:`
	`  [&id=]`
	`  [&datacenterId=]`
	`  [&networkDomainId=]`
	`  [&networkId=]`
	`  [&vlanId=]`
	`  [&sourceImageId=]`
	`  [&deployed=]`
	`  [&name=]`
	`  [&createTime=]`
	`  [&state=]`
	`  [&started=]`
	`  [&operatingSystemId=]`
	`  [&ipv6=]`
	`  [&privateIpv4=]`

Type	HTTP GET
Supported for	MCP 1.0 and MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual Server. *id=57gthb7c-37bd-11e2-a91c-0030487e546*	Yes	Yes
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes
networkDomainId	uuid (String)	Identifies an individual Network Domain. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No

networkId	uuid (String)	Identifies an individual Cloud Network (MCP 1.0). *Synonym to vlanId.* *networkId=b45tht7b7c-37bd-11e2-a91c-0030487e0306*	Yes	No
vlanId	uuid (String)	Identifies an individual VLAN (MCP 2.0). *Synonym to networkId.* *vlanId=b45hy7b7c-37bd-11e2-a91c-0030487e0306*	Yes	No
sourceImageId	uuid (String)	Identifies the Server Image used to deploy the Server. *sourceImageId=54567b7c-37bd-11e2-a91c-0030487e0309*	Yes	Yes
deployed	Boolean	This will be set to "true" when a Server has completed the deployment process.	Yes	Yes
name	String	Identifies Servers by their *name*. *name=ProductionServer* *S*upports the use of the LIKE operator defined in the ***Paging and Filtering for List API Functions*** overview.	Yes	Yes
createTime	Date	Identifies the date of creation of Servers. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes
state	String	Identifies Servers by their *state*. Case insensitive. The initial possible set of values for *state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.	Yes	Yes
started	Boolean	A boolean describing whether or not the server is currently running. *started=false*	No	Yes

operatingSystemId	uuid (String)	A human readable UUID, which names maps to the Operating System of the Server. *operatingSystemId=REDHAT664*	No	Yes
privateIpv4	String	The private IPv4 address of the Primary NIC on the Server. *privateIpv4=10.1.1.4*	Yes	Yes
ipv6	String	The IPv6 address of the Primary NIC on the Server. *ipv6= 2607:f480:1111:1102:0:0:0:0*	Yes	Yes

13.2.3. Response Details

We recommend referring to the XSD for this function as it uses the Schema "choice" element to separate different properties depending on the Data Center where a given Server is deployed. Viewing the XSD will also help clarify the difference for JSON users.

For an **MCP 1.0** Data Center, a *network* element is returned describing the networking configuration of the Server for the Cloud Network to which it is attached.

For an **MCP 2.0** Data Center, a *networkInfo* element is returned. *networkInfo* describes the Network Domain to which the Server is deployed. It contains a *primaryNic* element defining the required NIC for the Server and optional *additionalNic* elements defining any additional VLAN connections for the Server. Each NIC contains VLAN and IP information describing its connection to the given VLAN.

XML

An MCP 1.0 Server followed by an MCP 2.0 server:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<servers xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="2"
totalCount="2" pageSize="250">
<!-- MCP 1.0 Server -->
  <server id="6e160786-4136-41fd-b850-48648b72b679" datacenterId="NA1">
    <name>Production Web Server MCP 1</name>
    <description>nopassword0</description>
    <operatingSystem id="REDHAT632" displayName="REDHAT6/32" family="UNIX"/>
    <cpu count="4" speed="STANDARD" coresPerSocket="1"/>
    <memoryGb>2</memoryGb>
    <disk id="74f81c56-96cc-4cca-b4d7-a88f641a6ea2" scsiId="0" sizeGb="10"
speed="STANDARD" state="NORMAL"/>
    <nic id="43b24e9e-c1c9-4d53-965b-89bcaa725103" privateIpv4="10.160.117.25"
networkId="c550be0e-65c1-11e4-811f-005056806999" networkName="Test1"
state="NORMAL"/>
    <backup assetId="5579f3a7-4c32-4cf5-8a7e-b45c36a35c10"
servicePlan="Enterprise" state="NORMAL"/>
    <monitoring monitoringId="11049" servicePlan="ESSENTIALS" state="NORMAL"/>
    <sourceImageId>e9ec6eb4-4634-49de-b914-01eb74da5fb9</sourceImageId>
    <createTime>2015-08-11T16:51:05.000Z</createTime>
    <deployed>true</deployed>
    <started>true</started>
    <state>NORMAL</state>
    <vmwareTools versionStatus="NEED_UPGRADE" runningStatus="RUNNING"
apiVersion="8389"/>
    <virtualHardware version="vmx-08" upToDate="false"/>
  </server>
<!-- MCP 2.0 Server -->
  <server id="5a32d6e4-9707-4813-a269-56ab4d989f4d" datacenterId="NA9">
    <name>Production Web Server MCP 2</name>
    <description>Server to host our main web application.</description>
    <operatingSystem id="WIN2008S32" displayName="WIN2008S/32"
family="WINDOWS"/>
    <cpu count="2" speed="STANDARD" coresPerSocket="1"/>
```

```
    <memoryGb>4</memoryGb>
    <disk id="c2e1f199-116e-4dbc-9960-68720b832b0a" scsiId="0" sizeGb="50"
speed="STANDARD" state="NORMAL"/>
    <networkInfo networkDomainId="553f26b6-2a73-42c3-a78b-6116f11291d0">
      <primaryNic id="5e869800-df7b-4626-bcbf-8643b8be11fd"
privateIpv4="10.0.4.8" ipv6="2607:f480:1111:1282:2960:fb72:7154:6160"
vlanId="bc529e20-dc6f-42ba-be20-0ffe44d1993f" vlanName="Production VLAN"
state="NORMAL"/>
    </networkInfo>
    <backup assetId="91002e08-8dc1-47a1-ad33-04f501c06f87"
servicePlan="Advanced" state="NORMAL"/>
    <monitoring monitoringId="11039" servicePlan="ESSENTIALS" state="NORMAL"/>
    <softwareLabel>MSSQL2008R2S</softwareLabel>
    <sourceImageId>3ebf3c0f-90fe-4a8b-8585-6e65b316592c</sourceImageId>
    <createTime>2015-12-02T10:31:33.000Z</createTime>
    <deployed>true</deployed>
    <started>true</started>
    <state>PENDING_CHANGE</state>
    <progress>
      <action>SHUTDOWN_SERVER</action>
      <requestTime>2015-12-02T11:07:40.000Z</requestTime>
      <userName>devuser1</userName>
    </progress>
    <vmwareTools versionStatus="CURRENT" runningStatus="RUNNING"
apiVersion="9354"/>
    <virtualHardware version="vmx-08" upToDate="false"/>
  </server>
</servers>
```

JSON

An MCP 1.0 Server followed by an MCP 2.0 server:

```
{
"server": [
    {
      "name": "Production Web Server MCP 1",
      "description": "Hosts our web applications.",
      "operatingSystem": {
        "id": "REDHAT664",
        "displayName": "REDHAT6/64",
        "family": "UNIX"
      },
      "cpu": {
        "count": 1,
        "speed": "STANDARD",
        "coresPerSocket": 1
      },
      "memoryGb": 2,
      "disk": [
        {
          "id": "c10ca998-3e40-4e86-b0ed-54b589535143",
          "scsiId": 0,
          "sizeGb": 10,
          "speed": "STANDARD",
          "state": "NORMAL"
        }
      ],
      "nic": {
        "id": "a91a8e7d-8298-4409-b62c-d89e7e7501be",
        "privateIpv4": "10.160.16.11",
        "networkId": "076e08de-13ec-11e3-a91c-0030487e0302",
        "networkName": "test",
        "state": "NORMAL"
      },
      "backup": {
        "assetId": "5579f3a7-4c32-4cf5-8a7e-b45c36a35c10",
        "servicePlan": "Enterprise",
        "state": "NORMAL"
```

```
      },
      "monitoring": {
        "monitoringId": "11049",
        "servicePlan": "ESSENTIALS",
        "state": "NORMAL"
      },
      "softwareLabel": [],
      "sourceImageId": "42dceafc-f08c-49dd-a98c-83e5b526fa19",
      "createTime": "2015-12-02T16:29:06.000Z",
      "deployed": true,
      "started": true,
      "state": "NORMAL",
      "vmwareTools": {
        "versionStatus": "NEED_UPGRADE",
        "runningStatus": "RUNNING",
        "apiVersion": 8394
      },
      "virtualHardware": {
        "version": "vmx-08",
        "upToDate": false
      },
      "id": "0919ebcd-7593-46f2-a708-7f5b8e8c7ca5",
      "datacenterId": "NA1"
    }
  ],
  "server": [
    {
      "name": "Production Web Server MCP 2",
      "description": "Server to host our main web application.",
      "operatingSystem": {
        "id": "WIN2008S32",
        "displayName": "WIN2008S/32",
        "family": "WINDOWS"
      },
      "cpu": {
        "count": 2,
        "speed": "STANDARD",
        "coresPerSocket": 1
      },
      "memoryGb": 4,
      "disk": [
        {
          "id": "c2e1f199-116e-4dbc-9960-68720b832b0a",
          "scsiId": 0,
          "sizeGb": 50,
          "speed": "STANDARD",
          "state": "NORMAL"
        }
      ],
      "networkInfo": {
        "primaryNic": {
          "id": "5e869800-df7b-4626-bcbf-8643b8be11fd",
          "privateIpv4": "10.0.4.8",
          "ipv6": "2607:f480:1111:1282:2960:fb72:7154:6160",
          "vlanId": "bc529e20-dc6f-42ba-be20-0ffe44d1993f",
          "vlanName": "Production VLAN",
          "state": "NORMAL"
        },
        "additionalNic": [],
        "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0"
      },
      "monitoring": {
        "monitoringId": "11039",
        "servicePlan": "ESSENTIALS",
        "state": "NORMAL"
      },
      "backup": {
        "assetId": "91002e08-8dc1-47a1-ad33-04f501c06f87",
        "servicePlan": "Advanced",
```

```
        "state": "NORMAL"
      },
      "softwareLabel": [
        "MSSQL2008R2S"
      ],
      "sourceImageId": "3ebf3c0f-90fe-4a8b-8585-6e65b316592c",
      "createTime": "2015-12-02T10:31:33.000Z",
      "deployed": true,
      "started": true,
      "state": "PENDING_CHANGE",
      "progress": {
        "action": "SHUTDOWN_SERVER",
        "requestTime": "2015-12-02T11:07:40.000Z",
        "userName": "devuser1"
      },
      "vmwareTools": {
        "versionStatus": "CURRENT",
        "runningStatus": "RUNNING",
        "apiVersion": 9354
      },
      "virtualHardware": {
        "version": "vmx-08",
        "upToDate": false
      },
      "id": "5a32d6e4-9707-4813-a269-56ab4d989f4d",
      "datacenterId": "NA9"
    }
  ],
  "pageNumber": 1,
  "pageCount": 2,
  "totalCount": 2,
  "pageSize": 250
}
```


13.2.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

13.3. Get Server

13.3.1. Description

Returns details of a single Server belonging to the organization identified by the {org-id} parameter.

Note that it is more efficient to use **Get Server** than **List Servers** (filtered by *id*) if you wish to retrieve *state* status or details of a single Server. This is particularly useful when monitoring the *state* of a Server to see if an operation is complete.

13.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/server/{server-id}`
Type	HTTP GET
Supported for	MCP 1.0 and MCP 2.0
Processing	Synchronous
Roles	Any role

13.3.3. Response Details

Refer to the XSD on the Community for full payload details. Note that the XSD is applicable to JSON integrators too, even if not being used programmatically. In both payload samples below, an MCP 2.0 Server payload is shown. See **List Servers** above for an MCP 1.0 Server example.

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<server id="5a32d6e4-9707-4813-a269-56ab4d989f4d" datacenterId="NA9">
  <name>Production Web Server</name>
  <description>Server to host our main web application.</description>
  <operatingSystem id="WIN2008S32" displayName="WIN2008S/32"
family="WINDOWS"/>
  <cpu count="2" speed="STANDARD" coresPerSocket="1"/>
  <memoryGb>4</memoryGb>
  <disk id="c2e1f199-116e-4dbc-9960-68720b832b0a" scsiId="0" sizeGb="50"
speed="STANDARD" state="NORMAL"/>
  <networkInfo networkDomainId="553f26b6-2a73-42c3-a78b-6116f11291d0">
    <primaryNic id="5e869800-df7b-4626-bcbf-8643b8be11fd"
privateIpv4="10.0.4.8" ipv6="2607:f480:1111:1282:2960:fb72:7154:6160"
vlanId="bc529e20-dc6f-42ba-be20-0ffe44d1993f" vlanName="Production VLAN"
state="NORMAL"/>
  </networkInfo>
  <backup assetId="5579f3a7-4c32-4cf5-8a7e-b45c36a35c10"
servicePlan="Enterprise" state="NORMAL"/>
  <monitoring monitoringId="11049" servicePlan="ESSENTIALS" state="NORMAL"/>
  <softwareLabel>MSSQL2008R2S</softwareLabel>
  <sourceImageId>3ebf3c0f-90fe-4a8b-8585-6e65b316592c</sourceImageId>
  <createTime>2015-12-02T10:31:33.000Z</createTime>
  <deployed>true</deployed>
  <started>true</started>
  <state>PENDING_CHANGE</state>
  <progress>
    <action>SHUTDOWN_SERVER</action>
    <requestTime>2015-12-02T11:07:40.000Z</requestTime>
    <userName>devuser1</userName>
  </progress>
  <vmwareTools versionStatus="CURRENT" runningStatus="RUNNING"
apiVersion="9354"/>
  <virtualHardware version="vmx-08" upToDate="false"/>
</server>
```

JSON

```
{
  "server": [
```

```
{
  "name": "Production Web Server",
  "description": "Server to host our main web application.",
  "operatingSystem": {
    "id": "WIN2008S32",
    "displayName": "WIN2008S/32",
    "family": "WINDOWS"
  },
  "cpu": {
    "count": 2,
    "speed": "STANDARD",
    "coresPerSocket": 1
  },
  "memoryGb": 4,
  "disk": [
    {
      "id": "c2e1f199-116e-4dbc-9960-68720b832b0a",
      "scsiId": 0,
      "sizeGb": 50,
      "speed": "STANDARD",
      "state": "NORMAL"
    }
  ],
  "networkInfo": {
    "primaryNic": {
      "id": "5e869800-df7b-4626-bcbf-8643b8be11fd",
      "privateIpv4": "10.0.4.8",
      "ipv6": "2607:f480:1111:1282:2960:fb72:7154:6160",
      "vlanId": "bc529e20-dc6f-42ba-be20-0ffe44d1993f",
      "vlanName": "Production VLAN",
      "state": "NORMAL"
    },
    "additionalNic": [],
    "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0"
  },
  "backup": {
    "assetId": "91002e08-8dc1-47a1-ad33-04f501c06f87",
    "servicePlan": "Advanced",
    "state": "NORMAL"
  },
  "monitoring": {
    "monitoringId": "11049",
    "servicePlan": "ESSENTIALS",
    "state": "NORMAL"
  },
  "softwareLabel": [
    "MSSQL2008R2S"
  ],
  "sourceImageId": "3ebf3c0f-90fe-4a8b-8585-6e65b316592c",
  "createTime": "2015-12-02T10:31:33.000Z",
  "deployed": true,
  "started": true,
  "state": "PENDING_CHANGE",
  "progress": {
    "action": "SHUTDOWN_SERVER",
    "requestTime": "2015-12-02T11:07:40.000Z",
    "userName": "devuser1"
  },
  "vmwareTools": {
    "versionStatus": "CURRENT",
    "runningStatus": "RUNNING",
    "apiVersion": 9354
  },
  "virtualHardware": {
    "version": "vmx-08",
    "upToDate": false
  },
  "id": "5a32d6e4-9707-4813-a269-56ab4d989f4d",
  "datacenterId": "NA9"
```

```
    }
  ],
  "pageNumber": 1,
  "pageCount": 1,
  "totalCount": 1,
  "pageSize": 250
}
```

13.3.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Cannot find Server {server-id}. *The value passed for {server-id} does not correspond to a valid VLAN belonging to {org-id}. See **List Servers**.*

13.4. Delete Server

13.4.1. Description

Deletes a Server belonging to the organization identified by {org-id}.

13.4.2. Request Details

URL	https://<Cloud API URL>/caas/2.1/{org-id}/server/deleteServer
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<deleteServer id="d577a691-e116-4913-a440-022d2729fc84"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "5b00a2ab-c665-4cd6-8291-0b931374fb3d"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Server to be deleted.

13.4.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-03-
08T05:49:07.774-04:00/f0a53414-e991-4c9b-b52f-5770ff5b606b">
    <operation>DELETE_SERVER</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to Delete Server (Id:d577a691-e116-4913-a440-
022d2729fc84) has been accepted and is being processed</message>
</response>
```

JSON

```
{
    "operation": "DELETE_SERVER",
    "responseCode": "IN_PROGRESS",
    "message": "Request to Delete Server (Id:5b00a2ab-c665-4cd6-8291-
0b931374fb3d) has been accepted and is being processed",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-08T05:47:02.731-04:00/22303753-7e22-4741-9e80-
9da944a1f116"
}
```

13.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
HAS_DEPENDENCY	Server <id> is associated with a Real Server (MCP 1.0). Server <id> is associated with an Anti-Affinity Rule (MCP 1.0). Server <id> cannot be deleted because it is enabled for Backup. Server <id> cannot be deleted because it is enabled for Monitoring.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
INVALID_INPUT_DATA	Server <id> must be provided. *Server <id> is a required field but has been omitted from the request.*
RESOURCE_BUSY	Another operation is in progress on Server id <id>. Please try again later.
RESOURCE_LOCKED	Server <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Cannot find Server <id>. *The value passed for <id> does not correspond to a valid Server belonging to {org-id}. See **List Servers**.*
SERVER_STARTED	It is not possible to delete a Server until it has been shutdown or powered off.

13.5. Start Server

13.5.1. Description

Starts a non-running Server belonging to the organization identified by {org-id}. Requires the identifier {id} of the Server.

13.5.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/startServer`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<startServer id="d577a691-e116-4913-a440-022d2729fc84"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "5b00a2ab-c665-4cd6-8291-0b931374fb3d"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Server to be started.

13.5.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-08-
12T08:27:00.176-04:00/14b0a61f-7e85-49c9-8e7e-146fa3e562b7">
    <operation>START_SERVER</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to start Server 'Production Server' has been accepted and
is being processed.</message>
</response>
```

JSON

```
{
  "operation": "START_SERVER",
  "responseCode": "IN_PROGRESS",
  "message": "Request to start Server 'Production Server' has been accepted
and is being processed.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-08-12T08:40:02.133-04:00/58230e75-e6c9-4e9f-948c-
d3b1f0cce126"
}
```

13.5.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	*id* must be provided. *Server id is a required field but has been omitted from the request.*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
RESOURCE_BUSY	Another operation is in progress on Server id <id>. Please try again later.
RESOURCE_LOCKED	Server <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Server <id> not found. *The value passed for <id> does not correspond to a valid Server belonging to {org-id}. See **List Servers**.*
SERVER_STARTED	It is not possible to start a Server if it is already started. *For status refer to **Get Server**.*

13.6. Shutdown Server

13.6.1. Description

This function performs guest OS initiated shutdown of a running Server belonging to the organization identified by {org-id}. Requires the identifier {id} of the Server.

Note that this is generally preferable to **Power Off Server**, which is equivalent to cutting the power supply to a physical machine.

13.6.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/shutdownServer`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<shutdownServer id="d577a691-e116-4913-a440-022d2729fc84"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "5b00a2ab-c665-4cd6-8291-0b931374fb3d"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Server to be shutdown.

13.6.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-08-
12T09:06:01.949-04:00/fd2650a9-3473-41eb-9c6e-1eef29c0bb27">
    <operation>SHUTDOWN_SERVER</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to shutdown Server 'Production Server' has been accepted
and is being processed.</message>
</response>
```

JSON

```
{
  "operation": "SHUTDOWN_SERVER",
  "responseCode": "IN_PROGRESS",
  "message": "Request to shutdown Server 'Production Server' has been accepted
and is being processed.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-08-12T08:59:43.729-04:00/c75435a0-e1a8-4e56-8a56-
0864160821b4"
}
```

13.6.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	*id* must be provided. *Server id is a required field but has been omitted from the request.*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
RESOURCE_BUSY	Another operation is in progress on Server id <id>. Please try again later.
RESOURCE_LOCKED	Server <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Server <id> not found. *The value passed for <id> does not correspond to a valid Server belonging to {org-id}. See **List Servers**.*
SERVER_STOPPED	It is not possible to shutdown a Server if it is already stopped. *For status refer to **Get Server**.*
VMWARE_TOOLS_INVALID_STATUS	VMware Tools is not running on the Server. *For status refer to **Get Server**.*

13.7. Reboot Server

13.7.1. Description

This function performs guest OS initiated restart of a running Server belonging to the organization identified by {org-id}. Requires the identifier {id} of the Server.

Note that this is generally preferable to **Reset Server**, which is equivalent to cutting the power supply to a physical machine and restarting it.

13.7.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/rebootServer`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<rebootServer id="d577a691-e116-4913-a440-022d2729fc84"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "5b00a2ab-c665-4cd6-8291-0b931374fb3d"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Server to be shutdown.

13.7.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-08-
12T09:52:43.365-04:00/5260a4e5-ea21-49f4-909a-22341d8c39cb">
    <operation>REBOOT_SERVER</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to reboot Server 'Production Server' has been accepted
and is being processed.</message>
</response>
```

JSON

```
{
  "operation": "REBOOT_SERVER",
  "responseCode": "IN_PROGRESS",
  "message": "Request to reboot Server 'Production Server' has been accepted
and is being processed.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-08-12T10:00:53.218-04:00/a1b44b97-4466-40b3-bd67-
70db4964033a"
}
```

13.7.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	*id* must be provided. *Server id is a required field but has been omitted from the request.*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
RESOURCE_BUSY	Another operation is in progress on Server id <id>. Please try again later.
RESOURCE_LOCKED	Server <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Server <id> not found. *The value passed for <id> does not correspond to a valid Server belonging to {org-id}. See **List Servers**.*
SERVER_STOPPED	It is not possible to reboot a Server if it is already stopped. *For status refer to **Get Server**.*
VMWARE_TOOLS_INVALID_STATUS	VMware Tools is not running on the Server. *For status refer to **Get Server**.*

13.8. Reset Server

13.8.1. Description

This function performs hard reset of a running Server belonging to the organization identified by {org-id}. Requires the identifier {id} of the Server.

This is an *aggressive* function; equivalent to cutting the power supply to a physical machine and then restarting it.

13.8.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/resetServer`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<resetServer id="d577a691-e116-4913-a440-022d2729fc84"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "5b00a2ab-c665-4cd6-8291-0b931374fb3d"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Server to be reset.

13.8.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-08-
12T11:37:13.823-04:00/7b264a73-d73b-424c-bec8-5debba8b4626">
    <operation>RESET_SERVER</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to reset Server 'Production Server' has been accepted and
is being processed.</message>
</response>
```

JSON

```
{
  "operation": "RESET_SERVER",
  "responseCode": "IN_PROGRESS",
  "message": "Request to reset Server 'Production Server' has been accepted
and is being processed.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-08-12T11:45:02.400-04:00/efdb8f23-79d6-4e35-8f3b-
2e7495065c0d"
}
```

13.8.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	*id* must be provided. *Server id is a required field but has been omitted from the request.*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
RESOURCE_BUSY	Another operation is in progress on Server id <id>. Please try again later.
RESOURCE_LOCKED	Server <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Server <id> not found. *The value passed for <id> does not correspond to a valid Server belonging to {org-id}. See **List Servers**.*
SERVER_STOPPED	It is not possible to reset a Server if it is already stopped. *For status refer to **Get Server**.*

13.9. Power Off Server

13.9.1. Description

This function performs hard power off of a running Server belonging to the organization identified by {org-id}. Requires the identifier {id} of the Server.

This is an *aggressive* function; equivalent to cutting the power supply to a physical machine.

13.9.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/powerOffServer`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<powerOffServer id="d577a691-e116-4913-a440-022d2729fc84"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "5b00a2ab-c665-4cd6-8291-0b931374fb3d"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Server to be powered off.

13.9.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-08-
12T11:49:17.375-04:00/d5bb0975-1ade-4350-aaec-24807bdf7038">
    <operation>POWER_OFF_SERVER</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to power off Server 'Production Server' has been accepted
and is being processed.</message>
</response>
```

JSON

```
{
  "operation": "POWER_OFF_SERVER",
  "responseCode": "IN_PROGRESS",
  "message": "Request to power off Server 'Production Server' has been
accepted and is being processed.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-08-12T11:45:02.400-04:00/efdb8f23-79d6-4e35-8f3b-
2e7495065c0d"
}
```

13.9.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	*id* must be provided. *Server id is a required field but has been omitted from the request.*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
RESOURCE_BUSY	Another operation is in progress on Server id <id>. Please try again later.
RESOURCE_LOCKED	Server <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Server <id> not found. *The value passed for <id> does not correspond to a valid Server belonging to {org-id}. See **List Servers**.*
SERVER_STOPPED	It is not possible to power off a Server if it is already stopped. *For status refer to **Get Server**.*

13.10. Reconfigure Server

13.10.1. Description

This function updates compute resource properties for a Server belonging to the organization identified by {org-id}.

Requires the identifier {id} of the Server and at least one of the updatable properties, which are individually optional.

13.10.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/reconfigureServer`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<reconfigureServer id="7389da66-7685-400a-81ca-665781076bb1"
xmlns="urn:didata.com:api:cloud:types">
  <memoryGb>8</memoryGb>
  <cpuCount>5</cpuCount>
  <cpuSpeed>HIGHPERFORMANCE</cpuSpeed>
  <coresPerSocket>1</coresPerSocket>
</reconfigureServer>
```

JSON Request Sample

```
{
  "memoryGb": 8,
  "cpuCount": 5,
  "cpuSpeed": "STANDARD",
  "coresPerSocket": 1,
  "id": "f8fe7965-3b7c-4cee-827e-f1e0b40a72e0"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	The UUID identifying the Server to be reconfigured.
memoryGb	No	Integer.
cpuCount	No	Integer. Note the relationship between this value and the *coresPerSocket* value. See **List Data Centers** for the *MAX_CPU_COUNT* property which definies the maximum allowed for this element.
coresPerSocket	No	Integer. Must be an integer factor of *cpuCount*. The default and recommended value for cores per socket is 1. Cores per socket is an advanced setting, which can adversely affect performance if adjusted without understanding what it is doing. Please refer to the following article: VMware vSphere Blog: Does corespersocket Affect Performance?
cpuSpeed	No	String. See **List Data Centers** for the list of valid values for the data center. This value applies to all of the CPUs on the Server.

13.10.3. Response Details

XML

```
<response xmlns="urn:didata.com:api:cloud:types" requestId="na9/2015-11-
30T12:07:27.838-05:00/1e22a049-d772-4c39-943c-5c773a7ac9ae">
  <operation>RECONFIGURE_SERVER</operation>
  <responseCode>OK</responseCode>
  <message>Request to reconfigure Server 'Production Server' has been accepted
and is being processed.</message>
</response>
```

JSON

```
{
  "operation": "RECONFIGURE_SERVER",
  "responseCode": "OK",
  "message": "Request to reconfigure Server 'Production Server' has been
accepted and is being processed.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "na9/2015-11-30T12:08:58.987-05:00/cf11d8e6-447f-4021-bcd7-
bf5278a99072"
}
```

13.10.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	memoryGb (*memoryGb*) is not within the allowed range for this data center (*minMemoryGb* to *maxMemoryGb*). See **List Data Centers**. cpuCount (*cpuCount*) exceeds maximum allowed value (*maxCpuCount*) supported by the Data Center *datacenterId*. See **List Data Centers**.
CPU_SPEED_NOT_AVAILABLE	cpuSpeed (*cpuSpeed*) is not available in Data Center *datacenterId*. See **List Data Centers**.
INVALID_INPUT_DATA	*id* must be provided. *Server id is a required field but has been omitted from the request.* At least one of *memoryGb*, *cpuCount*, *cpuSpeed*, *coresPerSocket* must be supplied. <*field name*>, if supplied, must be a positive integer. cpuSpeed (*cpuSpeed*) is not a valid CPU Speed. If only one of cpuCount, coresPerSocket is specified then current value is used for validation, for example: • Requested coresPerSocket (2) must be an integer factor of current cpuCount (5). • Current coresPerSocket (2) must be an integer factor of requested cpuCount (5).

INFRASTRUCTURE_IN_MAINTENANCE	Data center *datacenterId* hypervisor is in a maintenance state.
RESOURCE_BUSY	Another operation is in progress on Server id *id*. Please try again later.
RESOURCE_LOCKED	Server *id* is locked. Please contact support.
RESOURCE_NOT_FOUND	*id id* not found. *The value passed for id does not correspond to a valid Server belonging to {org-id}. See* ***List Servers****.*
VIRTUAL_HARDWARE_INVALID	memoryGb (*memoryGb*) exceeds maximum allowed value (<*maximum supported by Virtual Hardware version*>) supported by the Server's Virtual Hardware version. cpuCount (*cpuCount*) requires the Server to have Virtual Hardware version <*minimum compatible Virtual Hardware version*> or greater. Requested coresPerSocket (*codesPerSocket*) is not supported by the Server's Virtual Hardware version.

13.11. Update VMware Tools

13.11.1. Description

This function triggers an update of the VMware Tools software running on the guest OS of a Server belonging to the organization identified by {org-id}. Requires the identifier {id} of the Server.

Notes:

- VMware Tools will only be updated if it is reported as out of date by VMware.
- The process of updating VMware Tools requires that the Server is rebooted. Ensure that the Server is in a safe state to do so prior to using this function.

13.11.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/updateVmwareTools`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<updateVmwareTools id="d577a691-e116-4913-a440-022d2729fc84"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "5b00a2ab-c665-4cd6-8291-0b931374fb3d"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Server upon which VMware Tools requires an update.

13.11.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-08-
12T11:49:17.375-04:00/d5bb0975-1ade-4350-aaec-24807bdf7038">
    <operation>UPDATE_VMWARE_TOOLS</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to update VMware Tools on Server 'Production Server' has
been accepted and is being processed.</message>
</response>
```

JSON

```
{
  "operation": "UPDATE_VMWARE_TOOLS",
  "responseCode": "IN_PROGRESS",
  "message": "Request to update VMware Tools on Server 'Production Server' has
been accepted and is being processed.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-08-12T11:45:02.400-04:00/efdb8f23-79d6-4e35-8f3b-
```

```
2e7495065c0d"
}
```

13.11.4.Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	*id* must be provided. *Server id is a required field but has been omitted from the request.*
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
NO_CHANGE	VMware Tools version is already current.
RESOURCE_BUSY	Another operation is in progress on Server id <id>. Please try again later.
RESOURCE_LOCKED	Server <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	*id* <id> not found. *The value passed for <id> does not correspond to a valid Server belonging to {org-id}. See **List Servers**.*
VMWARE_TOOLS_INVALID_STATUS	VMware Tools is not installed. VMware Tools on this server could not be upgraded. VMware Tools is in unmanaged state. VMware Tools cannot be upgraded because the version status is <status>.

13.12. Upgrade Server Virtual Hardware

13.12.1. Description

This function triggers an update of the VMware Virtual Hardware running on the guest OS of a Server belonging to the organization identified by {org-id}.

Requires the identifier {id} of the Server.

Exercise caution prior to upgrading Virtual Hardware: VMware recommend cloning the Server prior to performing the upgrade in case something goes wrong during the upgrade process.

The **List Data Centers** function response includes the version, deemed up to date for the data center in which the Server is deployed. Note that when you upgrade Virtual Hardware, that the version received can be *higher* than the benchmark version for the Data Center location where the Cloud Server resides. This is because an upgrade guarantees *at least* the version specified for the Data Center.

13.12.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/upgradeVirtualHardware`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<upgradeVirtualHardware id="d577a691-e116-4913-a440-022d2729fc84"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
  "id": "d577a691-e116-4913-a440-022d2729fc84"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the Server upon which Virtual Hardware is indicated as out of date versus the value on the Data Center in which the server is deployed. See **List Data Centers**.

13.12.3. Response Details

XML

```
<response xmlns="urn:didata.com:api:cloud:types" requestId="na9/2015-11-
30T11:29:24.004-05:00/56210aa5-3c6d-48fd-b4e8-aa8916f7e900">
  <operation>UPGRADE_VIRTUAL_HARDWARE</operation>
  <responseCode>IN_PROGRESS</responseCode>
  <message>Request to upgrade Virtual Hardware on Server 'Production Server'
has been accepted and is being processed.</message>
</response>
```

JSON

```
{
  "operation": "UPGRADE_VIRTUAL_HARDWARE",
  "responseCode": "IN_PROGRESS",
  "message": "Request to upgrade Virtual Hardware on Server 'Production
Server' has been accepted and is being processed.",
  "info": [],
```

```
  "warning": [],
  "error": [],
  "requestId": "na9/2015-11-19T11:45:02.400-04:00/efdb8f23-79d6-4e35-8f3b-
2e7495065c0d"
}
```

13.12.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Data center *datacenterId* hypervisor is in a maintenance state.
INVALID_INPUT_DATA	*id* must be provided. *Server id is a required field but has been omitted from the request.*
NO_CHANGE	Virtual Hardware version is already up to date.
RESOURCE_BUSY	Another operation is in progress on Server id *id*. Please try again later.
RESOURCE_LOCKED	Server *id* is locked. Please contact support.
RESOURCE_NOT_FOUND	id *id* not found. *The value passed for id does not correspond to a valid Server belonging to {org-id}. See* ***List Servers****.*
SERVER_STARTED	Server must be in a stopped state in order to upgrade the Virtual Hardware version.
VIRTUAL_HARDWARE_INVALID	Upgrading a Server's Virtual Hardware version from <*current version*> is not supported.
VMWARE_TOOLS_INVALID_STATUS	VMware Tools version is not current on the server. VMware tools must be current before upgrading the Virtual Hardware. Please update VMware Tools and try again. VMware Tools version cannot be determined on the server. VMware tools must be current before upgrading the Virtual Hardware. Please update VMware Tools and try again. VMware Tools is not installed. VMware tools must be current before upgrading the Virtual Hardware. Please install VMware Tools and try again.

13.13. Add NIC

13.13.1. Description

Adds an additional NIC to a Server belonging to the organization identified by {org-id}.

Requires the UUID of the Server and either; the UUID of the target VLAN to be added as a new NIC **or** a valid RFC 1918 private IPv4 address to be added to the Server as a new NIC.

Note that the server must not be running to execute this function.

13.13.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/addNic`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<addNic xmlns="urn:didata.com:api:cloud:types">
 <serverId>1c7762ca-f379-4eef-b08e-aa526d602589</serverId>
 <nic>
   <vlanId>2e312054-532a-46aa-ab4f-226660bfba6d</vlanId> or
   <privateIpv4>10.0.3.18</privateIpv4>
 </nic>
</addNic>
```

JSON Request Sample

```
{
 "serverId": "1c7762ca-f379-4eef-b08e-aa526d602589",
 "nic":
  {
    "vlanId": "2e312054-532a-46aa-ab4f-226660bfba6d" or
    "privateIpv4": "10.0.3.18"
  }
}
```

Request Properties

Field	Required	Type and Constraints
serverId	Yes	Identifies the Server to which the NIC will be added.
nic.vlanId	No*	UUID of the VLAN to be added as a new NIC to the Server.
nic.privateIpv4	No*	Private IPv4 address to be added as a new NIC to the Server. privateIpv4 must be an RFC1918 dot-decimal IPv4 address, and must not be in the system reserved space for the VLAN. The system reserved IP addresses are the first six (.0 - .5), eighth (.7) and broadcast address (the last .255 in the VLAN). For example: “10.0.3.18”.

*One of *vlanId* or *privateIpv4* must be provided.

13.13.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-03-
```

```
08T06:17:52.133-04:00/70e7458c-038c-4819-a1de-90d15007c2f5">
    <operation>ADD_NIC</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>The request to add NIC for VLAN 'Subsystem VLAN' on Server
'Production Mail Server' has been accepted and is being processed..</message>
    <info name="nicId" value="a202e51b-41c0-4cfc-add0-b1c62fc0ecf6"/>
</response>
```

JSON

```
{
    "operation": "ADD_NIC",
    "responseCode": "IN_PROGRESS",
    "message": "The request to add NIC for VLAN 'Subsystem VLAN' on Server
'Production Mail Server' has been accepted and is being processed",
    "info": [
        {
            "name": "nicId",
            "value": "5999db1d-725c-46ba-9d4e-d33991e61ab1"
        }
    ],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-08T05:47:02.731-04:00/22303753-7e22-4741-9e80-
9da944a1f116"
}
```

13.13.4.Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	The Server already has the maximum number of additional NICs allowed in a <hypervisor.type> data center (<maximum number of additional NICs allowed>). See **List Data Centers.**
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
INVALID_INPUT_DATA	Server <id> must be provided. Either privateIpv4 or vlanId must be provided for the nic field. privateIpv4 <ip> must be a valid IPv4 address. Server <id> already has a NIC on the VLAN <id>.
IP_ADDRESS_NOT_UNIQUE	privateIpv4 <ip> is already reserved in the VLAN <id>.
IP_ADDRESS_OUT_OF_RANGE	privateIpv4 <privateIpv4> is outside the permitted range (<VLAN specific>.8 to <VLAN specific>.254). *See note above about system reserved space.*
OPERATION_NOT_SUPPORTED	Server with id <id> is deployed in a data center <datacenterId>, which is not an MCP 2.0 data center.
RESOURCE_BUSY	Another operation is in progress on Server id <id>. Please try again later.
RESOURCE_LOCKED	Server <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Cannot find Server <id>. See **List Servers**. VLAN with id <id> cannot be found on Network Domain with id <network domain id>. See **List VLANs** and **List Network Domains**.
SERVER_STARTED	It is not possible to add a NIC until the Server has been shutdown or powered off.

13.14. Remove NIC

13.14.1. Description

Removes an additional NIC from a Server belonging to the organization identified by {org-id}.

Requires the UUID of the NIC to be removed.

Note that the server must not be running to execute this function.

13.14.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/removeNic`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<removeNic xmlns="urn:didata.com:api:cloud:types" id="5999db1d-725c-46ba-9d4e-
d33991e61ab1" />
```

JSON Request Sample

```
{
    "id": "5999db1d-725c-46ba-9d4e-d33991e61ab1"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	Identifies the NIC to be removed.

13.14.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-03-
08T06:57:51.297-04:00/6857b7e6-e3a9-438c-87fe-ed2248c65c55">
    <operation>REMOVE_NIC</operation>
    <responseCode>IN_PROGRESS</responseCode>
    <message>Request to Remove NIC 5999db1d-725c-46ba-9d4e-d33991e61ab1 for
VLAN 'Subsystem VLAN' from Server 'Production Mail Server' has been accepted
and is being processed.</message>
</response>
```

JSON

```
{
    "operation": "REMOVE_NIC",
    "responseCode": "IN_PROGRESS",
    "message": "Request to Remove NIC 5999db1d-725c-46ba-9d4e-d33991e61ab1 for
VLAN 'Subsystem VLAN' from Server 'Production Mail Server' has been accepted
and is being processed.",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-08T06:53:59.331-04:00/75b54fae-dedb-49ce-aa7f-
494c73fc45e5"
}
```

13.14.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	Cannot remove NIC with id <id> as it is the Primary NIC of the Server.
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
INVALID_INPUT_DATA	<id> must be provided.
RESOURCE_BUSY	Another operation is in progress on Server id <id>. Please try again later.
RESOURCE_LOCKED	Server <id> is locked. Please contact support.
RESOURCE_NOT_FOUND	Cannot find Server <id>. See **List Servers**. VLAN with id <id> cannot be found on Network Domain with id <network domain id>. See **List VLANs** and **List Network Domains**.
SERVER_STARTED	It is not possible to remove a NIC until the Server has been shutdown or powered off.

13.15. Notify System of NIC IP Address Change

13.15.1. Description

On occasion manual intervention on a deployed Server or a API 0.9 **Restore Backup for Server (Out of Place)** operation can require that an IP address changes from the one assigned when it was originally deployed to a different IP address within the same network range. This API function updates the Cloud record to match the value set on the deployed Server.

Manual network intervention on a Server and this API function should be used with extreme care because it is possible to disrupt network traffic to the Server or cause an IP conflict with another resource on your network. Note for Servers deployed on Hyper-V based Private Cloud data centers, Hyper-V requires the server to be stopped.

13.15.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/notifyNicIpChange`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator", "server" and "backup"

XML Request Sample

```
<notifyNicIpChange xmlns="urn:didata.com:api:cloud:types" nicId="5999db1d-
725c-46ba-9d4e-d33991e61ab1">
    <privateIpv4>10.0.1.5</privateIpv4>
    <ipv6>fdfe::5a55:caff:fefa::1:9089</ipv6>
</notifyNicIpChange>
```

JSON Request Sample

```
{
    "nicId": "5999db1d-725c-46ba-9d4e-d33991e61ab1",
    "privateIpv4": "10.0.1.5",
    "ipv6": "fdfe::5a55:caff:fefa::1:9089"
}
```

Request Properties

Field	Required	Type and Constraints
nicId	Yes	Identifies the NIC to be updated.
privateIpv4	No*	IPv4 address to replace the IPv4 address currently associated with the NIC and must not be in the system reserved space for the VLAN. The system reserved IP addresses are the first six (.0 - .5), eighth (.7) and broadcast address (the last .255 in the VLAN).
ipv6	No*	IPv6 address to replace the IPv6 address currently associated with the NIC.

*One of *privateIpv4* or *ipv6* must be supplied.

13.15.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response requestId="NA9/2015-03-08T06:53:59.331-04:00/75b54fae-dedb-49ce-
aa7f-494c73fc45e5"
 xmlns="urn:didata.com:api:cloud:types">
  <operation>NOTIFY_NIC_IP_ADDRESS_CHANGE</operation>
  <responseCode>IN_PROGRESS</responseCode>
```

```
  <message>Request to notify system of IP change for NIC 5999db1d-725c-46ba-
9d4e-d33991e61ab1 is being processed</message>
</response>
```

JSON

```
{
    "operation": "NOTIFY_NIC_IP_ADDRESS_CHANGE",
    "responseCode": "IN_PROGRESS",
    "message": "Request to notify system of IP change for NIC 5999db1d-725c-
46ba-9d4e-d33991e61ab1 is being processed",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-08T06:53:59.331-04:00/75b54fae-dedb-49ce-aa7f-
494c73fc45e5"
}
```

13.15.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	MCP 1.0 data centers do not support IPv6.
INCOMPATIBLE_DEPENDENCY	Server <serverId> is associated with a Real Server (MCP 1.0).
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
INVALID_INPUT_DATA	<id> must be provided.
IP_ADDRESS_NOT_UNIQUE	privateIpv4 <privateIpv4> is already reserved in the VLAN <vlanId>. ipv6 <ipv6> is already reserved in the VLAN <vlanId>.
IP_ADDRESS_OUT_OF_RANGE	privateIpv4 <privateIpv4> is outside the permitted range (<VLAN specific>.8 to <VLAN specific>.254). ipv6 <ipv6> is outside the permitted range (x:x:x:x:[0:0:0:20-FFFF:FFFF:FFFF:FFFF]).
RESOURCE_BUSY	Another operation in progress on NIC with id <nicId>. Please try again later. Operation in progress on Server with id <serverId>. Please try again later.
RESOURCE_LOCKED	NIC with id <nicId> is locked. Please contact support. Server with id <serverId> is locked. Please contact support.
RESOURCE_NOT_FOUND	NIC <nicId> not found. See **List Servers**.
SERVER_STARTED	It is not possible to update a NIC IP address until the Server has been shutdown or powered off in a "HYPERV" hypervisor type data center. See **List Data Centers.**

13.16. Clean Failed Server Deployment

13.16.1. Description

This function is used to remove a failed Server Deployment. Failed Server deployments are visible as Servers in the **List Servers** response with the value "FAILED_ADD" for their *state* property.

If the attempt to remove the failed server with the Clean Failed Server Deployment function fails, you will need to contact Support to remove the server from the list and free up the associated private IP address.

13.16.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/cleanServer`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Asynchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<cleanServer xmlns="urn:didata.com:api:cloud:types" id="1c7762ca-f379-4eef-
b08e-aa526d602589" />
```

JSON Request Sample

```
{
    "id": "1c7762ca-f379-4eef-b08e-aa526d602589"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	UUID for the Server to be cleaned.

13.16.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response requestId="NA9/2015-03-08T06:53:59.331-04:00/75b54fae-dedb-49ce-
aa7f-494c73fc45e5"
 xmlns="urn:didata.com:api:cloud:types">
  <operation>CLEAN_SERVER</operation>
  <responseCode>IN_PROGRESS</responseCode>
  <message>Request to Clean Server '1c7762ca-f379-4eef-b08e-aa526d602589' is
being processed.</message>
</response>
```

JSON

```
{
    "operation": "CLEAN_SERVER",
    "responseCode": "IN_PROGRESS",
    "message": "Request to Clean Server '1c7762ca-f379-4eef-b08e-aa526d602589'
is being processed.",
    "info": [],
    "warning": [],
    "error": [],
    "requestId": "NA9/2015-03-08T06:53:59.331-04:00/75b54fae-dedb-49ce-aa7f-
494c73fc45e5"
}
```

13.16.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state.
OPERATION_NOT_SUPPORTED	This function can only be used against Servers with a FAILED_ADD *state* property.
RESOURCE_BUSY	Operation in progress on Server with id <serverId>. Please try again later.
RESOURCE_LOCKED	Server with id <serverId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Server <id> not found. See **List Servers**.

13.17. List Server Anti-Affinity Rules

13.17.1. Description

Please refer to the 0.9 Cloud API for API functions to Create and Delete Server Anti-Affinity Rules.

Lists the Server Anti-Affinity rules between Servers on a Network Domain belonging to the organization identified by the {org-id} parameter.

This function supports MCP 1.0 and MCP 2.0 data centers. As a result, three of the available filters are listed as required but these are mutually exclusive. One of the required filters must be specified for a given request. See the *Filter Name and Filter Parameters* table for details.

13.17.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/antiAffinityRule?`
	
	`Filter Required Parameters. One of the following must be provided:`
	`  networkDomainId= or`
	`  serverId= or`
	`  networkId=`
	
	`Filter Optional Parameters:`
	`  [&id=]`
	`  [&state=]`
	`  [&createTime=]`
	
	`Paging/Ordering Optional Parameters:`
	`  [&pageSize=]`
	`  [&pageNumber=]`
	`  [&orderBy=]`

Type	HTTP GET
Supported for	MCP 1.0 and MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
networkDomainId	uuid (String)	**Required**. **MCP 2.0.** Exclusive of *serverId* and *networkId*. Filters by Network Domain. *networkDomainId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
serverId	uuid (String)	**Required**. **MCP 2.0 and MCP 1.0.** Exclusive of networkDomain*Id* and *networkId*. Filters by Server. *serverId=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	No
networkId	uuid (String)	**Required**. **MCP 1.0.** Exclusive of *networkDomainId* and	Yes	No

		serverId. Filters by Cloud Network. *networkId=9a857b7c-37bd-11e2-a91c-0030487e0302*		
id	uuid (String)	Identifies an individual Server Anti-Affinity Rule. *id=9a857b7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
state	String	Identifies Server Anti-Affinity Rules by their *state*. Case insensitive. The initial possible set of values for *state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.	Yes	Yes
createTime	String	Identifies the date of creation of Server Anti-Affinity Rules. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes

13.17.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<antiAffinityRules
    xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="1"
totalCount="1" pageSize="250">
    <antiAffinityRule id="d4ebfdd1-ec03-45c7-b0be-fbcc0861e9bf" state="NORMAL"
created="2015-06-05T14:44:54.000Z" datacenterId="NA9">
        <serverSummary id="681a6db2-9c7c-4d98-a0c4-7b3d7c1619ba">
            <name>Production Server 1</name>
            <description></description>
            <networkingDetails>
                <networkInfo networkDomainId="553f26b6-2a73-42c3-a78b-
6116f11291d0" networkDomainName="Production Network Domain">
                    <primaryNic id="a6a16a86-7e5b-4138-8c94-3c09f5195a98"
privateIpv4="10.0.0.13" ipv6="2607:f480:1111:1348:5909:96d3:29f5:5e4d"
vlanId="a6d3e2d7-0092-4f87-b00c-f276127bc26d" vlanName="Main VLAN"/>
                    <additionalNic id="1e35ea03-a8e5-4771-abc0-8269a45c5735"
privateIpv4="10.0.3.12" ipv6="2607:f480:1111:1351:73e4:7d49:93f1:31a3"
vlanId="8f19ad6d-ebbb-4393-9ef3-060e5f0b0618" vlanName="Secondary VLAN"/>
                </networkInfo>
            </networkingDetails>
        </serverSummary>
        <serverSummary id="5783e93f-5370-44fc-a772-cd3c29a2ecaa">
            <name>Production Server 2</name>
            <description>this is a
```

```
multiline description
so it is</description>
            <networkingDetails>
                <networkInfo networkDomainId="553f26b6-2a73-42c3-a78b-
6116f11291d0" networkDomainName="Production Network Domain">
                    <primaryNic id="b3b9261e-50d8-4919-bbe6-866b65b223a1"
privateIpv4="10.0.3.13" ipv6="2607:f480:1111:1351:5793:ebd8:d01c:53f3"
vlanId="8f19ad6d-ebbb-4393-9ef3-060e5f0b0618" vlanName="Production VLAN"/>
                    <additionalNic id="c9975e8c-77e8-4c0c-8aa1-17df726c37cc"
privateIpv4="10.0.2.13" ipv6="2607:f480:1111:1350:c7f:bef6:bc89:f0e4"
vlanId="5404a06d-a084-4f15-a18a-9dfea006d00c" vlanName="Bus VLAN"/>
                </networkInfo>
            </networkingDetails>
        </serverSummary>
    </antiAffinityRule>
</antiAffinityRules>
```

JSON

```
{
  "antiAffinityRule": [
    {
      "serverSummary": [
        {
          "name": "Production Server 1",
          "description": "",
          "networkingDetails": {
            "networkInfo": {
              "primaryNic": {
                "id": "a6a16a86-7e5b-4138-8c94-3c09f5195a98",
                "privateIpv4": "10.0.0.13",
                "ipv6": "2607:f480:1111:1348:5909:96d3:29f5:5e4d",
                "vlanId": "a6d3e2d7-0092-4f87-b00c-f276127bc26d",
                "vlanName": "Main VLAN"
              },
              "additionalNic": [
                {
                  "id": "1e35ea03-a8e5-4771-abc0-8269a45c5735",
                  "privateIpv4": "10.0.3.12",
                  "ipv6": "2607:f480:1111:1351:73e4:7d49:93f1:31a3",
                  "vlanId": "8f19ad6d-ebbb-4393-9ef3-060e5f0b0618",
                  "vlanName": "Secondary VLAN"
                }
              ],
              "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
              "networkDomainName": "Production Network Domani"
            }
          },
          "id": "681a6db2-9c7c-4d98-a0c4-7b3d7c1619ba"
        },
        {
          "name": "Production Server 2",
          "description": "",
          "networkingDetails": {
            "networkInfo": {
              "primaryNic": {
                "id": "b3b9261e-50d8-4919-bbe6-866b65b223a1",
                "privateIpv4": "10.0.3.13",
                "ipv6": "2607:f480:1111:1351:5793:ebd8:d01c:53f3",
                "vlanId": "8f19ad6d-ebbb-4393-9ef3-060e5f0b0618",
                "vlanName": "Production VLAN"
              },
              "additionalNic": [
                {
                  "id": "c9975e8c-77e8-4c0c-8aa1-17df726c37cc",
                  "privateIpv4": "10.0.2.13",
                  "ipv6": "2607:f480:1111:1350:c7f:bef6:bc89:f0e4",
                  "vlanId": "5404a06d-a084-4f15-a18a-9dfea006d00c",
```

```
                    "vlanName": "Bus VLAN"
                  }
                ],
                "networkDomainId": "553f26b6-2a73-42c3-a78b-6116f11291d0",
                "networkDomainName": "Production Network Domain"
              }
            },
            "id": "5783e93f-5370-44fc-a772-cd3c29a2ecaa"
          }
        ],
        "id": "d4ebfdd1-ec03-45c7-b0be-fbcc0861e9bf",
        "state": "NORMAL",
        "created": "2015-06-05T14:44:54.000Z",
        "datacenterId": "NA9"
      }
    ],
    "pageNumber": 1,
    "pageCount": 1,
    "totalCount": 1,
    "pageSize": 250
}
```


13.17.4.Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	One of the following filters must be provided: networkDomainId, networkId or serverId.

14. Server API – Cloud Monitoring

14.1. Enable Monitoring for a Server

14.1.1. Description

This function is used to enable Monitoring for a Server on a particular Service Plan.

14.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/enableServerMonitoring`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Synchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<enableServerMonitoring id="5783e93f-5370-44fc-a772-cd3c29a2ecaa"
xmlns="urn:didata.com:api:cloud:types">
    <servicePlan>ESSENTIALS</servicePlan>
</enableServerMonitoring>
```

JSON Request Sample

```
{
    "id": "5783e93f-5370-44fc-a772-cd3c29a2ecaa",
    "servicePlan": "ESSENTIALS"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	UUID for the Server to have monitoring enabled on the *servicePlan* provided.
servicePlan	Yes	ESSENTIALS and ADVANCED are the supported values for *servicePlan*.

14.1.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-06-
10T09:22:40.684-04:00/c0ca4a5d-3611-417e-8209-fd3046cff1d5">
    <operation>ENABLE_SERVER_MONITORING</operation>
    <responseCode>OK</responseCode>
    <message>Monitoring on Server 'Production Server' has been
enabled.</message>
</response>
```

JSON

```
{
  "operation": "ENABLE_SERVER_MONITORING",
  "responseCode": "OK",
  "message": "Monitoring on Server 'Production Server' has been enabled.",
  "info": [],
```

```
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-06-10T09:21:16.414-04:00/100f5361-5823-47e3-b22e-
be78bd012c69"
}
```

14.1.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state. Cloud Monitoring is in a maintenance state.
INVALID_INPUT_DATA	id must be provided. servicePlan must be provided.
NO_CHANGE	Monitoring with service plan <servicePlan> is already enabled on '<server name>'.
OPERATION_NOT_SUPPORTED	Cloud Monitoring is not enabled in this datacenter <id>.
RESOURCE_BUSY	Another operation is in progress on Server with id <id> . Please try again later. We are currently enabling Monitoring for this Organization as a result of another Enable Server Monitoring operation. Please try again later.
RESOURCE_LOCKED	Server with id <serverId> is locked. Please contact support.
RESOURCE_NOT_FOUND	Server <id> not found. See **List Servers**.

14.2. Change Server Monitoring Service Plan

14.2.1. Description

This function is used to change the Monitoring Service Plan for a Server that has Monitoring already enabled.

14.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/changeServerMonitoringPlan`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Synchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<changeServerMonitoringPlan id="5783e93f-5370-44fc-a772-cd3c29a2ecaa"
xmlns="urn:didata.com:api:cloud:types">
    <servicePlan>ADVANCED</servicePlan>
</changeServerMonitoringPlan>
```

JSON Request Sample

```
{
    "id": "5783e93f-5370-44fc-a772-cd3c29a2ecaa",
    "servicePlan": "ADVANCED"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	UUID for the Server to have monitoring enabled on the *servicePlan* provided.
servicePlan	Yes	ESSENTIALS and ADVANCED are the supported values for *servicePlan*.

14.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-08-
10T09:22:40.684-04:00/c0ca4a5d-3611-417e-8209-fd3046cff1d5">
    <operation>CHANGE_SERVER_MONITORING_PLAN</operation>
    <responseCode>OK</responseCode>
    <message>Monitoring on Server 'Production Server' has been changed to
Service Plan 'ADVANCED'.</message>
</response>
```

JSON

```
{
  "operation": "CHANGE_SERVER_MONITORING_PLAN",
  "responseCode": "OK",
  "message": "Monitoring on Server 'Production Server' has been change to
Service Plan 'ADVANCED'.",
  "info": [],
  "warning": [],
```

```
  "error": [],
  "requestId": "NA9/2015-08-10T09:21:16.414-04:00/100f5361-5823-47e3-b22e-
be78bd012c69"
}
```

14.2.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state. Cloud Monitoring is in a maintenance state.
INVALID_INPUT_DATA	*id* must be provided. *servicePlan* must be provided.
NO_CHANGE	Monitoring with service plan <servicePlan> is already enabled on '<server name>'.
OPERATION_NOT_SUPPORTED	Server '<server name>' is not enabled for monitoring.
RESOURCE_NOT_FOUND	Server <id> not found. See **List Servers**.

14.3. Disable Monitoring for a Server

14.3.1. Description

This function is used to disable Monitoring for a Cloud Server.

14.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/server/disableServerMonitoring`
Type	HTTP POST
Supported for	MCP 1.0 and MCP 2.0
Processing	Synchronous. For status refer to **Get Server** and **List Servers**.
Roles	"primary administrator" and "server"

XML Request Sample

```
<disableServerMonitoring id="5783e93f-5370-44fc-a772-cd3c29a2ecaa"
xmlns="urn:didata.com:api:cloud:types"/>
```

JSON Request Sample

```
{
    "id": "5783e93f-5370-44fc-a772-cd3c29a2ecaa"
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	UUID for the Server to have monitoring enabled on the servicePlan provided.

14.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<response
    xmlns="urn:didata.com:api:cloud:types" requestId="NA9/2015-06-
10T09:19:48.692-04:00/929c91ac-5377-4dcf-aac6-a89610971139">
    <operation>DISABLE_SERVER_MONITORING</operation>
    <responseCode>OK</responseCode>
    <message>Monitoring on Server 5783e93f-5370-44fc-a772-cd3c29a2ecaa has
been disabled.</message>
</response>
```

JSON

```
{
  "operation": "DISABLE_SERVER_MONITORING",
  "responseCode": "OK",
  "message": "Monitoring on Server 'Production Server' has been disabled.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "NA9/2015-06-10T09:21:16.414-04:00/100f5361-5823-47e3-b22e-
be78bd012c69"
}
```

14.3.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INFRASTRUCTURE_IN_MAINTENANCE	Data center <datacenterId> hypervisor is in a maintenance state. Cloud Monitoring is in a maintenance state.
INVALID_INPUT_DATA	*id* must be provided.
NO_CHANGE	Monitoring is not enabled on '<server name>'.
OPERATION_NOT_SUPPORTED	Cloud Monitoring is not enabled in datacenter <id>.
RESOURCE_NOT_FOUND	Server <id> not found. See **List Servers**.

14.4. View Monitoring Usage Report

14.4.1. Description

This function returns the Monitoring Usage Report at the given geographic region for the organization identified by {org-id}, limited (inclusively) to the supplied date range input parameters. The report is returned in Comma Separated Value (CSV) format.

A maximum of 31 days of data will be returned. If *endDate* exceeds 31 days from *startDate*, the range will be limited to 31 days. If *endDate* is not supplied the report will be limited to 31 days.

If executed where Monitoring is not available for the organization {org-id}, an empty report will be returned.

14.4.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/report/usageMonitoring?`
	
	`Filter Required Parameters:`
	`  startDate=`
	
	`Filter Optional Parameters:`
	`  [&endDate=]`

Type	HTTP GET
Supported for	n/a
Processing	Synchronous.
Roles	"primary administrator" and "report"

Request Properties

Field	Required	Type and Constraints
startDate	Yes	The inclusive start date for the range of days to be included in the report. In the form YYYY-MM-DD.
endDate	No	The inclusive end date for the range of days to be included in the report. In the form YYYY-MM-DD.

14.4.3. Response Details

CSV

```
Date,Location,Server,Server ID,Essentials Monitoring Hours,Advanced Monitoring
Hours
2/24/15,NA5,Server A,5a2378ba-11de-11e5-ad2a-ca9798323470,14.25,0
2/24/15,NA5,Server B,6330a644-11de-11e5-ad2a-ca9798323470,0,8.64
2/25/15,NA5,Server A,5a2378ba-11de-11e5-ad2a-ca9798323470,24,0
2/25/15,NA5,Server B,6330a644-11de-11e5-ad2a-ca9798323470,0,24
2/26/15,NA5,Server A,5a2378ba-11de-11e5-ad2a-ca9798323470,17.59,6.41
2/26/15,NA5,Server B,6330a644-11de-11e5-ad2a-ca9798323470,2.11,21.89
```

14.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
INVALID_INPUT_DATA	startDate is required. endDate (<endDate>) must be later than startDate (<startDate>). startDate (<startDate>) cannot be the current day or in the future.

15. Server Image API

15.1. List OS Images

15.1.1. Description

By default this API function lists all of the OS Images available at the geographic region to the organization identified by {org-id}.

Most integrating users will find it userful to filter **at least** by Cloud Data Center (*datacenterId)* on every request because it is at that level the OS Images are applicable for Cloud Server deployment.

Various filter values can be used to narrow the list returned as detailed below, including the ability to filter based on the Operating System (*operatingSystemId* and/or *operatingSystemFamily*), which is of frequent use for Cloud Server Deployment.

Note that it is more efficient to use **Get OS Image** than **List OS Images** (filtered by *id*) if you wish to retrieve the details of a single OS Image.

15.1.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/image/osImage[?]`
	
	`Filter Optional Parameters:`
	`[&id=]`
	`[&datacenterId=]`
	`[&name=]`
	`[&createTime=]`
	`[&state=]`
	`[&operatingSystemId=]`
	`[&operatingSystemFamily=]`
	
	`Paging/Ordering Optional Parameters:`
	`[&pageSize=]`
	`[&pageNumber=]`
	`[&orderBy=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual OS Image. Supports the LIKE operator. *id=j6ghb7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes
name	String	Identifies OS Images by name. Supports the LIKE operator.	Yes	Yes

		name=MyOSImage		
createTime	Date	Identifies the date of creation of OS Images. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes
state	String	Identifies OS Images by their *state*. For OS Images this is expected to be NORMAL in all but exceptional circumstances. Case insensitive. The initial possible set of values for *state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.	Yes	Yes
operatingSystemId	String	Identifies a single Operating System. *operatingSystemId=REDHAT664* See **List Operating Systems**.	Yes	Yes
operatingSystemFamily	String	Identifies a set of Operating Systems with a common family. See **List Operating Systems**. *operatingSystemFamily=WINDOWS*	Yes	Yes

15.1.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<osImages xmlns="urn:didata.com:api:cloud:types" pageNumber="1" pageCount="47"
totalCount="47" pageSize="250">
  <osImage id="c14b1a46-2428-44c1-9c1a-b20e6418d08c" datacenterId="NA9">
    <name>RedHat 6 64-bit 2 CPU</name>
    <description>RedHat 6.6 Enterprise (Santiago) 64-bit</description>
    <operatingSystem id="REDHAT664" displayName="REDHAT6/64" family="UNIX"/>
    <cpu count="2" speed="STANDARD" coresPerSocket="1"/>
    <memoryGb>4</memoryGb>
    <disk id="a02b7244-99d8-4889-84a5-5e4373c1bb26" scsiId="0" sizeGb="10"
speed="STANDARD"/>
    <createTime>2015-09-17T11:23:48.000Z</createTime>
    <osImageKey>T-RHEL-6-64-2-4-10</osImageKey>
  </osImage>
  <osImage id="6b4fb0c7-a57b-4f58-b59c-9958f94f971a" datacenterId="NA9">
    <name>Win2012 DC 2 CPU</name>
    <description>Windows 2012 Datacenter</description>
    <operatingSystem id="WIN2012DC64" displayName="WIN2012DC/64"
family="WINDOWS"/>
```

```
    <cpu count="2" speed="STANDARD" coresPerSocket="1"/>
    <memoryGb>4</memoryGb>
    <disk id="f5e01854-a211-4ec6-96d6-2753b6d47877" scsiId="0" sizeGb="50"
speed="STANDARD"/>
    <createTime>2015-09-17T11:44:43.000Z</createTime>
    <osImageKey>T-WIN-2012-DATACTR-64-2-4-50</osImageKey>
  </osImage>
  <osImage id="3ebf3c0f-90fe-4a8b-8585-6e65b316592c" datacenterId="NA9">
    <name>Win2008 Std 32-bit 2 CPU</name>
    <description>Windows 2008 Enterprise R2 32-bit installed with Microsoft
SQL Server 2012 Standard Edition</description>
    <operatingSystem id="WIN2008S32" displayName="WIN2008S/32"
family="WINDOWS"/>
    <cpu count="2" speed="STANDARD" coresPerSocket="1"/>
    <memoryGb>4</memoryGb>
    <disk id="6e5b5112-0eae-44eb-83cd-1bd0d58fbeab" scsiId="0" sizeGb="50"
speed="STANDARD"/>
    <softwareLabel>MSSQL2008R2S</softwareLabel>
    <createTime>2014-11-20T12:54:22.000Z</createTime>
    <osImageKey>T-WIN-2008-ENT-32-2-4-50</osImageKey>
  </osImage>
</osImages>
```

JSON

```
{
  "osImage": [
    {
      "name": "RedHat 6 64-bit 2 CPU",
      "description": "RedHat 6.6 Enterprise (Santiago) 64-bit",
      "operatingSystem": {
        "id": "REDHAT664",
        "displayName": "REDHAT6/64",
        "family": "UNIX"
      },
      "cpu": {
        "count": 2,
        "speed": "STANDARD",
        "coresPerSocket": 1
      },
      "memoryGb": 4,
      "disk": [
        {
          "id": "a02b7244-99d8-4889-84a5-5e4373c1bb26",
          "scsiId": 0,
          "sizeGb": 10,
          "speed": "STANDARD"
        }
      ],
      "softwareLabel": [],
      "createTime": "2015-09-17T11:23:48.000Z",
      "id": "c14b1a46-2428-44c1-9c1a-b20e6418d08c",
      "datacenterId": "NA9",
      "osImageKey": "T-RHEL-6-64-2-4-10"
    },
    {
      "name": "Win2012 DC 2 CPU",
      "description": "Windows 2012 Datacenter",
      "operatingSystem": {
        "id": "WIN2012DC64",
        "displayName": "WIN2012DC/64",
        "family": "WINDOWS"
      },
      "cpu": {
        "count": 2,
        "speed": "STANDARD",
        "coresPerSocket": 1
      },
```

```
      "memoryGb": 4,
      "disk": [
        {
          "id": "f5e01854-a211-4ec6-96d6-2753b6d47877",
          "scsiId": 0,
          "sizeGb": 50,
          "speed": "STANDARD"
        }
      ],
      "softwareLabel": [],
      "createTime": "2015-09-17T11:44:43.000Z",
      "id": "6b4fb0c7-a57b-4f58-b59c-9958f94f971a",
      "datacenterId": "NA9",
      "osImageKey": "T-WIN-2012-DATACTR-64-2-4-50"
    },
    {
      "name": "Win2008 Std 32-bit 2 CPU",
      "description": "Windows 2008 Enterprise R2 32-bit installed with
Microsoft SQL Server 2012 Standard Edition",
      "operatingSystem": {
        "id": "WIN2008S32",
        "displayName": "WIN2008S/32",
        "family": "WINDOWS"
      },
      "cpu": {
        "count": 2,
        "speed": "STANDARD",
        "coresPerSocket": 1
      },
      "memoryGb": 4,
      "disk": [
        {
          "id": "6e5b5112-0eae-44eb-83cd-1bd0d58fbeab",
          "scsiId": 0,
          "sizeGb": 50,
          "speed": "STANDARD"
        }
      ],
      "softwareLabel": [
        "MSSQL2008R2S"
      ],
      "createTime": "2014-11-20T12:54:22.000Z",
      "id": "3ebf3c0f-90fe-4a8b-8585-6e65b316592c",
      "datacenterId": "NA9",
      "osImageKey": "T-WIN-2008-ENT-32-2-4-50"
    }
  ],
  "pageNumber": 1,
  "pageCount": 3,
  "totalCount": 3,
  "pageSize": 250
}
```


15.1.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

15.2. Get OS Image

15.2.1. Description

Returns details of a single OS Image available to organization identified by {org-id}.

Note that it is more efficient to use **Get OS Image** than **List OS Images** (filtered by *id*) if you wish to retrieve the details of a single OS Image.

15.2.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/image/osImage/{image-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

15.2.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<osImage xmlns="urn:didata.com:api:cloud:types" id="3ebf3c0f-90fe-4a8b-8585-
6e65b316592c" datacenterId="NA9">
  <name>Win2008 Std 32-bit 2 CPU</name>
  <description>Windows 2008 Enterprise R2 32-bit installed with Microsoft SQL
Server 2012 Standard Edition</description>
  <operatingSystem id="WIN2008S32" displayName="WIN2008S/32"
family="WINDOWS"/>
  <cpu count="2" speed="STANDARD" coresPerSocket="1"/>
  <memoryGb>4</memoryGb>
  <disk id="6e5b5112-0eae-44eb-83cd-1bd0d58fbeab" scsiId="0" sizeGb="50"
speed="STANDARD"/>
  <softwareLabel>MSSQL2008R2S</softwareLabel>
  <createTime>2014-11-20T12:54:22.000Z</createTime>
  <osImageKey>T-WIN-2008-ENT-32-2-4-50</osImageKey>
</osImage>
```

JSON

```
{
  "name": "Win2008 Std 32-bit 2 CPU",
  "description": "Windows 2008 Enterprise R2 32-bit installed with Microsoft
SQL Server 2012 Standard Edition",
  "operatingSystem": {
    "id": "WIN2008S32",
    "displayName": "WIN2008S/32",
    "family": "WINDOWS"
  },
  "cpu": {
    "count": 2,
    "speed": "STANDARD",
    "coresPerSocket": 1
  },
  "memoryGb": 4,
  "disk": [
    {
      "id": "6e5b5112-0eae-44eb-83cd-1bd0d58fbeab",
      "scsiId": 0,
      "sizeGb": 50,
      "speed": "STANDARD"
    }
  ],
  "softwareLabel": [
    "MSSQL2008R2S"
  ],
```

```
  "createTime": "2014-11-20T12:54:22.000Z",
  "id": "3ebf3c0f-90fe-4a8b-8585-6e65b316592c",
  "datacenterId": "NA9",
  "osImageKey": "T-WIN-2008-ENT-32-2-4-50"
}
```


15.2.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Server Image {image-id} not found. *The value passed for {image-id} does not correspond to a valid OS Image available to {org-id}. See **List OS Images**.*

15.3. List Customer Images

15.3.1. Description

By default this API function lists all of the Customer Images at the geographic region owned by the organization identified by {org-id}.

Most integrating users will find it userful to filter **at least** by Cloud Data Center (*datacenterId)* on every request because it is at that level the Customer Images are applicable for Cloud Server deployment.

Various filter values can be used to narrow the list returned as detailed below, including the ability to filter based on the Operating System (*operatingSystemId* and/or *operatingSystemFamily*), which is of frequent use for Cloud Server Deployment.

Note that it is more efficient to use **Get Customer Image** than **List Customer Images** (filtered by *id*) if you wish to retrieve the status or details of a single Customer Image.

15.3.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/image/customerImage[?]`
	
	`Filter Optional Parameters:`
	`[&id=]`
	`[&datacenterId=]`
	`[&name=]`
	`[&createTime=]`
	`[&state=]`
	`[&operatingSystemId=]`
	`[&operatingSystemFamily=]`
	
	`Paging/Ordering Optional Parameters:`
	`[&pageSize=]`
	`[&pageNumber=]`
	`[&orderBy=]`

Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

Field Name and Filter Parameters

fieldName	Type	Description	Filter Parameter	orderBy Parameter
id	uuid (String)	Identifies an individual Customer Image. Supports the LIKE operator. *id=j6ghb7c-37bd-11e2-a91c-0030487e0302*	Yes	Yes
datacenterId	String	Identifies an individual Data Center. See **List Data Centers**. *datacenterId=NA9*	Yes	Yes
name	String	Identifies Customer Images by name. Supports the LIKE operator.	Yes	Yes

		name=MyCustomerImage		
createTime	Date	Identifies the date of creation of Customer Images. Supports MIN, MAX, LT and GT. Refer to samples in **Paging and Filtering for List API Functions.**	Yes	Yes
state	String	Identifies Customer Images by their *state*. This can be useful as part of checking progress status for Customer Images with operations in progress. Case insensitive. The initial possible set of values for *state* are: "NORMAL", "PENDING_ADD", "PENDING_CHANGE", "PENDING_DELETE", "FAILED_ADD", "FAILED_CHANGE", "FAILED_DELETE" and "REQUIRES_SUPPORT". This set of values should not be assumed to be static and can increase at any time.	Yes	Yes
operatingSystemId	String	Identifies a single Operating System. *operatingSystemId=REDHAT664* See **List Operating Systems**.	Yes	Yes
operatingSystemFamily	String	Identifies a set of Operating Systems with a common family. See **List Operating Systems**. *operatingSystemFamily=WINDOWS*	Yes	Yes

15.3.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<customerImages xmlns="urn:didata.com:api:cloud:types" pageNumber="1"
pageCount="35" totalCount="35" pageSize="250">
  <customerImage id="5234e5c7-01de-4411-8b6e-baeb8d91cf5d" datacenterId="NA9">
    <name>ImportedCustomerImage</name>
    <description/>
    <operatingSystem id="REDHAT664" displayName="REDHAT6/64" family="UNIX"/>
    <cpu count="4" speed="STANDARD" coresPerSocket="1"/>
    <memoryGb>2</memoryGb>
    <disk id="1a82316f-23ed-4fe9-b6d8-6b92ac467423" scsiId="0" sizeGb="12"
speed="STANDARD"/>
    <createTime>2015-11-19T14:29:02.000Z</createTime>
    <source type="IMPORT">
      <artifact type="MF" value="ImportedCustomerImage.mf" date="2015-11-
19T14:28:54.000Z"/>
      <artifact type="OVF" value="ImportedCustomerImage.ovf" date="2015-11-
19T14:28:05.000Z"/>
```

```
      <artifact type="VMDK" value="ImportedCustomerImage-disk1.vmdk"
date="2015-11-19T12:22:31.000Z"/>
    </source>
    <state>NORMAL</state>
    <vmwareTools versionStatus="NEED_UPGRADE" runningStatus="NOT_RUNNING"
apiVersion="8389"/>
    <virtualHardware version="vmx-10" upToDate="true"/>
  </customerImage>
  <customerImage id="2ffa36c8-1848-49eb-b4fa-9d908775f68c" datacenterId="NA9">
    <name>CustomerImageWithPricedSoftwareLabels</name>
    <description/>
    <operatingSystem id="WIN2008S32" displayName="WIN2008S/32"
family="WINDOWS"/>
    <cpu count="1" speed="STANDARD" coresPerSocket="1"/>
    <memoryGb>1</memoryGb>
    <disk id="29455efc-51af-4b4d-91b3-d81ca0dff7d8" scsiId="0" sizeGb="50"
speed="STANDARD"/>
    <softwareLabel>MSSQL2008R2S</softwareLabel>
    <createTime>2015-11-03T15:25:34.000Z</createTime>
    <source type="CLONE">
      <artifact type="SERVER_ID" value="7c9c2551-269d-4274-a247-
126ba7c6215c"/>
    </source>
    <state>NORMAL</state>
    <vmwareTools versionStatus="CURRENT" runningStatus="NOT_RUNNING"/>
    <virtualHardware version="vmx-08" upToDate="false"/>
  </customerImage>
  <customerImage id="1fc1844f-45d6-4364-b447-f7c7645b47de" datacenterId="NA9">
    <name>CopiedCustomerImage</name>
    <description/>
    <operatingSystem id="REDHAT664" displayName="REDHAT6/64" family="UNIX"/>
    <cpu count="1" speed="STANDARD" coresPerSocket="1"/>
    <memoryGb>2</memoryGb>
    <disk id="42b20819-c161-4dec-aa94-73ec370a6e37" scsiId="0" sizeGb="10"
speed="STANDARD"/>
    <createTime>2015-11-11T17:17:00.000Z</createTime>
    <source type="COPY">
      <artifact type="IMAGE_ID" value="0b8357b6-f156-4b27-b4fd-b81d09c15efc"/>
    </source>
    <state>NORMAL</state>
    <vmwareTools versionStatus="NEED_UPGRADE" runningStatus="NOT_RUNNING"
apiVersion="9355"/>
    <virtualHardware version="vmx-10" upToDate="true"/>
  </customerImage>
</customerImages>
```

JSON

```
{
  "customerImage": [
    {
      "name": "CustomerImageWithPricedSoftwareLabels",
      "description": "",
      "operatingSystem": {
        "id": "WIN2008S32",
        "displayName": "WIN2008S/32",
        "family": "WINDOWS"
      },
      "cpu": {
        "count": 1,
        "speed": "STANDARD",
        "coresPerSocket": 1
      },
      "memoryGb": 1,
      "disk": [
        {
          "id": "29455efc-51af-4b4d-91b3-d81ca0dff7d8",
          "scsiId": 0,
```

```
          "sizeGb": 50,
          "speed": "STANDARD"
        }
      ],
      "softwareLabel": [
        "MSSQL2008R2S"
      ],
      "createTime": "2015-11-03T15:25:34.000Z",
      "id": "2ffa36c8-1848-49eb-b4fa-9d908775f68c",
      "datacenterId": "NA9",
      "source": {
        "artifact": [
          {
            "type": "SERVER_ID",
            "value": "7c9c2551-269d-4274-a247-126ba7c6215c"
          }
        ],
        "type": "CLONE"
      },
      "state": "NORMAL",
      "vmwareTools": {
        "versionStatus": "CURRENT",
        "runningStatus": "NOT_RUNNING"
      },
      "virtualHardware": {
        "version": "vmx-08",
        "upToDate": false
      }
    },
    {
      "name": "CopiedCustomerImage",
      "description": "",
      "operatingSystem": {
        "id": "REDHAT664",
        "displayName": "REDHAT6/64",
        "family": "UNIX"
      },
      "cpu": {
        "count": 1,
        "speed": "STANDARD",
        "coresPerSocket": 1
      },
      "memoryGb": 2,
      "disk": [
        {
          "id": "42b20819-c161-4dec-aa94-73ec370a6e37",
          "scsiId": 0,
          "sizeGb": 10,
          "speed": "STANDARD"
        }
      ],
      "softwareLabel": [],
      "createTime": "2015-11-11T17:17:00.000Z",
      "id": "1fc1844f-45d6-4364-b447-f7c7645b47de",
      "datacenterId": "NA9",
      "source": {
        "artifact": [
          {
            "type": "IMAGE_ID",
            "value": "0b8357b6-f156-4b27-b4fd-b81d09c15efc"
          }
        ],
        "type": "COPY"
      },
      "state": "NORMAL",
      "vmwareTools": {
        "versionStatus": "NEED_UPGRADE",
        "runningStatus": "NOT_RUNNING",
        "apiVersion": 9355
```

```
      },
      "virtualHardware": {
        "version": "vmx-10",
        "upToDate": true
      }
    },
    {
      "name": "ImportedCustomerImage",
      "description": "",
      "operatingSystem": {
        "id": "REDHAT664",
        "displayName": "REDHAT6/64",
        "family": "UNIX"
      },
      "cpu": {
        "count": 4,
        "speed": "STANDARD",
        "coresPerSocket": 1
      },
      "memoryGb": 2,
      "disk": [
        {
          "id": "1a82316f-23ed-4fe9-b6d8-6b92ac467423",
          "scsiId": 0,
          "sizeGb": 12,
          "speed": "STANDARD"
        }
      ],
      "softwareLabel": [],
      "createTime": "2015-11-19T14:29:02.000Z",
      "id": "5234e5c7-01de-4411-8b6e-baeb8d91cf5d",
      "datacenterId": "NA9",
      "source": {
        "artifact": [
          {
            "type": "MF",
            "value": "ImportedCustomerImage.mf",
            "date": "2015-11-19T14:28:54.000Z"
          },
          {
            "type": "OVF",
            "value": "ImportedCustomerImage.ovf",
            "date": "2015-11-19T14:28:05.000Z"
          },
          {
            "type": "VMDK",
            "value": "ImportedCustomerImage-disk1.vmdk",
            "date": "2015-11-19T12:22:31.000Z"
          }
        ],
        "type": "IMPORT"
      },
      "state": "NORMAL",
      "vmwareTools": {
        "versionStatus": "NEED_UPGRADE",
        "runningStatus": "NOT_RUNNING",
        "apiVersion": 8389
      },
      "virtualHardware": {
        "version": "vmx-10",
        "upToDate": true
      }
    }
  ],
  "pageNumber": 1,
  "pageCount": 3,
  "totalCount": 3,
  "pageSize": 250
}
```

15.3.4. Response Codes

See ***Common Response Codes.*** This function returns no additional response codes.

15.4. Get Customer Image

15.4.1. Description

Returns details of a single Customer Image available to organization identified by {org-id}.

Note that it is more efficient to use **Get Customer Image** than **List Customer Images** (filtered by *id*) if you wish to retrieve the details of a single Customer Image. This API function is particularly useful if you want to monitor an operation in progress on a Customer Image.

15.4.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/image/customerImage/{image-id}`
Type	HTTP GET
Supported for	MCP 2.0
Processing	Synchronous
Roles	Any role

15.4.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<customerImage xmlns="urn:didata.com:api:cloud:types" id="2ffa36c8-1848-49eb-
b4fa-9d908775f68c" datacenterId="NA9">
  <name>CustomerImageWithPricedSoftwareLabels</name>
  <description/>
  <operatingSystem id="WIN2008S32" displayName="WIN2008S/32"
family="WINDOWS"/>
  <cpu count="1" speed="STANDARD" coresPerSocket="1"/>
  <memoryGb>1</memoryGb>
  <disk id="29455efc-51af-4b4d-91b3-d81ca0dff7d8" scsiId="0" sizeGb="50"
speed="STANDARD"/>
  <softwareLabel>MSSQL2008R2S</softwareLabel>
  <createTime>2015-11-03T15:25:34.000Z</createTime>
  <source type="CLONE">
    <artifact type="SERVER_ID" value="7c9c2551-269d-4274-a247-126ba7c6215c"/>
  </source>
  <state>NORMAL</state>
  <vmwareTools versionStatus="CURRENT" runningStatus="NOT_RUNNING"/>
  <virtualHardware version="vmx-08" upToDate="false"/>
</customerImage>
```

JSON

```
{
  "name": "CustomerImageWithPricedSoftwareLabels",
  "description": "",
  "operatingSystem": {
    "id": "WIN2008S32",
    "displayName": "WIN2008S/32",
    "family": "WINDOWS"
  },
  "cpu": {
    "count": 1,
    "speed": "STANDARD",
    "coresPerSocket": 1
  },
  "memoryGb": 1,
  "disk": [
    {
      "id": "29455efc-51af-4b4d-91b3-d81ca0dff7d8",
      "scsiId": 0,
      "sizeGb": 50,
      "speed": "STANDARD"
```

```
    }
  ],
  "softwareLabel": [
    "MSSQL2008R2S"
  ],
  "createTime": "2015-11-03T15:25:34.000Z",
  "id": "2ffa36c8-1848-49eb-b4fa-9d908775f68c",
  "datacenterId": "NA9",
  "source": {
    "artifact": [
      {
        "type": "SERVER_ID",
        "value": "7c9c2551-269d-4274-a247-126ba7c6215c"
      }
    ],
    "type": "CLONE"
  },
  "state": "NORMAL",
  "vmwareTools": {
    "versionStatus": "CURRENT",
    "runningStatus": "NOT_RUNNING"
  },
  "virtualHardware": {
    "version": "vmx-08",
    "upToDate": false
  }
}
```


15.4.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
RESOURCE_NOT_FOUND	Server Image {image-id} not found. *The value passed for {image-id} does not correspond to a valid Customer Image available to {org-id}. See **List Customer Images**.*

15.5. Edit Customer Image Metadata

15.5.1. Description

This API function allows an integrator to update the metadata associated with one of the Customer Images owned by the organization identified by {org-id}.

The metadata set is currently comprised of the following:

- Description
- CPU Speed
- Operating System ID
- Disk Speeds

15.5.2. Request Details

URL	`https://<Cloud API URL>/caas/2.1/{org-id}/image/editImageMetadata`
Type	HTTP POST
Supported for	MCP 2.0
Processing	Synchronous
Roles	"primary administrator" and "create image"

XML Request Sample

```
<imageMetadata id="2ffa36c8-1848-49eb-b4fa-9d908775f68c"
xmlns="urn:didata.com:api:cloud:types">
    <description>This is a priced software label Customer Image</description>
    <cpuSpeed>PERFORMANCE</cpuSpeed>
    <operatingSystemId>WIN2008S64</operatingSystemId>
    <disk scsiId="0" speed="HIGHPERFORMANCE"/>
</imageMetadata>
```

JSON Request Sample

```
{
  "id": "2ffa36c8-1848-49eb-b4fa-9d908775f68c",
  "description": "This is a priced software label Customer Image.",
  "operatingSystemId": "WIN2008S64",
  "disk": {
    "scsiId": "0",
    "speed": "STANDARD"
  }
}
```

Request Properties

Field	Required	Type and Constraints
id	Yes	The “id” of a Customer Image belonging to {org-id}.
description	No	Maximum 255 characters. Nillable.
cpuSpeed	No	See **List Data Centers** for available CPU Speed values.
operatingSystemId	No	IMPORTANT: Only update this value in conjunction with input from a support representative. See **List Operating Systems** for the list of available operatingSystemId values at the Data Center.
disk	No	*scsiId* and *speed* are required for any **disk** element included. There can be 0-14 disk elements depending on the number of disks on the Customer Image. See **Get Customer Image** or **List Customer Images** to retrieve

		the list of storage disks on the Customer Image, which includes their *scsiId* identifiers.
scsiId	Yes*	The value specified by *speed* will be set on the disk identified by *scsiId*.
speed	Yes*	See **List Data Centers** for available CPU Speed values.

15.5.3. Response Details

XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<response xmlns="urn:didata.com:api:cloud:types" requestId="na9/2015-11-
29T07:28:51.892-05:00/38202950-d1ea-408a-939b-359cb410f0bd">
  <operation>EDIT_IMAGE_METADATA</operation>
  <responseCode>OK</responseCode>
  <message>Server Image 'CustomerImageWithPricedSoftwareLabels' was edited
successfully.</message>
</response>
```

JSON

```
{
  "operation": "EDIT_IMAGE_METADATA",
  "responseCode": "OK",
  "message": "Server Image CustomerImageWithPricedSoftwareLabelswas edited
successfully.",
  "info": [],
  "warning": [],
  "error": [],
  "requestId": "na9/2015-11-29T07:41:54.142-05:00/6661799b-8276-481c-80df-
6d5bd55ba925"
}
```

15.5.4. Response Codes

See ***Common Response Codes.***

The table below defines additional response codes, all are returned with a **HTTP 400 (Error)** response:

Response Code	Message
CONFIGURATION_NOT_SUPPORTED	*operatingSystemId* is not supported in Data Center *datacenterId*. *See **List Data Centers**.*
CPU_SPEED_NOT_AVAILABLE	CPU Speed *cpuSpeed* is not available in Data Center *datacenterId*. *See **List Data Centers**.*
DISK_SPEED_NOT_AVAILABLE	Disk Speed *speed* is not currently available in Data Center *datacenterId. See **List Data Centers**.*
INFRASTRUCTURE_IN_MAINTENANCE	Datacenter <datacenterId> networking is in a maintenance state.
INVALID_INPUT_DATA	*id* must be provided. At least one of description, *cpuSpeed*, *operatingSystemId* and *disk* must be supplied. Server Image *description* must not be more than 255 characters in length. If you are requesting disk updates both *scsiId* and *speed* must be supplied. Cannot find disk(s) with scsiId <list of *scsiId*>. You have supplied two or more disk elements with the same scsiId *scsiId*.

	Disk Speed *speed* is not a valid Disk Speed. *See* ***List Data Centers****.* *operatingSystemId* is not a valid Operating System Id. *See* ***List Data Centers****.* *operatingSystemId* is not a <Operating System Family of Customer Image> Operating System Id. *operatingSystemId* must belong to the same family of operating systems. *See* ***List Data Centers****.*
RESOURCE_BUSY	Another operation is in progress on Server Image with id *id*. Please try again later.
RESOURCE_LOCKED	Server Image with id *id* is locked. Please contact support.
RESOURCE_NOT_FOUND	Server Image *id* not found. *The value passed for id does not correspond to a valid Customer Image belonging to {org-id}. See* ***List Customer Images****.*

16. Older Documentation Revisions

None.