API Reference Guide
for Public Cloud
Densify 3.0
April 2025

Contents

Introduction

The Densify API is the foundation for the next generation of capacity intelligence for your cloud infrastructure. The Densify API connects to your other infrastructure software components such as cloud stacks and provides rich capacity information and recommended optimization actions. With the Densify API, these systems become aware of the overall capacity and can make better automatic choices with a fully-informed model of the infrastructure.

The Densify API is a REST-based web service that is designed for the following:

- to connect to your cloud platforms and collect cloud infrastructure data;

- perform optimization **analysis**[1] on the collected data;

- provide **recommendations**[2] to reduce risk and cost, for automated execution or downstream processing;

New Features

New features and functionality are added in each Densify release. See *API New Features* (Help Topic ID 340760) in the Densify Online help to see what's new in each version.

---

[1]Collected data is analyzed using Densify's patented analytics and results are loaded into the reporting database for viewing through various reports, consoles, dashboards, and user interfaces.
[2]One or more actions for moving or resizing cloud or VM instances that will result in improved peformance and/or reduced cost.

RESTful API

The Densify API has been implemented as a RESTful web service. Requests and responses are built around the transfer of representations of resources.

The Densify RESTful web API is a web API implemented using HTTP. It is a collection of resources, with three defined aspects:

- the base URI for the web API, such as `http://localhost:8086/api/v2/`.

- the Internet media type of the data supported by the web API. The Densify API supports JSON.

- the set of operations supported by the web API using HTTP methods (e.g. `GET`, `PUT`, `POST`, or `DELETE`).

The following table shows the HTTP methods used to implement the web API.

Table: RESTful Web API HTTP Methods

Resource	GET	PUT	POST	DELETE
Collection URI, such as `http://localhost:8086/api/v2/<resource>/`	List the URIs and perhaps other details of the collection's members.	Not used.	Create a new entry in the collection. The new entry's URI is assigned automatically and is usually returned by the operation.	Not used.
Element URI, such as `http://localhost:8086/api/v2/<resource>/<item>`	Retrieve a representation of the addressed member of the collection, expressed in an appropriate Internet media type.	Modify the addressed member.	Not generally used, but is used to modify attributes and multiple objects within a given collection.	Delete the addressed member of the collection.

Using the API

Authentication

Authentication Methods

Densify supports only token-based authentication for securing API requests. Densify employs the JSON Web Token (JWT) standard. You need to obtain a signed token using an `/authorize` request, which can then be used for authenticating subsequent Densify API calls. See Authorize on page 62 for details.

API-enabled User Credentials

To access the API, you need Densify API-enabled user credentials. An API-enabled user must have the SaaS_User role assigned.

Configuration Settings

The following configuration settings can affect your API requests.

Table: API and Report Settings

Setting	Description	Default Value
API Detailed Logging (key=rest.api.logging)	Enables additional logging of API requests to the Densify web server log file, /var/opt/CRBAcat/logs/web-service.log.	OFF
API Page Size (key=rest.api.paging.pageSize)	The number of objects returned when	100

Setting	Description	Default Value
	performing a `GET` request on a collection of objects. To override the page size, use the following syntax (see Paging on page 21 for details): `GET /<resource object>/?page=<pagenumber> [&page_size=<pagesize>]`	
API Key Rotation (Days) (key=rest.api.key.rotation)	Specifies the number of days the Densify API key is rotated.	30
API Token Expiring (Minutes) (key=rest.api.token.expiry)	Specifies the JWT API token expiry time, in minutes. Configuring a short expiry time is recommended.	5
Enable Password Policy (key=password.policy.enabled)	Enables the password policy and enforces password character, length and lockout policies. See *Managing Configuration Settings> Password Policy* (Help Topic ID 111250). If this setting is set to TRUE, Densify API user accounts are subjected to the Password policy rules on login attempts, expiry date, and inactivity lockout. **Note:** *The Densify API does not enforce Character and Length Password policies during authentication.*	FALSE

Contact Support@Densify.com for details on changing any of these properties.

Note: *Densify API elements, parameters, and filters are case-sensitive.*

Logging

You can view the requests and the owner who performed the requests in the web server log file (`%CIRBA_HOME%\logs\cirba-webserver.log`). The log file also includes POST request details such as creation and state changes. This information is logged independently of configuration setting, API Detailed Logging, below.

To enable detailed logging, specify the configuration setting API Detailed Logging to "ON":

```
rest.api.logging=ON
```

Contact Support@Densify.com for details on changing this property.

Common Elements

To be consistent across the Densify API, there are some components that are common across all the resources.

ID, Name and Self Reference (id, name, href)

Every instance of an individual persistent resource is referable via its own URI. Thus each resource instance has a unique ID with a corresponding self link element that any integration code can store to refer back to that element.

In general, each resource object and any reference to a resource object use the following three elements to identify the object:

- `id`—UUID of the resource
- `name`—name of the resource (if applicable)
- `href`—link to the resource

Every collection request returns the above elements of each object in the collection.

> **Note:** Densify API elements, parameters, and filters are case-sensitive.

Date-Time

Date-time elements are specified in Coordinated Universal Time (UTC) format, using the UNIX epoch time in milliseconds.

Common Operations

Summary

When performing a `GET` request, you can qualify the request using one or more operations described in this section, e.g. specifying a sort order or a filter.

This following table summarizes which operations can be combined with other operations. For example, the Filters operation can be combined with Paging, Collection Details and Sort By, as shown by the Filter row.

For example, if you want the detailed list of all AWS Systems, listed in alphabetical order by name, you would do the following:

Request:

```
GET /systems/?platform=AWS&sort_by=name
```

or without trailing '/':

```
GET /systems?platform=AWS&sort_by=name
```

Table: Operations Summary

Operation	Filters	Paging	Collection Details	Sort By
Filters		✓	✓	✓
Paging	✓		✓	✓
Collection Details	✓	✓		✓
Sort By	✓	✓	✓	

Filters

Most collections can be filtered based on the elements of the resource objects within the collection. The elements that support filtering are identified in the **Filter** column (marked "F") of each Resource Elements table within this reference. If the filter finds no matching objects, an empty list is returned. If matching objects are found, a list of objects (in the form ID, Name and Self Reference (id, name, href) on page 19 for each object) is returned. Note that for date/time elements, only a range is supported as described below. To define an exact match for a date/time element, specify the exact match value using the `_from` and `_to` tags.

To filter the list of resource objects, simply provide a query string after the resource object of interest, where both `<element>` and `<value>` are case insensitive (with no quotes specified):

```
GET /<resource object>/?<element>=<value>
```

For example, if you wanted to filter the list of Systems that are of type `host` you would do the following:

Example: Filtering a List for Host Systems

Request:

```
GET /systems/?type=host
```

Response:

```
[
  {
    "id": "844608bd-2e6a-4201-a9f2-edb4b84389bc",
    "name": "esx-host-82",
    "href": "/systems/844608bd-2e6a-4201-a9f2-edb4b84389bc",
    "resource_id": "ac-59d7-339",
    "type": "host",
    "platform_model": "PowerEdge M610",
    ...
```

More complex filtering is supported using any combination of the following:

■ **Multiple Criterion**–specifying "`&`" to logically AND criteria.
   `/<resource object>/?<element1>=<value1>[&<element2>=<value2>]…`

■ **Multiple Values**–specifying "`,`" between values to logically OR a list of possible values for an element.
   `/<resource object>/?<element>=<value1>[,<value2>]…`

■ **Ranges**–specifying "`_from`" and "`_to`" appended to the end of the element name to specify from/to values, respectively. All elements of type number (including date/time elements) support a range.

The values must be valid values (e.g. valid UTC values for date/time types); otherwise, a wrong format error is returned.

- **Multi-Valued Elements**—specifying at least one value of the multi-valued element.

- **Special Character Support**—for filter values with characters that have special meaning in the query string, use the following URL encoding characters instead:

  - "%2B" for "+"

  - "%20" or "+" for space

  For example, to filter cluster name "eastus+test" and platform category "External Cloud", use:

  ```
  GET /systems/?infrastucture_group=eastus%2Btest&platform_
  category=External+Cloud
  ```

- **Name_Like**—use `?name_like="<substring>"` in your collection request. The '%' character can be used to match zero or more characters.

Note: *Densify API elements, parameters, and filters are case-sensitive.*

Paging

When performing a `GET` request on a collection of objects, a page of objects can be returned instead of the entire collection. To retrieve a specific page, supply the page number (integer >= 0, where 0 is the first page) and the page size, as follows:

```
GET /<resource object>/?page=<pagenumber>[&page_size=<pagesize>]
```

If the page size is not specified, then the page size will be taken from configuration setting API Page Size (parameter key `rest.api.paging.pageSize`).

For example, suppose you have 125 Systems and you want to see the last 25 (with a page size of 100):

```
GET /systems/?page=1
```

Note: *If the collection changes in between page requests, the items are shifted appropriately.*

Collection Details

Most collection `GET` requests support a `details` option, which returns the details of every resource object in the collection:

```
GET /<resource object>/?details=true
```

The default is `?details=false`, if not specified. This option is supported for most collections.

Individual Details

Every individual `GET` request uses an `id` or `name` to identify the resource object instance to retrieve (where `name` is used only if the resource object does not have an `id`). The individual `GET` request returns all the elements of the identified or named resource object.

```
GET /<resource object>/<id>
GET /<resource object>/<name>
```

When `name` is used, the search is case insensitive and returns all matching instances.

Sort By

Some collection requests support a `sort_by` option. This option returns the collection of objects in ascending or descending order by the element specified:

```
GET /<resource object>/?sort_by=<element>[,asc|desc]
```

The elements that support sorting are identified in the **Sort By** column (marked "S") of each Resource Elements table within this reference. The objects are returned in ascending order by default (or if `asc` is specified) or descending order if `desc` is specified. If the collection is sorted with objects having "`__Unknown__`" values, these objects are listed at the end independent of the sort order applied to such an element. The sorting is alphabetical or numerical, depending if the `<element>` type is a string or a number, respectively. If the `sort_by` is not specified, a default sort order is provided and is documented in the Supported Operations section of each resource object.

If the `<element>` is not a valid element for the resource object (note that `<element>` is case insensitive), then an error is returned.

For example, to sort all Systems by `name` in ascending order, specify:

```
GET /systems/?sort_by=name
```

To sort all Systems by `size` in descending order, specify the following:

```
GET /systems/?sort_by=size,desc
```

> *Note:* *Densify API elements, parameters, and filters are case-sensitive.*

Other Operations (Create/Modify)

The **Create/Modify-(Req)** column of each Resource Elements table within this reference is used to identify the elements that can be specified when the resource object is created or modified. The elements for create are marked "C" and the elements for modify are marked "M". Elements that must be specified for create are marked "-R" for required.

Error Responses

All error responses (logical or caught exceptions) are in JSON format as:

```
{
  "message" : "error details",
  "status" : errorCode
}
```

Handled exceptions do not expose stack traces, to avoid security threats. Detailed stack traces are logged to the server logs for diagnostics.

Properties Returned

Table: Error Responses

Element	Type	Description
message	string	Detailed message of the exception.
status	errorCode	The HTTP response status code (e.g. 500, 403).

Examples

Example: Caught Exceptions

The response message when a UUID is expected:

Example: Caught Exception

Request:

```
GET /systems/badparm
```

Response:

```
{
  "status" : 404,
  "message" : "UUID is malformed"
}
```

Postman Collection

Densify provides a Postman collection of sample API requests for public cloud and containers.

▶ To learn more watch the following video: Using the Postman Collection.

Download the latest Densify Public Cloud Postman collection from
https://www.densify.com/docs-api/WebHelp_Densify_API_Cloud/Content/API_
Guide/Postman_Collection.htm.

Follow the steps below to use the downloaded Postman collection:

1. Unzip the downloaded file and import both the **Densify API Collection** collection and **Densify Environment** variables into your Postman workspace.

2. Modify the variables in the **Densify Environment** to match your Densify settings and credentials.

   Note: *If you already have a* **Densify** *environment in your Postman application, you can either delete the previous version or rename it. Otherwise, you will have duplicate Densify environments after the new collection is imported.*

3. Review the **Documentation** section of the collection for an overview of the workflow and API

requests.

4. Use this sample collection to familiarize yourself with Densify API requests.

Troubleshooting the Postman Collection

Token Requests

When using token-based authentication you need to request a token for each session. Subsequent API requests within the session are automatically authenticated with this token. Then in a future session you need to again request and acquire a token.

Postman is only intended to provide the tools for learning and testing Densify's APIs.

Once you are comfortable with how the Densify API works you can start to place the API calls within your code, used to interface with Densify. For example you could write a call to get all analysis IDs, then loop through each ID making another API call to get the associated recommendations, thus providing the full list of recommendations for the scope of systems.

"Bad request, invalid accept header" Error

If you use Postman v9.x or later you may encounter a "Bad request, invalid accept header" error. This is due to the fact that the Densify API does not currently support the **Accept** key value "*/" for **Headers**.

Set the **Accept** key value to "application/json".

Figure: Postman Header Configuration
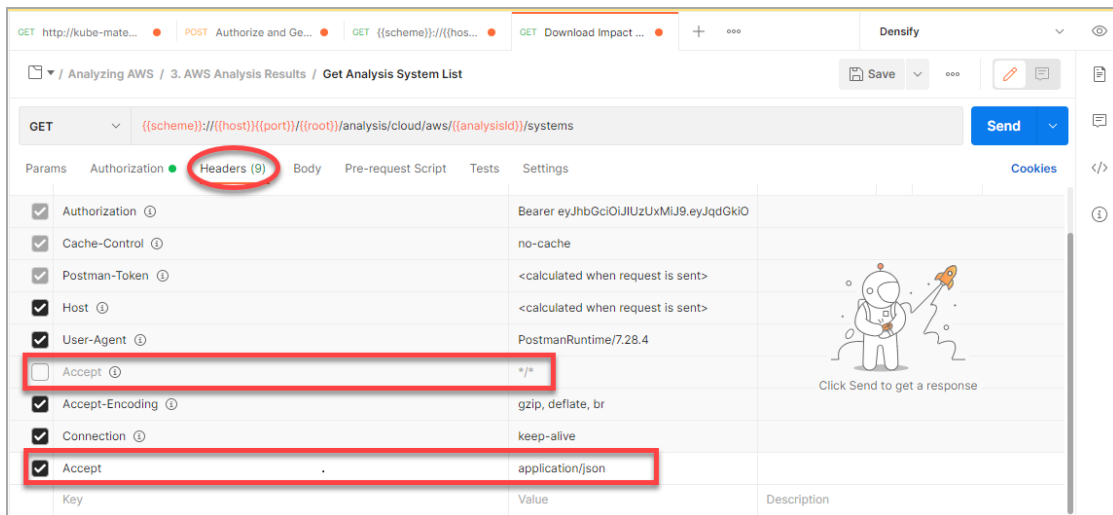
Use Cases

Optimizing AWS EC2 Instances

This use case includes API-controlled data collection, analysis, and generation of right-sizing recommendations for EC2 instances within an AWS account. The recommendations, which can be downloaded on demand or posted to a webhook-specified receiver, provide extensive details that can be used to feed downstream automation engines in either JSON or terraform-map formats.

After the reporting tables have been updated with the analysis results, you can view the recommendation reports from the Densify Console. In addition, a comprehensive PDF report can be downloaded for each instance to supply supporting evidence for a recommendation; this can be reviewed by a system owner before approvals are given for the suggested changes.

Each `/analysis/cloud/aws/<analysisId>/results` API request returns recommendations for one AWS account, since each analysis entity is defined by the AWS account connection. If you have more than one AWS account, an API request against a single account represents a subset of the full set of EC2 recommendations that are displayed in the Densify Console, since the optimization opportunities dashboard in the Densify Console contains recommendations for all connected AWS accounts. See *EC2* in the topic *Optimizing Your Public Cloud* (Help Topic ID 380320).

▶ To learn more watch the following video: Understanding the Densify API Workflow.

The diagram below provides an overview of the three major processes in Densify:

Figure: Optimizing AWS EC2 Processes
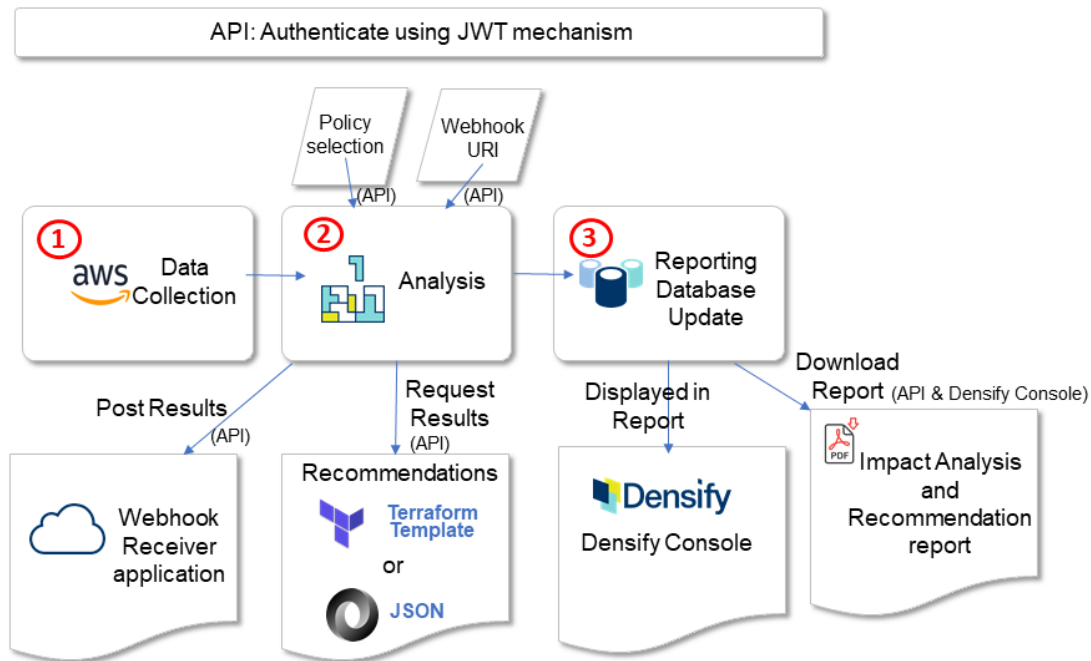
1. **Data Collection**–Collect AWS data anbd load into Densify.

   Use the Densify API or the Densify Console to initiate this process.

2. **Analysis**–Analyze collected data based on pre-defined policies to make recommendations.

   The analysis process is initiated automatically after data collection completes. Results can be immediately pulled from the Densify API or pushed to a webhook URI once the analysis is complete.

3. **Reporting Database Update**–Update reporting database tables with analysis results for reporting.

   The reporting database update process is scheduled to run nightly after the analysis process. The Densify Console optimization opportunity reports and the Impact Analysis and Recommendation reports are available after the reporting database update is complete.

Note: *Before you can collect data via AWS CloudWatch API, you need to create and configure an IAM role for Densify to have a trust relationship with your AWS account. See AWS Data Collection Prerequisites for an IAM Role (Help Topic ID 410060) for details.*

The steps below outline a typical API workflow to optimize AWS EC2 instances:

1. Authenticate using JWT–Retrieve token to authenticate each subsequent API call.

2. Optional: Review Policy–Review a list of available policies to be used for analysis.

3. Collect Data and Analyze–Start collecting data and initiate analysis (specify optional policy, optional webhook URI).

4. Check for Analysis Status–Check for data collection and analysis status or wait for a webhook trigger to indicate that the analysis finished and downstream processing has initiated.

5. Download EC2 Recommendations for Action–Retrieve recommendations for actioning or forward to orchestration engine.

6. Optional: Download Impact Analysis and Recommendation Instance Report for system owners.

Postman Collection

Densify provides a Postman collection of sample API requests for working with your public cloud environments. See the *Postman Collection* (Help Topic 340670) in the online help, to download the collection and environment settings.

Authenticate using JWT

Use the `/authorize` resource to retrieve an authenticated token to make subsequent API calls. See Authorize on page 62 for details. By default, the token expires in 5 minutes, therefore you will need to ensure that you have an active token for each Densify API request.

Example: Retrieving an Authenticated Token

Request:

```
POST /authorize
{
    "userName": "apiUser",
    "pwd": "apiPassword"
}
```

Response:

```
{
  "apiToken": "eyJh-
bGciOiJIUzUxMiJ9.eyJqdGkiOiIzNzI2Yzk0NC0wMmE4LTRlYzQtOGE2Ny04ODBmMDM2OTRhZD-
ciLCJpYXQiOjE1NDI2NTI0MDUsInN1YiI6InZh-
biIsIm-
lzcyI6IkRlb-
nNpZnkuY29tIiwiZXhwIjoxNTQyNjUyNzA1fQ.cJd8qFJfRoPnMEU7GzcdYGBT8WwlgmviQlOQp8P_
w9VUcjQA3FJaB9QkqJJ6d7zbrY5yjc4w0rOWjY-PPdbmqw",
  "expires" : 1542652705869,
  "status" : 200
}
```

Use the `apiToken` value in your Authorization request header for subsequent Densify API calls.

Review Policy

Use the `/analysis/cloud/aws/policy` resource to view a list of policies available in Densify. You can use one of these policies for optimization analysis. This is an optional step as the analysis runs with the default policy if no policy is specified. See for details.

Example: Listing AWS Cloud Policies with Description

Request:

```
GET /analysis/cloud/aws/policy?details=true
```

Response:

```
{
  "policyId": "4a63f651-a583-4157-97ff-35651370ffbe",
  "policyInstanceId": "0c0ef18b-9367-4071-b733-396f63e51925",
  "name": "DevOps-Automation"
  "description": "This policy is intended for generating instance sizing and
instance family optimization recommendations that require little or no
review before being implemented. Densify utilizes rule-driven analytics to
predict the effort of changing instance type from current to recommended,
and this policy favors Low effort recommendations, producing higher automation
at the expense of lower cost savings.\r\nThe resource utilization of each sys-
tem is modeled using a minimum of 7 days and up to 90 days of historical work-
load.\r\nWhen optimizing instance sizes and families, the predicted CPU and
memory usage must not exceed 65% and 85%, respectively.\r\nThis policy will
not specify burstable (T-series) instance families for workloads unless they
are already running in a burstable family and will not change CPU Architecture
from Intel to AMD or vice versa.\r\nWhen memory usage metrics are not avail-
able, the analysis assumes the existing memory allocation is required and will
not change the memory configuration."
  },
  {
  "policyId": "4a63f651-a583-4157-97ff-35651370ffbe",
  "policyInstanceId": "69fa4c99-1be2-4048-94a7-36fd83d07f37",
  "name": "DevOps-Default",
  "description": "This policy reflects best practices for generating instance
sizing and instance family optimization recommendations.\r\nThe resource util-
ization of each system is modeled using a minimum of 7 days and up to 60 days
of historical workload.\r\nWhen optimizing instance sizes and families, the
predicted CPU and memory usage levels must not exceed 70% and 90%, respect-
ively.\r\nWhen memory usage metrics are not available, the analysis effect-
ively assumes the existing memory allocation of the instance is required and
will not change the memory configuration."
}
```

Collect Data and Analyze

Initiate data collection and analysis for a given AWS account using the `/analysis/cloud/aws/analyze` resource. See for details.

You can specify an override policy to use for your analysis and/or specify a webhook to publish results when analysis is complete.

Example: Running AWS Data Collection and Analysis with Override Policy and Webhook

Request:

```
POST /analysis/cloud/aws/analyze
{
  "accountId": "888665225420",
  "roleArnName": "arn:aws:iam::888665225420:role/ReadOnlyAccess",
  "roleExternalId": "mypassword",
  "policyInstanceId": "f942b1a1-8f24-4032-b29a-0f628375d880",
  "webHook": {
      "uri": "http://mywebhookserver/webhook/results",
      "authType": "basic",
      "authValue": "webuser:wupassword"
  }
}
```

Response:

```
{
  "href": "Not available"
  "message": "Analysis in progress",
  "status": 200
}
```

Check for Analysis Status

To find your AWS analysis entity, request for a list of existing AWS analyses available. Refer to List all analyses for a particular platform and vendor for details.

Example: Listing all AWS Analyses Available

Request:

```
GET /analysis/cloud/aws/
```

Response:

```
{
      "analysisId": "9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf",
      "analysisName": "624756828528",
      "analysisCompletedOn": 0,
      "href": "/analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf",
      "analysisResults": "/analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0ce-
f8c5bf/results",
      "analysisStatus": "/analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0ce-
```

```
f8c5bf/status"
}
```

Use the `"analysisStatus"` resource element from the analysis entity to check for progress status. See Analysis: Status on page 206 for details on this resource.

The analysis is complete and recommendations are available when `"analysisStage"`: `"Completed"`.

Example: Checking AWS Analysis Status

Request:

```
GET /analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf/status
```

Response:

```
{
  "analysisStage": "Completed",
  "webHookStatus": "Success",
  "message": "Analysis 624756828528 was last completed on Wed Feb 27 10:03:13
EST 2019."
}
```

Download EC2 Recommendations for Action

Use the `/analysis/cloud/aws/<analysisId>/results` resource to retrieve EC2 recommendation results and forward them to a downstream orchestration engine for actioning. See Analysis: AWS Recommendations on page 83 for details.

Example: Returning Low-Effort EC2 Recommendations in Terraform-map

Request:

```
GET /analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0ce-
f8c5bf/results?serviceType=EC2&effortEstimate=Low
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

> *Note:* *This request returns low-effort EC2 recommendations only for the account defined in the specified analysis (analysisId=9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf). If you want all EC2 recommendations for your Densify collected AWS infrastructure, you must accumulate the EC2 request results for all AWS analysis entities available in Densify.*

Download Impact Analysis and Recommendation Instance Report

Use the `rptHref` resource element provided in the JSON recommendation output to download a PDF Impact Analysis and Recommendation Report for each instance. This report is available after a reporting database update, which is scheduled to run nightly. See rptHref on page 88 and Example: Downloading an Impact Analysis and Recommendation Report on page 102 for details.

Example: Downloading an Impact Analysis and Recommendation Report

Request:

```
GET /systems/9834335a-1942-4115-a65d-a298be1d390c/analysis-report
```

Headers:

```
Accept: application/octet-stream
Authorization: Bearer <apiToken>
```

Optimizing GCP Compute Engine Instances

This use case includes GCP cloud infrastructure data collection, analysis, and generation of right-sizing recommendations for Compute Engine instances using a combination of Densify Console and Densify API operations. The recommendations, which can be downloaded on demand via Densify API, provide extensive details that can be used to feed downstream automation engines in either JSON or terraform-map formats.

A comprehensive PDF report can be downloaded for each instance to supply supporting evidence for a recommendation; this can be reviewed by a system owner before approvals are given for the suggested changes.

The diagram below provides an overview of the three major processes in Densify:

Figure: Optimizing GCP Compute Engine Processes
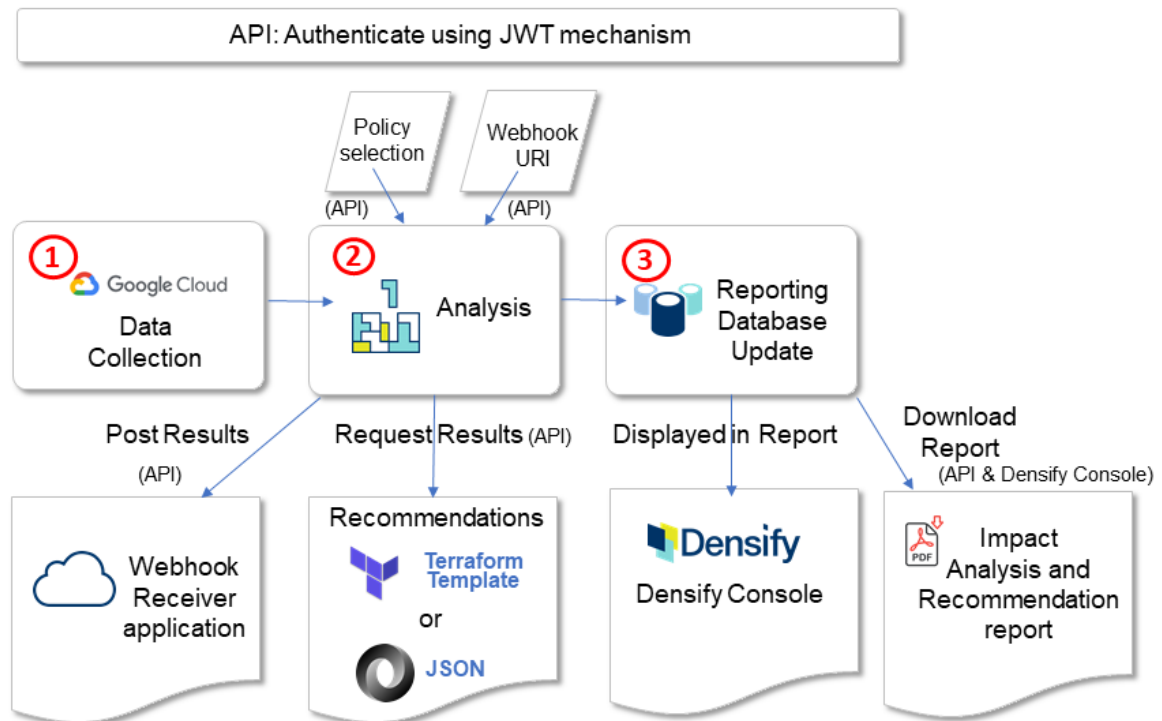
1. **Data Collection**—Collect GCP data into Densify.

    Use the Densify API or the Densify Console to initiate this process.

2. **Analysis**—Analyze collected data based on pre-defined policies to make recommendations.

    The analysis process is initiated automatically after data collection completes. Results can be immediately pulled from the Densify API or pushed to a webhook URI once the analysis is complete.

3. **Reporting Database Update**—Update reporting database tables with analysis results for reporting.

    The reporting database update process is scheduled to run nightly after the analysis refresh is completed. The Densify Console GCP optimization opportunity report and the Impact Analysis and Recommendation reports are available after the reporting database update is complete.

Note: *Before you can collect GCP cloud infrastructure data in Densify, you need to create a GCP service account with services enabled and permissions configured. See Google Cloud Platform Data Collection Prerequisites (Help Topic ID 380300) for details.*

The steps below outline a typical workflow to optimize GCP Compute Engine instances:

1. From the Densify API: Authenticate using JWT—Retrieve a token to authenticate each API call.

2. Collect GCP data through the Densify API or the Densify Console. See *Configuring a Google Cloud Platform Connection* from the topic *Using the Public Cloud Connection Wizard* (Help Topic

ID 380290) for details.

After data collection completes, an analysis is initiated for each project associated with the GCP account configured for data collection.

3. API: List All GCP Analyses—Query for the analysis entity of each project for which data was collected (from step 1). You can use the returned analysis entity to check for status and recommendations.

4. API: Poll for Analysis Status—Check for data collection and analysis status. The recommendations are available when analysis is complete.

5. API: Download Results to Apply Recommendations—Retrieve recommendations for actioning or forward to orchestration engine.

   Alternatively, you can Add a Webhook to an Existing GCP Analysis on page 35 to push recommendations to an external URI once the next analysis process completes.

6. API/Console (optional): Download Impact Analysis and Recommendation Instance Report for system owners. This can achieved from the Densify API or from the Densify Console. See *Viewing the Impact Analysis and Recommendation Report* (Help Topic ID 380450) for details on the content of the report.

7. Console (optional alternative): Review GCP Virtual Machine optimization opportunity report from the Densify Console. See *GCP* in the topic *Optimizing Your Public Cloud* (Help Topic ID 380320) for details. The optimization opportunity report is available after the nightly reporting database update.

8. Console (optional): Review GCP optimization policy settings used during analysis. Contact Support@Densify.com for details.

Postman Collection

Densify provides a Postman collection of sample API requests for working with your public cloud environments. See the *Postman Collection* (Help Topic 340670) in the online help, to download the collection and environment settings.

Authenticate using JWT

Use the /authorize resource to retrieve an authenticated token to make subsequent API calls. See Authorize on page 62 for details. By default, the token expires in 5 minutes, therefore you will need to ensure that you have an active token for each Densify API request.

Example: Retrieving an Authenticated Token

Request:

```
POST /authorize
{
    "userName": "apiUser",
    "pwd": "apiPassword"
}
```

Response:

```
{
  "apiToken": "eyJh-
bGciOiJIUzUxMiJ9.eyJqdGkiOiIzNzI2Yzk0NC0wMmE4LTRlYzQtOGE2Ny04ODBmMDM2OTRhZD-
ciLCJpYXQiOjE1NDI2NTI0MDUsInN1YiI6InZh-
biIsIm-
lzcyI6IkRlb-
nNpZnkuY29tIiwiZXhwIjoxNTQyNjUyNzA1fQ.cJd8qFJfRoPnMEU7GzcdYGBT8WwlgmviQ1OQp8P_
w9VUcjQA3FJaB9QkqJJ6d7zbrY5yjc4w0rOWjY-PPdbmqw",
  "expires" : 1542652705869,
  "status" : 200
}
```

Use the `apiToken` value in your Authorization request header for subsequent Densify API calls.

Initiating GCP Data Collection and Analysis

The following example shows you how to initiate GCP data collection and analysis, and send the results to a WebHook. See Analysis GCP Analyze.htm for details.

Example: Initiating GCP Data Collection and Analysis

Request:

```
POST /analysis/gcp/analyze
{
  "serviceAcctJSON": "gcpAcctEast-608378",
  "policyInstanceId": "4a63f651-a583-4157-97ff-35651370ffbe",
  "webHook": {
        "uri": "http://mywebhookserver/webhook/results",
        "authType": "basic",
        "authValue": "tester:testerpassword"
  }
}
```

Response:

```
{
  "href": "Not available"
  "message": "Analysis in progress",
  "status": 200
}
```

List All GCP Analyses

To see a list of all GCP analyses currently in Densify, use the `/analysis/cloud/gcp` resource. See List all analyses for a particular platform and vendor on page 149 for details of the resource operation.

Example: List all Saved GCPAnalyses

Request:

```
GET /analysis/cloud/gcp
```

Response:

```
{
    "analysisId": "cf25d8c1-4fdf-42b1-b3a3-1a8d8a425d13",
    "analysisName": "gcpAcctEast-608378",
    "analysisCompletedOn": 0,
    "href": "/analysis/cloud/gcp/cf25d8c1-4fdf-42b1-b3a3-1a8d8a425d13",
    "analysisResults": "/analysis/cloud/gcp/cf25d8c1-4fdf-42b1-b3a3-
1a8d8a425d13/results",
    "analysisStatus": "/analysis/cloud/gcp/cf25d8c1-4fdf-42b1-b3a3-
1a8d8a425d13/status"
}
```

Poll for Analysis Status

Use the `"analysisStatus"` resource element from the List All GCP Analyses response to poll for status of the data collection and analysis. See Check for analysis status on page 206 for details of the resource operation. The analysis is complete and recommendations are available when `"analysisStage": "Completed"`.

Example: Checking GCP Analysis Status

Request:

```
GET /analysis/cloud/gcp/cf25d8c1-4fdf-42b1-b3a3-1a8d8a425d13/status
```

Response:

```
{
  "analysisStage": "Completed",
  "webHookStatus": "",
  "statusMessage": "Analysis gcpAcctEast-608378 was last completed on Wed Feb
27 12:06:10 EST 2019."
}
```

Add a Webhook to an Existing GCP Analysis

You can add a webhook definition to an existing GCP analysis. The analysis will push the optimized results to the specified webhook URI. See *Add a webhook to an analysis* in the topic *Analysis: Webhook* (Help Topic ID 340490) for details on adding a webhook.

Example: Adding a Webhook Definition to an Existing GCP Analysis

Request:

```
POST /webhook/analysis/cloud/gcp/cf25d8c1-4fdf-42b1-b3a3-1a8d8a425d13
{
  "uri": "https://myInstanceConfigServer:443/api/densify/results",
  "authType": "bearer",
  "authValue": "eyJh-
bGciOiMsIUzUxMiJ9.eyJqdGkiOiJhYWU2MjIxOS1iOWQyLTQ3OGMtYWI3Mi00NGU2OTUzY2RjMDEi-
LCJpYXQiOjE1NDM2MDgxMTEsInN1YiI6ImFk-
bWluIiwiaXNzIjoiRGVuc2lmeS5jb20iLCJleHAiOjE1NDM2MDg0MTF9.h3bJrAP-
Z2LeqzjN3FYpFDyoaADvYT1MdLw5SuguqkGE7s-
jB4c7YgQgv3saj15r2IsgTWH8PW7eNnoZwFP9eiQ"
}
```

Response:

```
{
  "message": "ok",
  "status": 200
}
```

Download Results to Apply Recommendations

Use the `/analysis/cloud/gcp/<analysisId>/results` resource to retrieve recommendation results and forward to orchestration engine. See Analysis: GCP Recommendations on page 168 for details.

Example: Returning GCP Recommendations in Terraform-map

Request:

```
GET /analysis/cloud/gcp/cf25d8c1-4fdf-42b1-b3a3-1a8d8a425d13/results
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Download Impact Analysis and Recommendation Instance Report

Use the `rptHref` resource element provided in the instance recommendation output to download a PDF Impact Analysis and Recommendation Report for each instance. This report is available after a reporting database update, which is scheduled to run nightly. See rptHref on page 172 and Example: Downloading an Impact Analysis and Recommendation Report on page 179 for details.

Example: Downloading an Impact Analysis and Recommendation Report

Request:

```
GET /systems/b374d9d5-a529-4e19-9a14-077123f322b4/analysis-report
```

Headers:

```
Accept: application/octet-stream
Authorization: Bearer <apiToken>
```

Optimizing Microsoft Azure Virtual Machine Instances

This use case includes Microsoft Azure cloud infrastructure data collection, analysis, and generation of right-sizing recommendations for Virtual Machine instances using a combination of Densify Console and Densify API operations. The recommendations, which can be downloaded on demand via Densify API, provide extensive details that can be used to feed downstream automation engines in either JSON or terraform-map formats.

A comprehensive PDF report can be downloaded for each instance to supply supporting evidence for a recommendation; this can be reviewed by a system owner before approvals are given for the suggested changes.

The diagram below provides an overview of the three major processes in Densify:

Figure: Optimizing Microsoft Azure Virtual Machine Processes
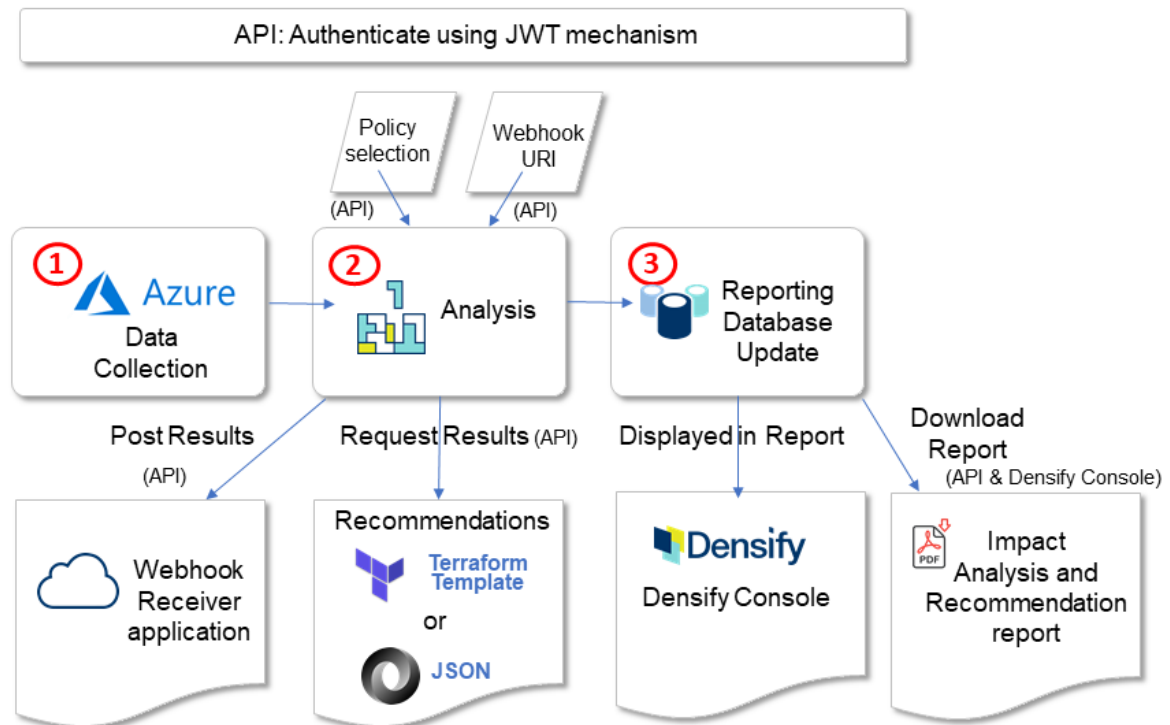
1. **Data Collection**—Collect Azure data into Densify.

   Use the Densify API or the Densify Console to initiate this process.

2. **Analysis**—Analyze collected data based on pre-defined policies to make recommendations.

   The analysis process is initiated automatically after data collection completes. Results can be immediately pulled from the Densify API or pushed to a webhook URI once the analysis is complete.

3. **Reporting Database Update**—Update reporting database tables with analysis results for reporting.

   The reporting database update process is scheduled to run nightly after the analysis refresh process. The Densify Console Azure optimization opportunity report and the Impact Analysis and Recommendation reports are available after the reporting database update is complete.

> **Note:** Before you can collect Microsoft Azure cloud infrastructure data in Densify, you need to create and configure an Azure account with access to your Azure subscriptions. See Microsoft Azure Data Collection Prerequisites (Help Topic ID 410110) or Microsoft Azure Data Collection Prerequisites for a Service Principal (Help Topic ID 410010) for details.

The steps below outline a typical workflow to optimize Microsoft Azure Virtual Machine instances:

1. From the Densify API: Authenticate using JWT—Retrieve token to authenticate each subsequent API call.

2. Collect Azure data through the Densify API or the Densify Console. See *Configuring a Microsoft Azure Connection* from the topic *Using the Public Cloud Connection Wizard* (Help Topic ID 380290) for details.

   After data collection completes, an analysis is initiated for each subscription associated with the Azure account configured for data collection.

3. API: List All Azure Analyses—Query for the analysis entity of each subscription for which data was collected (from step 1). You can use the returned analysis entity to check for status and recommendations.

4. API: Poll for Analysis Status—Check for data collection and analysis status. The recommendations are available when analysis is complete.

5. API: Download Results to Apply Recommendations—Retrieve recommendations to implement, manually or forward to your orchestration engine.

   Alternatively, you can Add a Webhook to an Existing Azure Analysis on page 42 to push recommendations to an external URI once the next analysis process completes.

6. API/Console (optional): Download Impact Analysis and Recommendation Instance Report for system owners. This can achieved from the Densify API or from the Densify Console. See *Viewing the Impact Analysis and Recommendation Report* (Help Topic ID 380450) for details of the report from the Densify Console.

7. Console (optional alternative): Review Azure Virtual Machine optimization opportunity report from the Densify Console. See *Azure* from the topic *Optimizing Your Public Cloud* (Help Topic ID 380320) for details. The optimization opportunity report is available after the nightly reporting database update.

8. Console (optional): Review Azure optimization policy settings used during analysis from the Densify Console. Contact Support@Densify.com for details.

Postman Collection

Densify provides a Postman collection of sample API requests for working with your public cloud environments. See the *Postman Collection* (Help Topic 340670) in the online help, to download the collection and environment settings.

Authenticate using JWT

Use the `/authorize` resource to retrieve an authenticated token to make subsequent API calls. See Authorize on page 62 for details. By default, the token expires in 5 minutes, therefore you will need to ensure that you have an active token for each Densify API request.

Example: Retrieving an Authenticated Token

Request:

```
POST /authorize
```

```
{
    "userName": "apiUser",
    "pwd": "apiPassword"
}
```

Response:

```
{
  "apiToken": "eyJh-
bGciOiJIUzUxMiJ9.eyJqdGkiOiIzNzI2Yzk0NC0wMmE4LTRlYzQtOGE2Ny04ODBmMDM2OTRhZD-
ciLCJpYXQiOjE1NDI2NTI0MDUsInN1YiI6InZh-
biIsIm-
lzcyI6IkRlb-
nNpZnkuY29tIiwiZXhwIjoxNTQyNjUyNzA1fQ.cJd8qFJfRoPnMEU7GzcdYGBT8WwlgmviQ1OQp8P_
w9VUcjQA3FJaB9QkqJJ6d7zbrY5yjc4w0rOWjY-PPdbmqw",
  "expires" : 1542652705869,
  "status" : 200
}
```

Use the `apiToken` value in your Authorization request header for subsequent Densify API calls.

Initiate Azure Data Collection and Analysis

The following example shows you how to initiate Azure data collection and analysis, and send the results to a WebHook. See Analysis Azure Analyze.htm for details.

Example: Running Azure Data Collection and Analysis

Request:

```
POST /analysis/azure/analyze
{
  "subscriptionId": "cc377154-9605-4cb0-8b41-1b39e1c4ac0f",
  "applicationId": "bd6529bf-85d8-410d-a207-ce963b4dd398",
  "secretKey": "w6Dvtd5wncPv7aLIXtSkzcIaIU8Y+MLwlcGuoVmFq4c=",
  "tenantId": "6c9190a7-bca6-4fcd-b35e-36378aadc695",
  "connectionName": "AzureEnvTest"

  "webHook": {
      "uri": "http://mywebhookserver/webhook/results",
      "authType": "basic",
      "authValue": "test:testpassword"
  }
}
```

Response:

```
{
  "href": "Not available",
  "message": "Analysis in progress",
  "status": 200
}
```

List All Azure Analyses

To see a list of all Azure analyses currently in Densify, use the `/analysis/azure` resource. See List all analyses for a particular platform and vendor on page 149 for details of the resource operation.

Example: List all Saved Azure Analyses

Request:

```
GET /analysis/cloud/azure
```

Response:

```
{
    "analysisId": "398d26f3-b705-4fa6-8d31-16724ae320a2",
    "analysisName": "00d89cbc-bc00-4d00-bcf6-ce6ec08d8fbc",
    "analysisCompletedOn": 1522179715493,
    "href": "/analysis/cloud/azure/398d26f3-b705-4fa6-8d31-16724ae320a2",
    "analysisResults": "/analysis/cloud/azure/398d26f3-b705-4fa6-8d31-
16724ae320a2/results",
    "analysisStatus": "/analysis/cloud/azure/398d26f3-b705-4fa6-8d31-
16724ae320a2/status"
}
```

Poll for Analysis Status

Use the `"analysisStatus"` resource element from the List All Azure Analyses response to poll for status of the data collection and analysis. See Check for analysis status on page 206 for details of the resource operation. The analysis is complete and recommendations are available when `"analysisStage": "Completed"`.

Example: Checking Azure Analysis Status

Request:

```
GET /analysis/cloud/azure/398d26f3-b705-4fa6-8d31-16724ae320a2/status
```

Response:

```
{
  "analysisStage": "Completed",
  "webHookStatus": "",
  "statusMessage": "Analysis 00d89cbc-bc00-4d00-bcf6-ce6ec08d8fbc was last com-
pleted on Mon Feb 25 11:25:11 EST 2019."
}
```

Add a Webhook to an Existing Azure Analysis

You can add a webhook definition to an existing Azure analysis. The analysis will push the optimized results to the specified webhook URI. See *Add a webhook to an analysis* in the topic *Analysis: Webhook* (Help Topic ID 340490) for details on adding a webhook.

Example: Adding a Webhook Definition to an Existing Azure Analysis

Request:

```
POST /webhook/analysis/cloud/azure/398d26f3-b705-4fa6-8d31-16724ae320a2
{
  "uri": "https://myInstanceConfigServer:443/api/densify/results",
  "authType": "bearer",
  "authValue": "eyJh-
bGciOiMsIUzUxMiJ9.eyJqdGkiOiJhYWU2MjIxOS1iOWQyLTQ3OGMtYWI3Mi00NGU2OTUzY2RjMDEi-
LCJpYXQiOjE1NDM2MDgxMTEsInN1YiI6ImFk-
bWluIiwiaXNzIjoiRGVuc2lmeS5jb20iLCJleHAiOjE1NDM2MDg0MTF9.h3bJrAP-
Z2LeqzjN3FYpFDyoaADvYT1MdLw5SuguqkGE7s-
jB4c7YgQgv3saj15r2IsgTWH8PW7eNnoZwFP9eiQ"
}
```

Response:

```
{
  "message": "ok",
  "status": 200
}
```

Download Results to Apply Recommendations

Use the `/analysis/cloud/azure/<analysisId>/results` resource to retrieve recommendation results and forward to orchestration engine. See Analysis: Azure Recommendations on page 122 for details.

Example: Returning Recommendations in Terraform-map

Request:

```
GET /analysis/cloud/azure/398d26f3-b705-4fa6-8d31-16724ae320a2/results
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Download Impact Analysis and Recommendation Instance Report

Use the `rptHref` resource element provided in the instance recommendation output to download a PDF Impact Analysis and Recommendation Report for each instance. This report is available after a reporting database update, which is scheduled to run nightly. See rptHref on page 127 and Example: Downloading an Impact Analysis and Recommendation Report on page 137 for details.

Example: Downloading an Impact Analysis and Recommendation Report

Request:

```
GET /systems/bdbdbd19-6928-4e7a-894c-e7e215687a4d/analysis-report
```

Headers:

```
Accept: application/octet-stream
Authorization: Bearer <apiToken>
```

Optimizing Kubernetes Containers

This use case provides an overview of how to collect Kubernetes container data into Densify and how to obtain Densify container recommendations.

The diagram below shows the major processes in Densify to optimize Kubernetes containers:

Figure: Optimizing Kubernetes Containers Processes
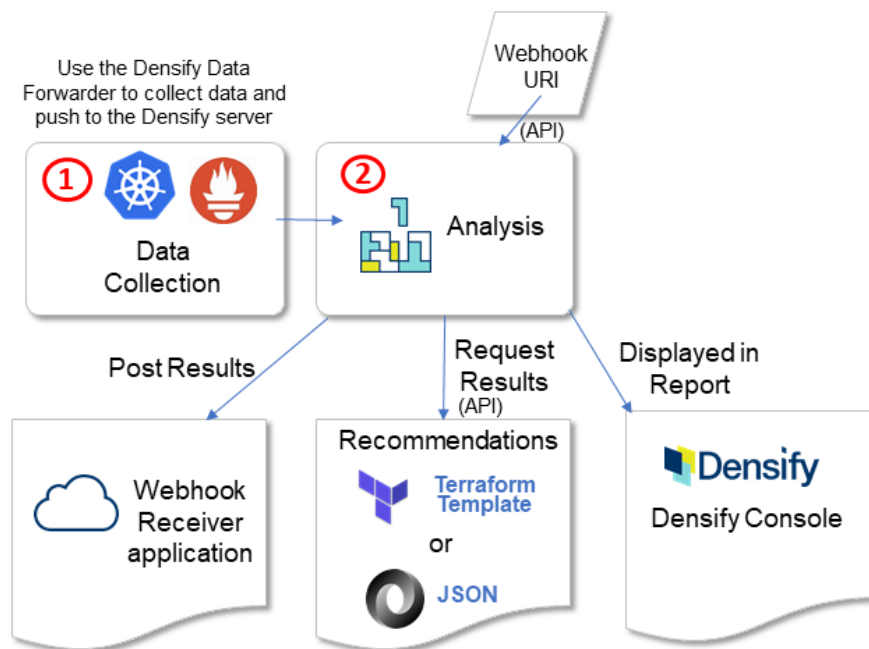
1. **Data Collection**–Collect Kubernetes data into Densify.

   You need to have an existing monitoring and metrics collection service, such as Prometheus, configured for your container clusters. See Container PrerequisitesContainer Prerequisites (Help Topic ID 410140) for details.

   Densify provides a collection and transfer tool called the Data Forwarder, which you can download from Docker Hub, configure, and launch into a container to transfer node information and resource metrics from Prometheus to Densify.

   Your Densify instance needs to be enabled with Data Forwarder support to accept and ingest incoming data for optimization. Contact Densify support (Support@Densify.com) to enable Data Forwarder support.

   > **Note:** You will need a Data Forwarder container for each Prometheus server you want to collect data from.

2. **Analysis**–Analyze collected data based on pre-defined policies to make recommendations.

   The analysis process is initiated after data is ingested into a Densify data model. Results can be pulled from the Densify API or pushed to a webhook URI once the analysis is complete.

The steps below outline a typical workflow to access Kubernetes container recommendations:

1. From the Densify API: Authenticate using JWT–Retrieve token to authenticate each subsequent API call.

2. API: List All Kubernetes Container Analyses—Query for the analysis entity of each cluster for which Kubernetes container data was collected.

3. API: Download Results for Action—Retrieve recommendations for actioning or forward to orchestration engine.

   Alternatively, you can Add a Webhook to an Existing Kubernetes Container Analysis on page 46 to push recommendations to an external URI once the next analysis process completes.

4. Console (optional alternative): Review the Kubernetes Container Optimization report from the Densify Console. See *Optimizing Your Containers* (Help Topic ID 380520) for details.

Postman Collection

Densify provides a Postman collection of sample API requests for working with your public cloud environments. See the *Postman Collection* (Help Topic 340670) in the online help, to download the collection and environment settings.

Authenticate using JWT

Use the `/authorize` resource to retrieve an authenticated token to make subsequent API calls. See Authorize on page 62 for details. By default, the token expires in 5 minutes, therefore you will need to ensure that you have an active token for each Densify API request.

Example: Retrieving an Authenticated Token

Request:

```
POST /authorize
{
    "userName": "apiUser",
    "pwd": "apiPassword"
}
```

Response:

```
{
  "apiToken":
"eyJhbGciOiJIUzUxMiJ9.eyJqdGkiOiIzNzI2Yzk0NC0wMmE4LTRlYzQtOGE2Ny04ODBmMDM2OTRhZ
DciLCJpYXQiOjE1NDI2NTI0MDUsInN1YiI6InZhbiIsImlzcyI6IkRlbnNpZnkuY29tIiwiZXhwIjox
NTQyNjUyNzA1fQ.cJd8qFJfRoPnMEU7GzcdYGBT8WwlgmviQ1OQp8P_
w9VUcjQA3FJaB9QkqJJ6d7zbrY5yjc4w0rOWjY-PPdbmqw",
  "expires" : 1542652705869,
  "status" : 200
}
```

Use the `apiToken` value in your Authorization request header for subsequent Densify API calls.

List All Kubernetes Container Analyses

To see a list of all Kubernetes analyses currently in Densify, use the `/analysis/containers/kubernetes` resource. See List all analyses for a particular platform and vendor on page 149 for details of the resource operation.

Example: List all Saved Kubernetes Container Analyses

Request:

```
GET /analysis/containers/kubernetes
```

Response:

```
    {
        "analysisId": "d7298ac3-a143-41bb-b7d7-62f659f2a8c5",
        "analysisName": "WEST222-Bench",
        "analysisCompletedOn": 1548889723847,
        "href": "/analysis/containers/kubernetes/d7298ac3-a143-41bb-
62f659f2a8c5",
        "analysisResults": "/analysis/containers/kubernetes/d7298ac3-a143-41bb-
b7d7-62f659f2a8c5/results",
        "analysisStatus": "/analysis/containers/kubernetes/d7298ac3-a143-41bb-
b7d7-62f659f2a8c5/status"
    }
```

Poll for Analysis Status

Use the `"analysisStatus"` resource element from the List All Kubernetes Container Analyses
response to poll for status of the data collection and analysis. See Check for analysis status on page
206 for details of the resource operation. The analysis is complete and recommendations are
available when `"analysisStage": "Completed"`.

Example: Checking Kubernetes Analysis Status

Request:

```
GET analysis/containers/kubernetes/d7298ac3-a143-41bb-b7d7-62f659f2a8c5/status
```

Response:

```
{
  "analysisStage": "Completed",
  "webHookStatus": "",
  "statusMessage": "Analysis WEST222-Bench was last completed on Wed Jan 30
19:03:41 EST 2019."
}
```

Add a Webhook to an Existing Kubernetes Container Analysis

You can add a webhook definition to an existing Kubernetes container analysis. The analysis will push
the optimized results to the specified webhook URI. See *Analysis: Webhook* (Help Topic ID 340490)
for details on managing webhooks.

Example: Adding a Webhook Definition to an Existing Kubernetes Container Analysis

Request:

```
POST /webhook/analysis/containers/kubernetes/d7298ac3-a143-41bb-b7d7-
62f659f2a8c5
{
  "uri": "https://myInstanceConfigServer:443/api/densify/ContainerResults",
  "authType": "bearer",
```

```
  "authValue":
"eyJhbGciOiMsIUzUxMiJ9.eyJqdGkiOiJhYWU2MjIxOS1iOWQyLTQ3OGMtYWI3Mi00NGU2OTUzY2Rj
MDEiLCJpYXQiOjE1NDM2MDgxMTEsInN1YiI6ImFkbWluIiwiaXNzIjoiRGVuc2lmeS5jb20iLCJleHA
iOjE1NDM2MDg0MTF9.h3bJrAP-
Z2LeqzjN3FYpFDyoaADvYT1MdLw5SuguqkGE7sjB4c7YgQgv3saj15r2IsgTWH8PW7eNnoZwFP9eiQ"
}
```

Response:
```
{
  "message": "ok",
  "status": 200
}
```

Download Results for Action

Use the `/analysis/containers/kubernetes/<analysisId>/results` resource to retrieve recommendation results and forward to orchestration engine. See Analysis: Kubernetes Container Recommendations on page 185 for details.

Example: Returning Kubernetes Container Recommendations in Terraform-map

Request:
```
GET /analysis/containers/kubernetes/d7298ac3-a143-41bb-b7d7-
62f659f2a8c5/results
```

Headers:
```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Subscribing to Densify Recommendations

This use case shows you how to subscribe to a set of Densify recommendation notifications using the Densify Subscription Service (DSS) API framework.

The DSS is a flexible framework that offers you the ability to customize the content of your Densify recommendations for targeted distribution. DSS features include:

- ability to filter recommendation systems based on recommendation fields (property filter);
- ability to filter recommendation systems based on attribute tags (tag filter);
- ability to suppress recommendation systems, for a period of time, based on recommendation fields and/or attribute tags via suppression filters;
- ability to customize the notification output format;
- ability to send the notification to an external webhook;
- ability to schedule the notification;
- ability to have private subscription definitions, as well as global subscription definitions for shared usage;
- supports both cloud and containers system recommendations.

To use these features, you need to create an instance of the Subscription resource.

Creating a Subscription

The diagram below provides a high-level overview of creating an instance of the Subscription resource through the Densify API:

Figure: Subscription High Level Overview
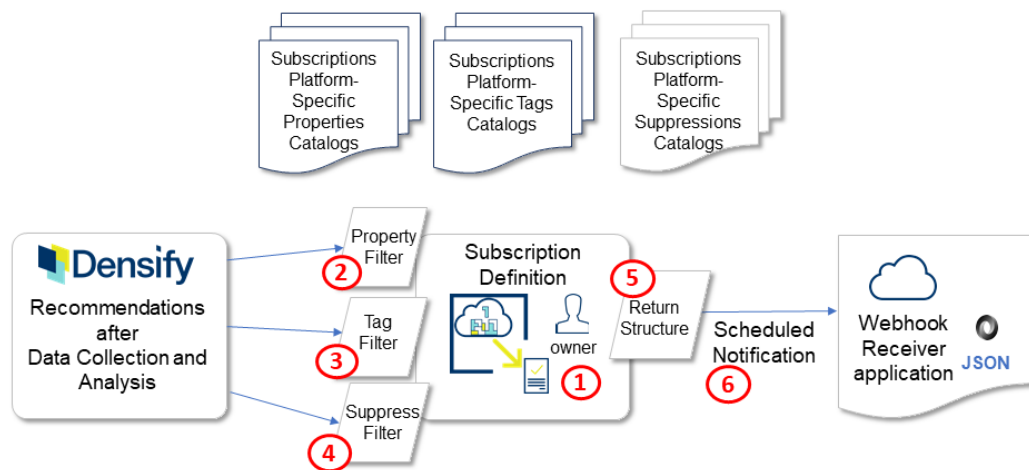
An instance of the Subscriptions on page 274 resource is created using the `POST /subscriptions/<platformType>` request. You must identify the `<platformType>` of the subscription and use similar platform resources to define the subscription. For instance, you need to use container resources for a subscription to container recommendations. After identifying the subscription platform resources to use, define the parameter components below to customize your subscription.

1. **Define the Subscription**—Specify the subscriptionName, description, and owner.

   The `subscriptionName` is used as a key identifier, so it needs to be unique for global subscriptions as well as unique within a private-scoped group of subscriptions. The `description` parameter is for documentation purposes, so its content should be as descriptive as possible to indicate the purpose of the subscription.

   See Example: Defining the Subscription.

2. **Property Filter**—Specify propertyReferences to filter the recommendation systems you are interested in receiving based on recommendation fields.

   This feature is similar to the Recommendation Filter Menu in the Densify Console, where you can filter recommendation systems based on selected fields from the menu. In the `propertyReferences` filter condition, fields are selected by referencing property elements from the platform-specific Subscriptions Properties catalog. The platform of properties you reference must correspond with the platform of your subscription. Refer to Subscriptions Properties Catalogs for details on referencing properties.

See Example: Specifying Property Filters for an example of a property filter condition block in a `POST /subscriptions/<platformType>` request.

3. **Tag Filter**—Specify tagReferences to filter the recommendation systems you are interested in receiving based on attribute fields.

   This feature is similar to the Guest Filter from the Densify Console, but it also extends to filter platform-specific attribute tags. In the `tagReferences` filter condition, attributes are selected by referencing tag elements from the platform-specific Subscriptions Tags catalog. The platform of tags you reference must correspond with the platform of your subscription. Refer to Subscriptions Tags Catalogs for details.

   > *Note:* *You must ensure that the selected Densify attributes are marked as* `searchable` *for use with DSS. Contact Densify support (Support@Densify.com) to confirm that the selected attributes are searchable in your Densify instance.*

   See Example: Specifying Tag Filters for an example of a tag filter condition block in a `POST /subscriptions/<platformType>` request.

4. **Suppress Filter**—Specify the suppressionReferences condition to suppress recommendation systems you are not interested in receiving based on recommendation or attribute fields.

   This feature allows you to suppress groups of recommendation systems for a period of time when the notification is actively scheduled. Suppression conditions are created by referencing suppression elements from the platform-specific Subscriptions Suppressions catalog. The platform of suppression conditions you reference must correspond with the platform of your subscription. Refer to the Subscriptions Suppressions Catalogs for an explanation of referencing the catalog.

   See Example: Specifying Suppression Conditions for an example of a suppression condition block in a `POST /subscriptions/<platformType>` request.

5. **Return Structure**—Specify the returnStructure to personalize the output of the notification.

   If you do not specify the `returnStructure` parameter in your subscription, all recommendation fields applicable to the platform-specific system are returned. To return specific fields, you can specify elements from the platform-specific Subscriptions Properties Catalogs and Subscriptions Tags Catalogs that are within your scope.

   See Example: Specifying the Return Structure for an example of a `returnStructure` block in a `POST /subscriptions/<platformType>` request.

6. **Schedule Notification to Webhook**—Specify when and where to send your notification by configuring the schedule parameter and the webhook parameter.

   If you do not specify the `schedule` parameter, the notification is triggered each night after data collection, analysis, and reporting database update processes. See Example: Specifying the Notification Schedule for an example of specifying a `schedule` block in a `POST /subscriptions/<platformType>` request.

   The `webhook` parameter is mandatory if you want notifications to be triggered. See Example: Specifying the Webhook for an example of specifying a `webhook` block in a `POST /subscriptions/<platformType>` request.

> **Note:** *You can test the subscription and request an on-demand results output by using the* Subscriptions: Results on page 299 *resource, even if the webhook is not defined.*

See Example: Creating a Subscription (Putting It All Together) for a complete example of the `POST /subscriptions/<platformType>` request with all the combined parameter components; then review the on-demand results of the subscription in Example: Getting On-Demand Results.

Refer to Subscriptions on page 274 for the complete subscription resource reference.

Subscriptions Properties Catalogs

Figure: Subscriptions Properties Catalogs Reference Overview
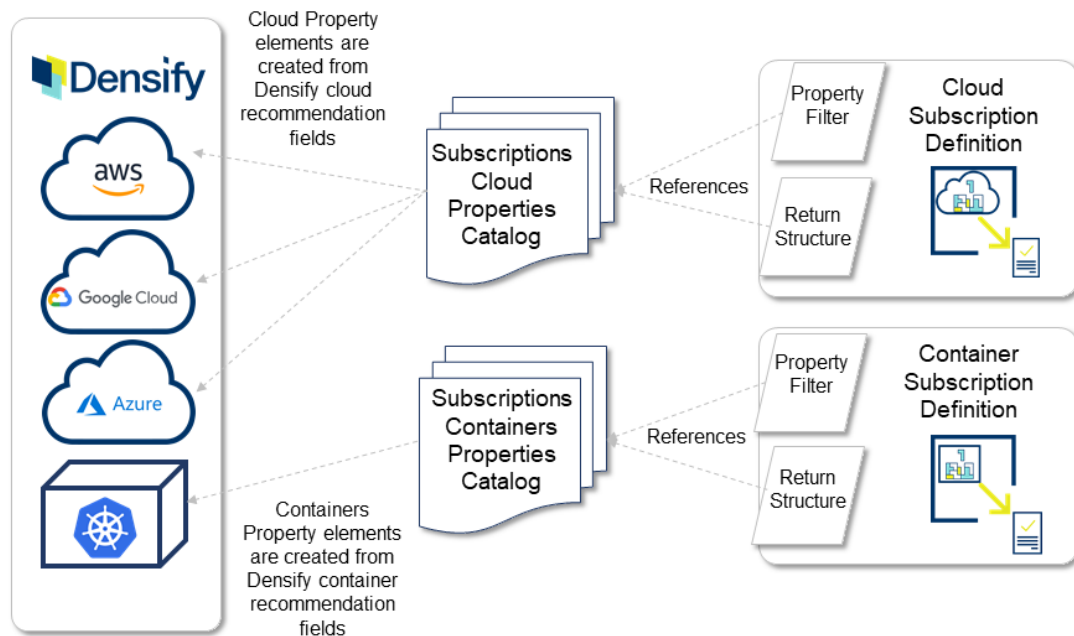
The platform-specific Subscriptions Properties catalogs are resources that provides you with a list of recommendation fields to use for filtering and for personalizing the notification output. The platform of the Subscriptions Properties catalog you use for filtering must correspond with the platform of your subscription.

By default, the platform-specific Subscriptions Properties catalogs already have a list of recommendation fields for you to use. You can also add additional recommendation fields to the catalogs. The list of possible recommendation fields to add corresponds to the response elements from the Densify supported cloud and container recommendations (see Analysis: AWS Recommendations: Response on page 86, Analysis: Azure Recommendations: Response on page 125, Analysis: GCP Recommendations: Response on page 171, and Analysis: Kubernetes Container Recommendations: Response on page 188).

> **Note:** *Some recommendation elements are not common to all technologies. For example, propertyName="minGroupRecommended" only applies to AWS Auto Scaling group recommendations.*

When you create property filter conditions for your subscription, you need to reference properties from the corresponding platform Subscriptions Properties catalog (see the Subscriptions: propertyReferences on page 280 parameter). Similarly, you need to reference properties from the corresponding platform Subscriptions Properties catalog to personalize the notification output (see the Subscriptions: returnStructure on page 282 parameter).

You can only reference properties that are accessible to the Densify username you use to authorize the API request. Property accessibility depends on the scope of the property: if the property is *global*, then the property is accessible to all API Densify users; if the property is *private*, then the property is accessible to the owner of the private property. If you are an **administrative user**[1], then you can override the property scope rule and access all properties from the catalog. See Subscriptions: Properties: owner on page 313 for more property scope details.

Refer to Subscriptions: Properties on page 306 for the full resource definition.

Subscriptions Tags Catalogs

Figure: Subscriptions Tags Catalogs Reference Overview
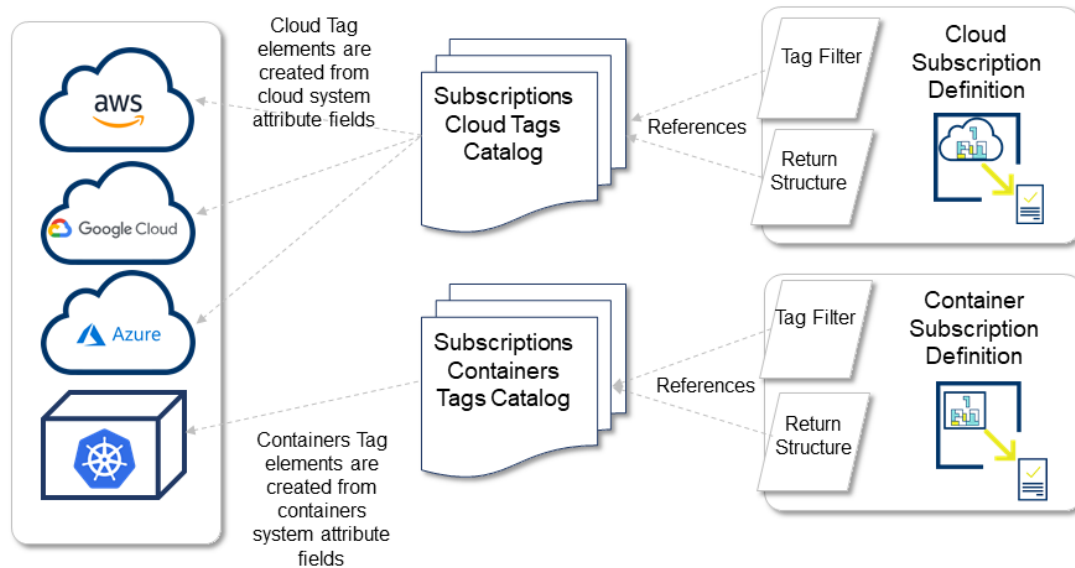
---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

The platform-specific Subscriptions Tags catalogs are resources that provides you with a list of system attribute fields to use for filtering and for personalizing the notification output. The platform of the Subscriptions Tags catalog you use for filtering must correspond with the platform of your subscription.

By default, the platform-specific Subscriptions Tags catalogs already have a list of common attributes for you to use. You can also add additional attribute fields to the catalogs. The possible attribute fields you can add comes from the set of Densify standard attributes or technology-specific attributes.

When you create tag filter conditions for your subscription, you need to reference tags from the corresponding platform Subscriptions Tags catalog (see the Subscriptions: tagReferences on page 281 parameter). Similarly, you need to reference tags from the corresponding platform Subscriptions Tags catalog to personalize the notification output (see the returnStructure parameter).

You can only reference tags that are accessible to the Densify username you use to authorize the API request. Access to a particular tag in the catalog depends on its scope: if the tag is *global*, then the tag is accessible to all API Densify users; if the tag is *private*, then the tag is accessible to the owner of the private tag. If you are an **administrative user**[1], then you can override the tag scope rule and access all tags from the catalog. See Subscriptions: Tags: owner on page 354 for additional details of tag scope.

Refer to Subscriptions: Tags on page 347 for the full resource definition.

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Subscriptions Suppressions Catalogs

Figure: Subscriptions Suppressions Catalogs Reference Overview
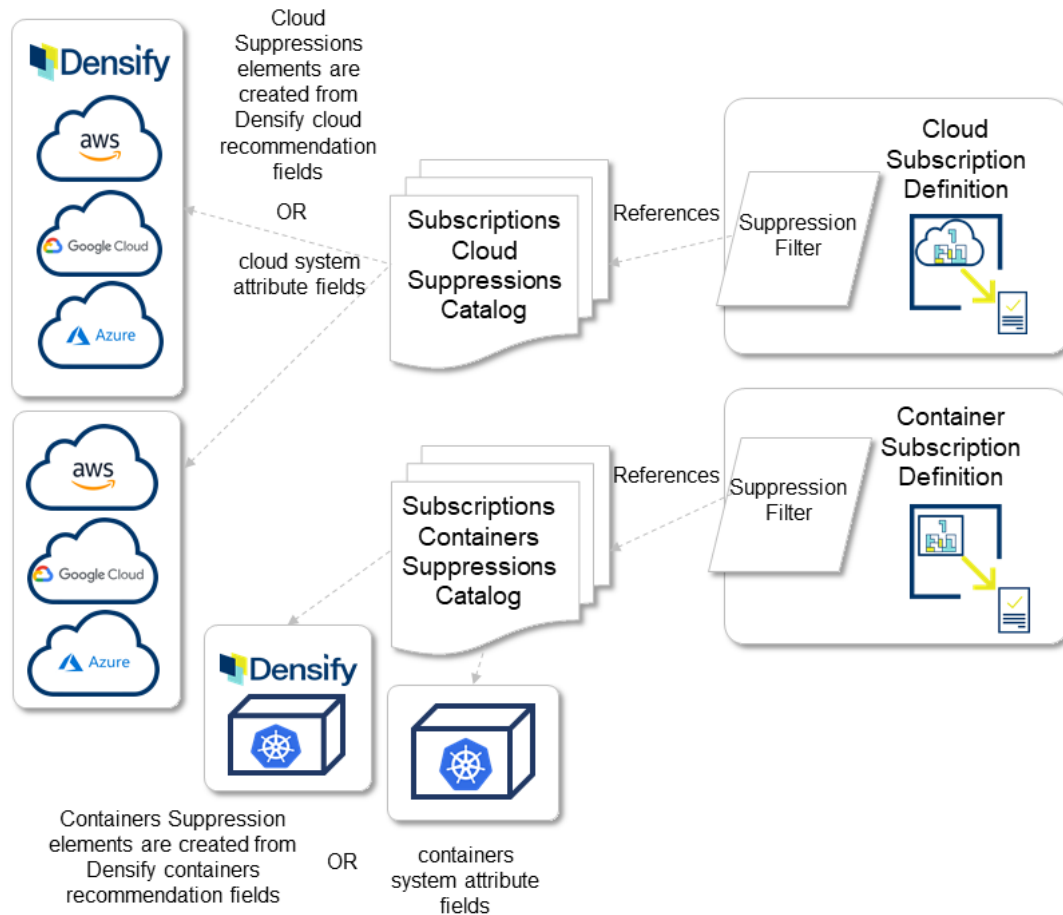
The platform-specific Subscriptions Suppressions catalogs are resources that provides you with a list of recommendation and system attribute fields to use for creating suppression conditions in your subscription. The platform of the Subscriptions Suppressions catalog you use for suppressing recommendations must correspond with the platform of your subscription.

By default, the platform-specific Subscriptions Suppressions catalog already has a list of recommendation and attribute field suppressions for you to use. You can also add additional suppression fields. The list of possible recommendation fields correspond to the response elements from the Densify supported cloud recommendations (see Analysis: AWS Recommendations: Response on page 86, Analysis: Azure Recommendations: Response on page 125, Analysis: GCP Recommendations: Response on page 171, and Analysis: Kubernetes Container Recommendations: Response on page 188). The possible attribute fields you can add to a Subscriptions Suppressions catalog comes from the set of Densify standard attributes or technology-specific attributes.

When you create suppression conditions for your subscription, you need to reference suppressions from the corresponding platform Subscriptions Suppressions catalog (see the Subscriptions: suppressionReferences on page 281 parameter).

You can only reference suppressions that are accessible to the Densify username you use to authorize the API request. Access to a particular suppression in the catalog depends on its scope: if the suppression is *global*, then it is accessible to all API Densify users; if the suppression is *private*, then it is accessible to the owner of the private suppression. If you are an **administrative user**[1], then you can override the suppression scope rule and access all suppressions from the catalog. See Subscriptions: Suppressions: owner on page 339 for details of suppression scope.

> *Note:* *The Subscriptions Suppressions catalogs are not used for a Subscription's return output.*

Refer to Subscriptions: Suppressions on page 330 for the full resource definition.

Example: Defining the Subscription

Use the `POST /subscriptions/cloud` request to create a cloud subscription. The first set of parameters to specify are the subscriptionName and description, which are free-form strings. The owner parameter is automatically set to your Densify username if you are not an **administrative user**[2]. You can also set the active parameter to indicate if the subscription is dormant or active (the default is `active`). Remember to specify your active authorization key for each token-based authentication request (see Authorize on page 62 for details).

Example: Specifying Subscription Name and Description

Headers:

```
Accept: application/json
Authorization: Bearer <apiToken>
```

Request:

```
POST /subscriptions/cloud

{
    "subscriptionName": "My Subscription",
    "description": "A subscription for my Use Case",
    "active": true,
    ...
```

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.
[2]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Example: Specifying Property Filters

To specify property filter conditions in the `POST /subscriptions/<platformType>` request, use the propertyReferences parameter. In this example, the property condition filters all AWS EC2s with predicted uptime between 50 -100%. The cloud subscription `propertyReferences` condition references the `propertyID` of `"serviceType"` and `"predictedUptime"` from the Cloud Subscriptions Properties catalog.

Example: Subscription - Property Filter Condition

Request Parameter - Property Filter Condition Section:

```
...
 "propertyReferences": [
    {
        "values": ["EC2"],
        "propertyID": "b4e7260f-1aae-4150-b75d-1b1234075500",
        "operator": "="
    },
    {
        "propertyID": "f2a38773-db60-478a-9982-1a2d1ba7d380",
        "operator": "[]",
        "values": [50,100]
    }
  ],
  ...
```

Example: Specifying Tag Filters

To specify tag filter conditions in the `POST /subscriptions/<platformType>` request, use the tagReferences parameter. In this example, the tag condition filters all systems belonging to the "Sales" department. The cloud subscription `tagReferences` condition references the `tagID` of `"Department"` from the Cloud Subscriptions Tags catalog.

Example: Subscription - Tag Filter Condition

Request Parameter - Tag Filter Condition Section:

```
...
 "tagReferences": [
    {
        "values": ["Sales"],
        "tagID": "9879308e-3162-499b-8fe5-7fd67e627079",
        "operator": "="
    }
  ],
  ...
```

Example: Specifying Suppression Conditions

To specify suppression conditions in the `POST /subscriptions/<platformType>` request, use the [suppressionReferences](#) parameter. In this example, the cloud suppression condition removes all terminate recommendations until April 20, 2020. After April 20, 2020 (the `revokeBy` date), the suppression condition is deprecated. The cloud subscription `suppressionReferences` condition references the `suppressionID` of "`recommendationType`" from the Cloud Subscriptions Suppressions catalog.

Example: Subscription - Suppression Filter Condition

Request Parameter - Suppression Filter Condition Section:

```
...
"suppressionReferences": [
    {
        "values": ["Terminate"],
        "suppressionID": "a6827ae4-fa2b-405e-a564-d70f2dad45c2",
        "operator": "=",
        "revokeBy": 1587415413000
    }
 ],
...
```

Example: Specifying the Return Structure

To specify a customized return structure in the `POST /subscriptions/<platformType>` request, use the [returnStructure](#) parameter. In this example, the return output will contain the following cloud fields: `effortEstimate`, `entityId`, `name`, `recommendationType`, `savingsEstimate`, `serviceType`, `Department` (attribute), `Virtual Domain` ("account"). These fields reference the `propertyID` and "`tagID`" from the Cloud Subscriptions Properties catalog and the Cloud Subscriptions Tags catalog, respectively. Some of these property or tag fields have the "`useAlias`" flag set to true, which returns the alias names instead of the property or tag names.

Example: Subscription - Return Structure

Request Parameter - Return Structure Section:

```
...
"returnStructure": {
    "properties": [
        {
            "propertyID": "b4e7260f-1aae-4150-b75d-1b1234075500"
        },
        {
            "useAlias": true,
            "propertyID": "e2ae92c5-91c7-4ff5-a29f-99aa92d65178"
        },
```

```
            {
                "propertyID": "08d84679-1816-4cd7-b766-e4ad441b9a6c"
            },
            {
                "propertyID": "b20b2499-e901-4d79-873c-5e953ae8be33"
            },
            {
                "useAlias": true,
                "propertyID": "cebcd841-89d8-4007-a4c6-1f0b06723db4"
            },
            {
                "propertyID": "88695e33-19c4-4813-819a-31e59322da6e"
            }
        ],
        "tags": [
            {
                "useAlias": true,
                "tagID": "be3cdc69-962e-4461-b8c3-07cec8e712be"
            },
            {
                "tagID": "9879308e-3162-499b-8fe5-7fd67e627079"
            }
        ]
    },
    ...
```

Example: Specifying the Webhook

For the destination of the notification, specify the webhook parameter in the
`POST /subscriptions/<platformType>` request. If you do not specify a `webhook`, no subscription
notifications are triggered; the subscription is considered dormant since there is no destination for the
notification. You can test the subscription and request an on-demand results output by using the
Subscriptions: Results on page 299 resource, even if the subscription is dormant or if the webhook is
not defined.

Example: Subscription - Webhook

Request Parameter - Webhook Section:

```
    ...
    "webhook": {
        "uri": "https://testsystem.com/webhook/receiver",
        "authType": "basic",
        "authValue": "test:test"
    },
    ...
```

Example: Specifying the Notification Schedule

To specify a notification schedule in the `POST /subscriptions/<platformType>` request, use the schedule parameter. In the example below, the cloud subscription is scheduled to be triggered on Mondays and Fridays of each week (`dayofWeek 1` = Monday). If you do not specify the `schedule` parameter, the subscription notifications are triggered nightly by default.

Example: Subscription - Schedule

Request Parameter - Schedule Section:

```
...
"schedule": {
    "dayofWeek": [
        1,
        5
    ]
},
...
```

Example: Creating a Subscription (Putting It All Together)

This example puts all the subscription components (from the previous examples) together to create a cloud subscription.

Example: Creating a Cloud Subscription

Request:

```
POST /subscriptions/cloud
```

Parameters:

```
[
    {
        "subscriptionName": "My Subscription",
        "description": "A subscription for my Use Case",
        "active": true,
        "webhook": {
            "uri": "https://testsystem.com/webhook/receiver",
            "authType": "basic",
            "authValue": "test:test"
        },
        "propertyReferences": [
            {
                "values": ["EC2"],
```

```
                    "propertyID": "b4e7260f-1aae-4150-b75d-1b1234075500",
                    "operator": "="
                },
                {
                    "propertyID": "f2a38773-db60-478a-9982-1a2d1ba7d380",
                    "operator": "[]",
                    "values": [50,100]
                }
            ],
            "tagReferences": [
                {
                    "values": ["Sales"],
                    "tagID": "9879308e-3162-499b-8fe5-7fd67e627079",
                    "operator": "="
                }
            ],
            "suppressionReferences": [
                {
                    "values": ["Terminate"],
                    "suppressionID": "a6827ae4-fa2b-405e-a564-d70f2dad45c2",
                    "operator": "=",
                    "revokeBy": 1587415413000
                }
            ],
            "returnStructure": {
                "properties": [
                    {
                        "propertyID": "b4e7260f-1aae-4150-b75d-1b1234075500"
                    },
                    {
                        "useAlias": true,
                        "propertyID": "e2ae92c5-91c7-4ff5-a29f-99aa92d65178"
                    },
                    {
                        "propertyID": "08d84679-1816-4cd7-b766-e4ad441b9a6c"
                    },
                    {
                        "propertyID": "b20b2499-e901-4d79-873c-5e953ae8be33"
                    },
                    {
                        "useAlias": true,
                        "propertyID": "cebcd841-89d8-4007-a4c6-1f0b06723db4"
                    },
                    {
                        "propertyID": "88695e33-19c4-4813-819a-31e59322da6e"
                    }
                ],
                "tags": [
                    {
                        "useAlias": true,
                        "tagID": "be3cdc69-962e-4461-b8c3-07cec8e712be"
                    },
                    {
                        "tagID": "9879308e-3162-499b-8fe5-7fd67e627079"
                    }
                ]
            },
```

```
            "schedule": {
                "dayofWeek": [
                    1,
                    5
                ]
            }
        }
    ]
```

Response:

```
[
    {
        "subscriptionRef": "7f2b2f50-3988-4275-a1ac-34a9b45408b9",
        "subscriptionName": "My Subscription"
    }
]
```

Example: Getting On-Demand Results

This example retrieves the cloud subscription (created from the previous example) results on-demand.
This request returns the cloud subscription results regardless of what is configured in the `webhook`,
`schedule`, or `active` parameters.

Example: Getting Cloud Subscription Results On-Demand

Request:

```
GET /subscriptions/cloud/7f2b2f50-3988-4275-a1ac-34a9b45408b9
```

Response:

```
{
    "subscription": {
        "name": "My Subscription",
        "description": "A subscription for my Use Case",
        "created": "Mon Apr 13 13:50:15 EST 2020",
        "createdBy": "saas",
        "lastRefreshed": "Fri Apr 17 1:00:10 EST 2020",
        "owner": "saas"
    },
    "count": 98,
    "results": [
        {
            "effortEstimate": "Low",
            "entityId": "c0ebf351-9b6c-4d57-b3d0-23c7861f9db4",
            "name": "ex-pro-asop-896",
```

```
                "recommendationType": "Downsize - Optimal Family",
                "savingsEstimate": "49.39808",
                "serviceType": "EC2",
                "divider": "-----------------------",
                "Department": "Sales",
                "awsAccount": "229132289155"
            },
            {
                "effortEstimate": "Low",
                "entityId": "234ca9b6-3def-4ace-87c5-466c2e41f1d1",
                "name": "ex-pro-kotl-966",
                "recommendationType": "Downsize - Optimal Family",
                "savingsEstimate": "163.6333",
                "serviceType": "EC2",
                "divider": "-----------------------",
                "Department": "Sales",
                "awsAccount": "229132289155"
            },
            {
            // ... *SNIP* additional 96 system recommendations not displayed
*SNIP* ...
            }
        ]
    }
```

Postman Collection

Densify provides a Postman collection of sample API requests for working with your public cloud environments. See *Postman Collection* (Help Topic 340670)

Authorize

Description

The `authorize` resource is used to obtain a JSON Web Token (JWT) for Densify users to make authorized API requests. The resource will return a token for any active Densify user when the proper credentials are provided. Only API-enabled users can make authorized API calls. See Using the API: Authentication on page 17 for details on API-enabled users.

The JWT API token is only valid within the expiry time of the token, which by default is set to 60 minutes.

The key used to validate the token is also refreshed every 30 days.

> **Note:** To access the Densify API, you need user credentials with membership in the following Densify user group: *SaaS_User_Admin*.or *SaaS_User*.

Security Considerations

The authorization workflow provides a progressively longer delay each time an invalid password is entered in an Authorize request.

This behaviour is applied regardless of whether or not you have enabled the Densify password policy. The workflow does change if the password policy has been enabled. Contact Support@Densify.com for details.

Contact Support@Densify.com for details on configuring the Densify password policy.

Resource

```
/authorize
```

API Reference Guide for Public Cloud

Supported Operations

Table: Authorize Supported Operations

Operation	HTTP Method	Input	Output	Description
Obtain a JWT API token	`POST /authorize`	Request Body Parameters: ▪ userName ▪ pwd	▪ apiToken ▪ expires ▪ status	Used to return an API token for an active Densify user. Example: Successful Authorize Example: Unauthorized

Parameters

Request Body Parameters

Table: Authorize Request Body Parameters

Parameter Name	Type	Description
userName	string	The username of an active Densify user account. For API access, the Densify user must be part of the Administrator or SaaS_User user group. See Using the API: Authentication on page 17 for details on API-enabled users. The `/authorize` resource will return a token for any active Densify user with proper credentials provided, however, only tokens for API-enabled users can be used to make authorized API calls.
pwd	string	The corresponding password for userName.

Note: *Densify highly recommends using an SSL web service to ensure that user credentials are encrypted. Contact Support@Densify.com for details.*

Response

The following is a list of possible response elements that are returned for the `/authorize` resource. If authorization failed, two elements are returned:

▪ message

▪ status

Table: Authorize Response Schema

Element	Type	Sort By	Filter	Description
apiToken	string			The returned token used to make subsequent authorized API calls. The API token follows the JSON Web Token (JWT), RFC 7519 standard. Specify the **Bearer** authorization type with the token in the header of subsequent API requests: `Authorization: Bearer <apiToken>` See Example: Using Authorize JWT Token.
expires	number			The date and time (in milliseconds) when the apiToken expires.
status	number			The HTTP response code of the request. Possible status values include: ■ 200–successful response; ■ 400–the payload is null or invalid (i.e. userName or pwd is empty or invalid); ■ 401–authentication failed (e.g. user does not exist, incorrect password, or user account is locked). ■ 403–forbidden access. The trial or subscription has expired. ■ 429–requests are too frequent. ■ 500–internal server error.
message	string			The message for an error status response. See the row above, for details

> **Note:** *Other response status and error messages could indicate issues with the Densify web server or connectivity issues to the web server.*

Examples

Example: Successful Authorize

The following example shows you how to obtain an API token using the `apiUser` account.

Example: Successful Authorize

Request:

```
POST /authorize
{
    "userName": "apiUser",
    "pwd": "apiPassword"
}
```

Response:

```
{
  "apiToken": "eyJh-
bGciOiJIUzUxMiJ9.eyJqdGkiOiIzNzI2Yzk0NC0wMmE4LTRlYzQtOGE2Ny04ODBmMDM2OTRhZD-
ciLCJpYXQiOjE1NDI2NTI0MDUsInN1YiI6InZh-
biIsIm-
lzcyI6IkRlb-
nNpZnkuY29tIiwiZXhwIjoxNTQyNjUyNzA1fQ.cJd8qFJfRoPnMEU7GzcdYGBT8WwlgmviQ1OQp8P_
w9VUcjQA3FJaB9QkqJJ6d7zbrY5yjc4w0rOWjY-PPdbmqw",
  "expires" : 1542652705869,
  "status" : 200
}
```

Example: Unauthorized

The following example shows an authorize request with invalid user credentials.

Example: Unauthorized

Request:

```
POST /authorize
{
    "userName": "APIUser",
    "pwd": "wrongPassword"
}
```

Response:

```
{
  "message" : "Unauthorized",
  "status" : 401
}
```

Example: Using Authorize JWT Token

The following example shows you how to use the API token obtained from the `/authorize` resource to make an authorized API request. In this example, an API request is made to list all cloud analyses in Densify using the JWT token for authorization, which is passed to the header as a Bearer token authorization type.

Example: Using JWT API token to Make Authorized Request to List All Cloud Analyses

Request:

```
GET /analysis/cloud
```

Headers:

```
Accept: application/json
Authorization: Bearer eyJh-
bGciOiJIUzUxMiJ9.eyJqdGkiOiIzNzI2Yzk0NC0wMmE4LTRlYzQtOGE2Ny04ODBmMDM2OTRhZD-
ciLCJpYXQiOjE1NDI2NTI0MDUsInN1YiI6InZh-
biIsIm-
lzcyI6IkRlb-
nNpZnkuY29tIiwiZXhwIjoxNTQyNjUyNzA1fQ.cJd8qFJfRoPnMEU7GzcdYGBT8WwlgmviQ1OQp8P_
w9VUcjQA3FJaB9QkqJJ6d7zbrY5yjc4w0rOWjY-PPdbmqw
```

> **Note:** The long JWT token string is taken from the `apiToken` element of an `/authorize` request. This example shows the raw token string, but typical implementations would save the API token as a variable and pass it to the request header. See *Example: Returning AWS Recommendations with Low Effort in Terraform-map Form*.

Ping API

Description

The `/ping` resource is used to check the status of the Densify API subsystem. No credentials are required for this resource to validate the API subsystem. A ping API call performs an end-to-end test from the client to the API subsystem, ensuring that the Densify database is available and is able to respond to requests. Network elements, such as load balancers, can use this resource to verify that the web server receiving the request can be kept as an active participant in the resource pool.

Resource

```
/CIRBA/api/ping
```

Supported Operations

Table: Ping API Supported Operations

Operation	HTTP Method	Input	Output	Description
Ping the Densify API subsystem	GET /CIRBA/api/ping	Query String Parameter: ▪ timeout *(optional)*	▪ message ▪ status	Used to return the status of a connectivity test to the Densify API subsystem with no authorization. See Response Elements for the returned output. Example: Successful Ping API

Parameters

Query String Parameters

Table: Ping API Query String Parameters

Parameter Name	Type	Description
timeout *(optional)*	integer	The maximum time, in seconds, for the Densify API Server to respond before returning a timeout error. If the timeout parameter is not specified, then the default timeout is set to 30 seconds. The valid range of timeout is between 15 and 180 seconds. Specifying any value outside of the valid range will produce a malformed request error. Example: Ping API Timeout

Response

Response Elements

The following table list elements and possible status codes returned from the `/ping` request.

Element	Type	Description
message	string	Detailed message of the status response.
status	errorCode	HTTP Response Code Possible response code values of the `/ping` request include: - 200—Successful response: API subsystem is responding to requests; - 400—The provided timeout value was invalid (outside the range of 15 - 180 seconds). - 500—There is an issue connecting to the Densify API subsystem. Contact Densify support Support@Densify.com for further details.

Note: *Other response error messages could indicate issues with the Densify web server or other connectivity issues to the web server.*

Examples

Example: Successful Ping API

The following example shows you a ping request to the Densify API subsystem with a timeout threshold of 100 seconds.

Example: Successful Ping API

Request:

```
GET /CIRBA/api/ping?timeout=100
```

Response:

```
{
  "message" : "ok",
  "status" : 200
}
```

Example: Ping API Timeout

The following example shows you a ping request with an invalid timeout value.

Example: Ping API Timeout

Request:

```
GET /CIRBA/api/ping?timeout=10
```

Response:

```
{
  "message" : "Timeout value valid range: 15-180",
  "status" : 400
}
```

Analysis: AWS Analyze

Description

The `/analysis/cloud/aws/analyze` resource is used to collect AWS CloudWatch data and initiate optimization analysis with the cloud infrastructure collected. Below are the series of processes that occur when the initial `/analysis/cloud/aws/analyze` request is triggered:

1. Set up and initiate data collection of the specified AWS account and schedule it to run automatically on a nightly basis.

2. Initiate analysis on the data collected using the default policy.

   - Subsequent analysis is scheduled to run on a nightly basis after the completion of data collection.

   - Optionally, you can configure the results to be sent to a webhook URI upon analysis completion. See Add webhook to an analysis on page 226 for details.

3. While data collection or analysis is in progress, you can check the status (using `/analysis/AWS/<subscriptionId>/status` resource) or wait for the results to be published to an optional webhook URI.

4. The reporting database update is scheduled to run automatically on a nightly basis after the completion of the analysis. This process produces reports for each instance recommendation, which is useful for analysts or application owners. These reports are only created after the scheduled analysis is completed, and may therefore only be available on the following day for a new analysis. Exact timing depends on the size of your environment.

Ad-Hoc Tasks

Generally you do not need to run once-off tasks as both data collection and analysis tasks are scheduled automatically. In cases where you need make an ad-hoc request in addition to the scheduled job, the functionality exists for this endpoint.

Historical Data Collection

When Densify initiates data collection, normally audits collect only the last 24 hours of data. You can optionally collect up to 60 days of historical data. The historical data provides a more representative set of data on which to base resizing and optimization recommendations. You can run an ad-hoc task to collect the historical data.

> **Note:** Collection of historical data can take a significant amount of time, depending on the number of instances from which Densifyis collecting data. Contact Support@Densify.com to enable historical data collection and details of the performance impact.

The following settings define the range of historical data to be collected:

- Start date offset—This is number of days from the 60-day maximum, used to define the start of the range.

  These extra API parameters allow you to reduce the number of days of historical data to be collected. If, for example, the daily audit has been running for a few days before the historical audit can be executed then you can set the end offset to exclude the number of days that have already been collected. Sixty days is the maximum number of days that you can go back and collect historical data.

- End date offset—This is number of days from yesterday, to end the range of data collected.

Figure: Adjusting Historical Range Using Start and End Dates

A connection to the specified cloud account must already exist before you can run an ad hoc audit. When you execute an ad hoc refresh an audit task will be configured but a new connection will not be created. If the cloud connection does not already exist and the API POST contains `triggerAdhocAudit=true`, then you will get an error message.

If there is more than one account associated with the specified account ID (i.e. a payer account with many linked accounts), the Densify API handles it in the same way that analyses are currently rerun using the POST operation.

Once the audit is complete you need to rerun the associated analyses as indicated below or you can wait for the next scheduled execution of the analyses and RDB populate.

Analysis Update

You can make an ad-hoc request to refresh an existing analysis, outside of the scheduled nightly run using `/analysis/cloud/<aws|azure|gcp>/analyze`. This manual, ad hoc analysis request does not perform data collection or reporting database (RDB) updates. It only runs the analysis on the existing data collected with the following behavior:

- The analysis uses the policy that is configured for the analysis. Contact Support@Densify.com to change the configured policy.

- If a new webhook is provided, the analysis will send results to the new webhook URI. If no webhook is provided, the analysis will send results to the existing webhook, if configured.

- If the same analysis is already running, the request does not proceed and an appropriate message is returned.

- If the specified analysis has data collection scheduled within 30 minutes, the request does not proceed and an appropriate message is returned. For example, if data collection is scheduled to run at 12:05 AM, and you initiate a manual, ad hoc analyze request at 11:45 PM, then the analysis will not proceed and an error message is returned.

Prerequisite Configuration

Before you can collect AWS CloudWatch data, you need to create an IAM role for Densify to have a trust relationship with your AWS account that contains data. See *AWS Data Collection Prerequisites for an IAM Role* (Help Topic ID 410060) for details on how to set up the IAM role for data collection.

Resource

```
/analysis/cloud/aws/analyze
/analysis/cloud/aws
```

Supported Operations

Table: AWS Analyze Supported Operations

Operation	HTTP Method	Input	Output	Description
Run AWS data collection and analysis	POST /analysis/cloud/aws/analyze	Request Body Parameter: accountId roleArnName roleExternalId connectionName (optional) webHook (optional)	message status	This resource operation is used to: 1. Collect AWS cloud data connected via IAM role access. 2. Run analysis on AWS cloud data

Operation	HTTP Method	Input	Output	Description
				collected. 3. (Optional) Send results to webhook receiving application. 4. Schedule data collection and analysis processes each night subsequent to the initial request. Example: Running AWS Data Collection and Analysis
Re-run AWS data analysis	POST /analysis/cloud/aws/analyze	Request Body Parameter: ■ accountId ■ roleArnName ■ roleExternalId ■ webHook *(optional)*	■ href ■ message ■ status	This resource operation is used to re-run an analysis that already exists. You can specify an updated policy and/or webhook to use for the analysis. Data collection is not run. Data collection only occurs during the first `/analyze` request, and is then scheduled to run nightly The updated webhook is saved and will be used in the next scheduled analyses. You cannot initiate a request if data collection or the analyses are in progress or within 30 minutes of the time that these tasks are scheduled to run.
List all	GET /analysis/cloud/aws/	Path Parameter:	Lists all	This resource

Operation	HTTP Method	Input	Output	Description
generated analyses		◾ N/A Request Body Parameter: ◾ N/A	analyses that have been created with details.	operation is used to obtain the analysis ID that is required for other operations.
Run the 60-day historical AWS audit	POST /analysis/cloud/aws/analyze	Request Body Parameter: ◾ accountId ◾ roleArnName ◾ roleExternalId ◾ triggerAdhocAudit ◾ startDayOffset *(optional)* ◾ endDayOffset *(optional)*	◾ href ◾ message ◾ status	This resource operation is used to re-run an audit for which a connection and daily, scheduled audit already exists. You can optionally specify the number of days of historical data to collect. If not specified the previous 60 days from yesterday's date are collected. If you initiate an audit request when data collection or analysis is already running or within 30 minutes of the time that these tasks are scheduled to run, then the request will fail and an error message is returned.

Parameters

AWS Path Parameters

Table: AWS Analysis Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique referenced ID of the AWS analysis.

AWS Request Body Parameters

Table: AWS Analysis Request Body Parameters

Parameter Name	Type	Description
accountId	string	The AWS account ID with the CloudWatch data to collect. See AWS Data Collection Prerequisites for an IAM RoleAWS Data Collection Prerequisites for an IAM Role (Help Topic ID 410060) for details on setting up the ARN.
connectionName *(optional)*	string	Use the connection name to clearly identify this connection within Densify. This name will appear in the **Saved Connections** list in theDensify UI. By default, the connection name is set to the Subscription ID. The connection name must be unique within the Azure connection type section, so if the name is already in use, the request fails with an error message. This connection name can be used for filtering. *Note:* *The Connection Name is limited to 32-characters.*
endDayOffset *(optional)*	string	Historical data end day offset. This parameter is optional and is used to configure the range of the historical audit. It is used in conjunction with the parameters, `Trigger ad-hoc audit` and `Start Day Offset` to set the end day of the range of historical days of data to collect. If no value is specified and the parameter, `Trigger ad-hoc audit` has been enabled, then the end date is set to yesterday. If you specify any number other than 0, then that number is used to offset the range's end date from yesterday. i.e. if End Day Offset=5 and yesterday was Dec 1, then the end date will be Nov 25. When AFv2 is enabled, this setting is not used and the end date is always "yesterday". Do not pass this parameter in your API call.
roleArnName	string	The Amazon Resource Name (ARN) for the IAM role that you created in AWS to collect data. See *AWS Data Collection Prerequisites for an IAM Role* (Help Topic ID 410060) for details on setting up the ARN. To update the ARN, refer to Analysis: AWS Analyze.
roleExternalId	string	The external ID specified for Densify when the IAM role was created. See *AWS Data Collection Prerequisites for an IAM Role* (Help Topic ID 410060) for details on setting up

Parameter Name	Type	Description
		the ARN.
		To update the external ID, refer to Analysis: AWS Analyze.
startDayOffset *(optional)*	string	Historical data start day offset.
		This parameter is optional and is used to configure the historical audit. It is used in conjunction with the parameters, `Trigger ad-hoc audit` and `End Day Offset` to set the range of historical days of data to collect.
		If no value is specified and the parameter, `Trigger ad-hoc audit` has been enabled, then the start date is set to a date 60 days previous to yesterday's date.
		If you specify any number less than 60, then that number is used to offset the start date from 60 days. i.e. if Start Day Offset=20 then the start date will be 40 days previous to yesterday.
		You can use these setting to define a smaller range (i.e. 30 days). You cannot collect more than 60 days of historical data.
triggerAdhocAudit *(optional)*	string	The flag to trigger an ad-hoc 60-day historical audit.
		This parameter is optional and used to run the 60-day historical audit immediately and once only.
		Typically the 60-day historical audit is run first, when data collection is initiated for the specified AWS account. If this audit has been disabled for performance reasons, it can be run to collect the historical data as compute resources are available.
		A connection to the specified AWS account must already exist before you can use this flag. A once-off task will be configured and a new connection will not be created, if the connection does not already exist. If there is no existing connection AND the API post contains ad hoc=true, then you will see an error message.
webHook *(optional)*	■ uri ■ authType ■ authValue	The webhook definition to an external application.
		Optimization results are sent to the webhook-defined application when analysis is complete. See Parameters on page 228 for details of each parameter in the webhook definition.

Response

Table: AWS Analysis Response Schema

Element	Type	Filter/Sort	Description
href	string		The referenced resource to the analysis entity.

Element	Type	Filter/Sort	Description
			See Analysis: Entity on page 149 for details of the analysis entity resource. When a new analysis is requested from the `/analyze` resource, the entity ID will not be available until after data collection completes and the analysis entity is created.
message	string		The message for the status response is returned.
status	number		The HTTP response code of the request. Possible status values include: ■ 200–success with request; ■ 400–invalid parameters; ■ 401–authentication failed; ■ 404–resource not found; ■ 500–internal server error.

Examples

Example: Running AWS Data Collection and Analysis

The following example shows you how to initiate AWS data collection and analysis, and send the results to a WebHook.

Example: Running AWS Data Collection and Analysis

Request:

```
POST /analysis/cloud/aws/analyze
{
  "accountId": "188665225420",
  "roleArnName": "arn:aws:iam::188665225420:role/ReadOnlyAccess",
  "roleExternalId": "password",
  "webHook": {
      "uri": "http://mywebhookserver/webhook/results",
      "authType": "basic",
      "authValue": "tester:testerpassword"
  }
}
```

Response:

```
{
  "href": "Not available",
  "message": "OK",
  "status": 200
}
```

Example: Running the 60-Day Historical Audit

The following example shows you how to run an audit to collect 60 days of historical data.

Example: Running 60-Day Historical Audit

Request:

```
POST /analysis/cloud/aws/analyze
{
  "accountId": "188665225420",
  "roleArnName": "arn:aws:iam::188665225420:role/ReadOnlyAccess",
  "roleExternalId": "password",
  "triggerAdhocAudit": "Yes",
  "startDayOffset":"20" ,
  "endDayOffset":"10"
}
```

Response:

```
{
  "href": "/analysis/aws/209726931496/status",
  "message": "Analysis in progress",
  "status": 200
}
```

Analysis: AWS Delete

Description

The `/analysis/cloud/aws/<analysisId>` resource can be used to delete the audit and all associated scheduler entries of the cloud analysis specified by `<analysisId>`.

- The single-day and 60-day historical audits are removed, if they exist.
- The audit and all associated scheduler entries, are deleted, if they exist.

When executing the delete operation in an environment observe the following:

- When deleting an audit for an environment associated with a single account, the operation proceeds as outlined above.
- When deleting an audit for a payer account with multiple linked accounts, If Densify determines that there is a many-to-one relationship for this account, then the delete request fails. This type of multi-account connection can only be created through the Cloud Connection Wizard and must be deleted through the Cloud Connection Wizard.

During a delete operation, any attempt to access other associated endpoints related to the specified `<analysisId>` (i.e. GET, PUT, POST) will trigger a "400 Bad Request" error response.

While the delete operation is in progress, you can check the status (using the `/analysis/cloud/aws/<analysisId>/status` resource) or wait for the results to be published to an optional webhook URI.

Collected data is saved for the defined retention period and the environment and associated analysis structure are saved in case you want to generate reports on the historical data or restart data collection. Since the environment and analyses are not deleted, a GET call for the specified `<analysisId>` will always succeed regardless of whether the associated connection has been deleted.

If you use a GET call to list available environment and analyses, for which the audits have been deleted, they are still listed. You can reestablish the audit without have to re-create the analysis. You will still see the data in the Densify Console.

Resource

```
/analysis/cloud/aws/analysisId
```

Supported Operations

Table: AWS Supported Operations

Operation	HTTP Method	Input	Output	Description
Delete AWS data collection and analysis	`DELETE /analysis/cloud/aws/< analysisId>`	Request Body Parameter: ■ analysisId ■ webHook *(optional)*	■ message ■ status	This resource operation is used to delete audit and audit-related items for the specified analysis. Example: Deleting AWS Data Collection

Parameters

Path Parameters

Table: AWS Analysis Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique, referenced ID of the AWS analysis.

Request Body Parameters

Table: AWS Analysis Request Body Parameters

Parameter Name	Type	Description
webHook *(optional)*	■ uri ■ authType ■ authValue	The webhook definition to an external application. Status can be sent to the webhook-defined application when the delete operation is complete. See Parameters on page 228 for details of each parameter in the webhook definition.

Response

Table: AWS Analysis Response Schema

Element	Type	Filter/Sort	Description
message	string		The informational message returned with the status response.
status	number		The HTTP response code of the request. Possible status values include: ▪ 200–success with request; ▪ 400–invalid parameter; ▪ 401–authentication failed; ▪ 404–resource not found; ▪ 405–method not allowed; ▪ 500–internal server error.

Examples

Example: Deleting AWS Data Collection

The following example shows you how to delete the AWS audit and send the results to a WebHook.

Example: Deleting AWS Data Collection Audit

Request:

```
{
    DELETE /analysis/cloud/aws/fd9234a7b43c-7e4b-4084-8f46-b898c2c2

    "webHook": {
    "uri": "http://mywebhookserver/webhook/results",
    "authType": basic,
    "authValue": "tester:testerpassword"
                }
}
```

Response:

```
{
  "message": "OK",
  "status": 200
}
```

Example: Deleting an AWS Audit that is in Progress

The following example shows you the response if you attempt to delete an audit that is in progress.

Example: Deleting AWS Data Collection Audit

Request:

```
DELETE /analysis/cloud/aws/fd9234a7b43c-7e4b-4084-8f46-b898c2c2
```

Response:

```
{
   "message": "Connection has an audit in progress. It cannot be deleted at
this time.",
   "status": 500
}
```

Example: Deleting an AWS Payer Account Audit

The following example shows you the response if you attempt to delete an audit that has more than one account associated with the specified analysis ID.

If Densify determines that there is a many-to-one relationship for this connection, then the delete request fails. This type of multi-subscription connection can only be created through the Cloud Connection Wizard and must be deleted through the Cloud Connection Wizard.

Example:Deleting a Many-to-1 Data Collection Audit

Request:

```
DELETE /analysis/cloud/aws/aws/fd9234a7b43c-7e4b-4084-8f46-b898c2c2
```

Response:

```
{
    "message": "Account dfc04848-3848-44c0-b85a-02311951de36 was created via UI
and subscriptions [cc377154-9605-4cb0-8b41-1b39e1c4ac0f,3d4ba999-cbd8-40b8-
9998-574be6824a97] are incompatible with API use; please delete via UI",
   "status": 500
}
```

Analysis: AWS Recommendations

Description

The `/analysis/cloud/aws/<analysisId>/results` resource is used to return a collection of AWS system (i.e. EC2 instance, Auto Scaling group, etc.) recommendations for an AWS account defined by a Densify Analysis entity (see *Analysis: Entity* (Help Topic ID 340610)).

For each system recommendation, you can also download a PDF version of the Impact Analysis and Recommendation Report, which details system impact based on the recommendations. This report is useful to application owners and other system stakeholders who need to review recommended changes to their application resources. See the rptHref resource element for details.

To return a collection of all systems included in an AWS optimization analysis, see Analysis: AWS Systems on page 106.

To see the catalog map for a selected instance, see Systems: Analysis Details on page 265

Resource

```
/analysis/cloud/aws/<analysisId>/results
```

Supported Operations

Returned recommendation details can be in either JSON or Terraform-map format. Specify your preference using `application/json` in the request header.

Table: AWS Recommendations Supported Operations

HTTP Method	Input	Output	Description
GET /analysis/cloud/aws/ *<analysisId>* /results	Path Parameter: • analysisId Query String Parameter Options: • Element Filters • includeAttributes • dataQuality Accept: • application/json	Collection of details in JSON: • accountIdRef • approvalType • attributes • auditInfo • currentCost • currentHourlyRate • currentRiCoverage • currentType • dataQuality • deferRecommendation • deferUntil • densifyPolicy • entityId • effortEstimate • implementationMethod • name • powerState • predictedUptime • recommendationType • recommendedCost • recommendedHostEntityId • recommendedHourlyRate • recommendedType • recommFirstSeen • recommLastSeen • recommSeenCount • region • resourceId • rptHref • savingsEstimate • serviceType • totalHoursRunning Collection of ASG-specific details included in the JSON: • avgInstanceCountCurrent • avgInstanceCountRecommended • currentDesiredCapacity • minGroupCurrent • minGroupRecommended • maxGroupCurrent • maxGroupRecommended	Returns a collection of recommendations for the specified analysis. Specify `application/json` in the request header for recommendations to be returned in JSON format. Example: Returning EC2 Instances with Upsize Recommendations
GET /analysis/cloud/aws/	Path Parameter: • analysisId	Collection of details in Terraform-map format:	Returns a collection of recommendations for an AWS analysis.

HTTP Method	Input	Output	Description
`<analysisId>` `/results`	Query String Parameter Options: • Element Filters Accept: • application/terraform-map	• provisioningId—This provides the label for each terraform-map recommendation. • approvalType • currentType • deferRecommendation • deferUntil • densifyPolicy • effortEstimate • implementationMethod • powerState • predictedUptime • recommendationType • recommendedType • savingsEstimate Collection of ASG-specific details included in the terraform-map: • avgInstanceCountCurrent • avgInstanceCountRecommended • minGroupCurrent • minGroupRecommended • maxGroupCurrent • maxGroupRecommended	Specify `application/terraform-map` in the request header for recommendations to be returned in Terraform-map format. Example: Returning AWS Recommendations with Low Effort in Terraform-map Form Example: Returning an ASG Recommendation in Terraform-map Form

Parameters

Path Parameters

Table: AWS Recommendations Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique reference ID of the AWS analysis. When the account grouping feature is enabled and you are working through the Densify API, with the cloud accounts that are part of a group, the workspace ID is replaced by the grouped account's primary key or a unique value that is used as the analysisId or as the ID in the href, that is used to obtain the Impact Analysis and Recommendation Report.

Query String Parameters

Table: AWS Recommendation Query String Parameters

Parameter Name	Type	Description
Element Filters	string	You can use element filters to return a specific subset of the

Parameter Name	Type	Description
		recommendations. See the "F" (filter) designation in the Response schema table for a list of elements that support filtering. Refer to Filters on page 20 for a complete description of this common operation feature. Usage example: `../results?recommendationType=Upsize`
includeAttributes	▪ true ▪ false	Indicate whether or not to return system attributes: ▪ true—returns all of the system attributes; ▪ false—(default) suppress system attributes from the response output. Usage example: `../results?includeAttributes=true`
dataQuality	string	Allows you to indicate which workload type data collection detail to return for the system. See dataQuality for the returned details. Specify a quoted list of workload type names for which data to be returned, separated by commas. Contact Support@Densify.com for a listing of the supported workload type names. Usage example: `../results?dataQuality="CPU Utilization (CINT2006 Rate),Memory Utilization in Percent"` *Note:* *A "400-Bad Request" error message is returned if a non-supported workload type is specified.*

Response

Table: AWS Recommendation Response Schema

Element	Type	Filter/Sort	Description
entityId	string	F	The Densify assigned entity ID of the cloud system.
resourceId	string	F	The AWS identifier assigned to the system.
accountIdRef	string	F	The AWS account identifier.
region	string	F	The region containing the AWS instance.
currentType	string		The current instance type.
recommendationType	string	F	The recommended action for the system. This is also known as the Optimization Type in the Densify Console (see *Optimization Type Color-Coding Summary* in the topic *Understanding the Instance Optimization Details Report* (Help

Element	Type	Filter/Sort	Description
			Topic ID 380390)).
			The following types of recommended actions are supported for this cloud platform:
			▪ `"Just Right"`—this instance is optimally sized for the workload;
			▪ `"Upsize - Optimal Family"`—this instance should be upsized to a more optimal instance family;
			▪ `"Upsize"`—this instance should be upsized to an instance within the same instance family;
			▪ `"Terminate"`—this instance should be terminated;
			▪ `"Downsize - Optimal Family"`—this instance should be downsized to an instance belonging to a more suited instance family;
			▪ `"Downsize"`—this instance should be downsized to an instance within the same instance family;
			▪ `"Modernize - Optimal Family"`—this instance should be modernized to an instance belonging to a more optimal instance family;
			▪ `"Modernize"`—this instance should be modernized to an instance within the same instance family.
			▪ `"Not Analyzed"`—this instance has no recommendation due to insufficient workload information.
			For ASGs, additional recommendation actions are supported:
			▪ `Upscale`—increase compute capacity by adjusting the maximum group size;
			▪ `Downscale`—decrease compute capacity by adjusting the minimum group size.
			Use *recommendationType* as a filter to retrieve only system actions of interest. See Example: Returning AWS Systems with No Recommendations.
			Systems with insufficient information for analysis do not have a recommendation and may be

Element	Type	Filter/Sort	Description
			returned with limited elements.
recommendedType	string	F	The recommended instance type after Densify optimization analysis.
implementationMethod	string	F	[Self Optimization|Manual|N/A] Specifies whether this system is configured for Self-Optimization or Manual actioning based on the recommended action (recommendationType) and on the Self-Optimizing Automation policies. Note: Self-Optimization is not supported for ASGs with more than one EC2 instance, retuned for these ASGs.
predictedUptime	percentage		The predicted uptime (%) for the system is based on the percentage of hours CPU utilization data is present in the workload range specified in the policy settings. Predicted uptime % for new systems started mid-way within the workload range is calculated from the time/date that the system was started, as opposed to the beginning of the interval resulting, in more accurate prediction for the future.
totalHoursRunning	string		The total hours that the system has been running, in the workload range. Total hours for a scale group is the total time the scale group has been running and does not include cumulative hours for all of the in-service instances.
name	string	F	The name of the AWS system. Typically, this is the "Provisioning Id" AWS user tag value. If the "Provisioning Id" value is not set, then the AWS name assigned to the system is used instead.
rptHref	string		The reference resource to the Impact Analysis and Recommendation Report (also known as the Impact Analysis and Recommendation Report report). Use this reference to downoad the report. This report for the specified instance is available for download after the reporting database tables have been updated (i.e. after RDB populate has

Element	Type	Filter/Sort	Description
			been executed). By default, the reporting database tables are updated once every night.
			You need to use the following in the request header to download the PDF file: `Accept : application/octet-stream`.
			Note: *HTTPS must be enabled to download the Impact Analysis and Recommendation Report PDF.*
			See Viewing the Impact Analysis and Recommendation Report *Viewing the Impact Analysis and Recommendation Report* (Help Topic ID 380450) for details on the content of the report.
			Example: Downloading an Impact Analysis and Recommendation Report
approvalType	string	F	The approval setting for the system recommendation. The value in this string is derived from the Self-Optimizing Automation policies in implementationMethod and the approval attribute `attr_ApprovalSetting`. Possible approval settings include: • `"na"`– not approved; • `"all"`–approve any change; • `"<recommended-instance-type>"`–approve changing the instance to the specified *<recommended-instance-type>*. This field is not applicable to Auto Scaling groups with maximum group size greater than one. For these systems, the approvalType will always be "na" (not approved).
densifyPolicy	string	F	The Densify policy used for optimization analysis.
savingsEstimate	string		The value of savingestimate output parameter is the difference between the current and recommended instance type cost (this is the catalog cost). When using the API, the predicted uptime is NOT taken into consideration (i.e. [currentCost -

Element	Type	Filter/Sort	Description
			recommendedCost]). The Impact Analysis and Recommendation Report report uses the predicted uptime % when calculating estimated savings regardless of whether the report is obtained through the UI or via API.
			See *FAQs-Cloud* (Help Topic ID 400000) for examples.
			This calculation also applies to ASGs with maximum group size = 1. In this case, the ASG is considered an EC2, with no ASG elements exposed.
			For ASGs with maximum group size greater than 1, the savings estimate is the difference between the current and recommended instance type cost (i.e. catalog cost), with the average group in-service instances taken into consideration (i.e. [Analysis: AWS Recommendations * currentCost] - [Analysis: AWS Recommendations* recommendedCost]).
effortEstimate	string	F	[Moderate|Low|Very Low|None|Impossible]
			This element describes the effort required to investigate and implement the Densify recommendations. Effort for each system is calculated by rule-driven analytics based on factors (such as instance family change, storage change, data quality checks, feature changes, etc.) that can be configured in the policy settings and rule sets which capture best practices.
			"Impossible" effort is a result of a manual override for the instance.
			If a system is not 'Not Analyzed', Densify does not return an effortEstimate.
			Note: *When using the Subscription API, Densify returns an effortEstimate of "impossible" for systems that are 'Not Analyzed'.*
powerState	string	F	The power state of the system.
			The powerState details are displayed for EC2, RDS and ASG

Element	Type	Filter/Sort	Description
			instances via the API.
			Power state is not shown in the UI.
recommendedHostEntityId	string	F	The Densify entity ID of the catalog instance for the recommended instance type.
currentCost	string		The cost of the existing instance type (i.e. the instance catalog cost).
recommendedCost	string		The cost of the recommended instance type (i.e. the instance catalog cost) after Densify optimization analysis.
serviceType	string	F	The AWS service type: • EC2 • RDS • ASG • SPOT Scale groups with maximum group size of 1 (i.e. maxGroupCurrent=1) have serviceType=ASG. All other returned elements are similar to those for an EC2 service (i.e. min/max group size values are not returned).
currentHourlyRate	string		The hourly rate for the current instance type (i.e. instance catalog cost / monthly hours). This value is rounded to the nearest penny.
recommendedHourlyRate	string		The hourly rate for the recommended instance type (i.e. instance catalog cost / monthly hours). This value is rounded to the nearest penny.
currentRiCoverage	integer		This is the percentage of reserved instance (RI) coverage available for the current instance (or scale group). Flexible RIs are taken into consideration when calculating the percentage of RI coverage. However, convertible RIs are included in the calculation depending on the AWS Defer Recommendation settings. For individual compute or database instances, the coverage percentage is either 100% or 0%. Coverage between 0 and 100% occurs when Flexible RIs are converted into partial coverage. For scale groups, the percentage of coverage is based on the RIs available for in-service instances.

Element	Type	Filter/Sort	Description
			Note: *This element is returned for* *serviceType* *= EC2, RDS, and ASG.*
deferRecommendation	string		[yes|no] Defer recommending an instance type change until reserved instance coverage for this instance expires. Values include: • yes–defer instance type recommendation due to RI coverage; • no–do not defer instance type recommendation. If "*AWS - Defer Recommendation - Exclude Upsize Recommendation*" setting is TRUE (from Densify configuration settings), then deferRecommendation = no, for Upsize recommendations, even if currentRiCoverage is 100%. *Note:* *This element is returned for serviceType = EC2, RDS, and ASG. This element is not returned when recommendationType is "Just Right" or "Terminate".*
deferUntil	string		If deferRecommendation = "yes", then the expiry date of the RI providing coverage for the current instance is returned. The expiry date and time is in milliseconds. *Note:* *This element is only returned when deferRecommendation = "yes".*
minGroupCurrent	integer		Specifies the current minimum group size of the ASG. *Note:* *This element is only returned for ASGs with maximum group size greater than one.* Example: Returning an ASG Recommendation in Terraform-map Form
minGroupRecommended	integer		Specifies the recommended minimum group size for the Auto Scaling group.

Element	Type	Filter/Sort	Description
			Note: *This element is only returned for ASGs with maximum group size greater than one.* Example: Returning an ASG Recommendation in Terraform-map Form
maxGroupCurrent	integer		Specifies the current maximum group size of the Auto Scaling group. *Note:* *This element is only returned for ASGs with maximum group size greater than one.* Example: Returning an ASG Recommendation in Terraform-map Form
maxGroupRecommended	integer		Specifies the recommended maximum group size for the Auto Scaling group. *Note:* *This element is only returned for ASGs with maximum group size greater than one.*
currentDesiredCapacity	integer		Specifies the desired capacity currently configured for the Auto Scaling group. *Note:* *This element is only returned for ASGs with maximum group size greater than one.*
avgInstanceCountCurrent	float		The average of in-service instances over the historical interval (or workload range). This interval is defined by your policy (e.g. over the last 60 days). This element is only returned for scale groups with maximum group size greater than one.
avgInstanceCountRecommended	float		The predicted average instance count if the scale group recommendations are implemented. This element is only returned for scale groupss with maximum group size greater than one. Example: Returning an ASG Recommendation in Terraform-map Form

Element	Type	Filter/Sort	Description

scalingPolicies	string		This value indicates whether the scaling mode needs to be reviewed. Possible values are:
			
			▪ No Value-If the optimization type = Terminate, then:
			  • In the UI, a dash, "–" is displayed.
			  • In the API,"Terminate" is returned by the `/results` endpoint.
			▪ Keep–If the current average group size, minimum and maximum group size are not changed.
			▪ Adjust–The current average group size is different than the predicted average group size AND the instance type, minimum and maximum group size do not change.
			▪ Review–If Densify recommends changing one or more of instance type, average, minimum or maximum group size.

provisioningId	string		This element is used to identify the terraform-map recommendations for a unique system and corresponds to the "Provisioning Id" AWS user tag value. The "Provisioning Id" user tag is used to uniquely identify a system, since its resourceId or system name could possibly change after an instance type update. See Example: Returning AWS Recommendations with Low Effort in Terraform-map Form. If the "Provisioning Id" user tag value is not set, then the system name is used to identify the recommendations. *Note: This element is only returned for terraform-map responses.*
attributes	array of ▪ id ▪ name ▪ value		System attributes are properties set during the data collection process by a vendor platform (i.e. AWS) or by Densify for analytics. *Note: The attribute array is only returned when includeAttributes=true is in the query string.*

Element	Type	Filter/Sort	Description
recommFirstSeen	Unix time (in milliseconds)		The first date and time the recommended instance type (i.e. the recommendedType element) was provided by Densify (Unix Epoch time, in milliseconds).
recommLastSeen	Unix time (in milliseconds)		The latest date and time the recommended instance type (i.e. the recommendedType element) was provided by Densify (Unix Epoch time, in milliseconds).
recommSeenCount	integer		The number of times Densify suggested the recommended instance type (i.e. the recommendedType element). This is the count of Densify analysis processes which produced the same recommended instance type from recommFirstSeen to recommLastSeen. This value is updated when the RDB populate task (i.e. the reporting database update process) is executed. The RDB populate task compares the current recommendedType with the new recommendedType to update the recommSeenCount counter. **Note:** *In typical production environments, where the RDB populate task is scheduled to run once daily post data collection and analysis, the* recommSeenCount *value will reflect exactly the number of times the recommended instance type was provided. If your environment executes the RDB populate task more than once daily, the* recommSeenCount *value will be inflated beyond the actual number of times the instance type was recommended. Contact Support@Densify.com if you have concerns about the* recommSeenCount *value.*

auditInfo	■ dataCollection:
	  • dateFirstAudited
	  • dateLastAudited
	  • auditCount
	■ workloadDataLast30:
			The following system data collection details are returned:
			■ dateFirstAudited—the first time data was collected for this

Element	Type	Filter/Sort	Description
	- firstDate - lastDate - totalDays - seenDays		system (Unix epoch time, in milliseconds); ■ `dateLastAudited`—the most recent data collection time (Unix epoch time, in milliseconds); ■ `auditCount`—the number of times that data was collected. The following system workload collection details, for the last 30 days, are returned: ■ `firstDate`—the first time workload data was collected for this system (Unix epoch time, in milliseconds); ■ `lastDate`—the most recent workload data collection time for this system (Unix epoch time, in milliseconds); ■ `totalDays`—the difference in days between `firstDate` and `lastDate`; ■ `seenDays`—the number of days that at least one workload data was added into Densify for this system. *Note:* *If no workload data is collected for the system in the last 30 days, then the workloadDataLast30 element block is not returned.* *Note:* *The values in* `auditInfo` *are updated once a day, after the data collection and RDB populate processes are complete (i.e. the reporting tables have been updated with latest data collected).*
dataQuality	array of: ■ workloadName ■ firstSeen ■ lastSeen ■ completeDays ■ partialDays		The dataQuality array provides workload type data collection details for the system: ■ `workloadName`—the workload type name (see DCE Virtual Environment Workload Viewer to find a list of the supported workload types from the Analysis Console); ■ `firstSeen`—the first time this workload was collected (Unix Epoch time, in milliseconds);

Element	Type	Filter/Sort	Description
			▪ `lastSeen`—the most recent time this workload was collected (Unix Epoch time, in milliseconds); ▪ `completeDays`—the number of complete days that this workload data was collected; ▪ `partialDays`—the number of partial days that this workload data was collected. *Note:* *The dataQuality array is only returned when the* [dataQuality](#) *query string is specified in the request. In addition, if you specify a workload type in the request, for which system workload data does not exist in Densify, then no data is returned.* *Note:* *The values in the dataQuality array are updated after data collection and subsequent updates to the reporting tables (i.e. RDB Populate process) are completed.* See [Example: Returning EC2 Instances with Upsize Recommendations](#).

Examples

Example: Returning EC2 Instances with Upsize Recommendations

The following example shows you how to return a collection of AWS instances with "Upsize" recommendations. In addition, this example also returns the "CPU Utilization", and "Disk I/O Bytes" workload data collection details.

Example: Returning AWS Upsize Recommendations with Workload Data Collection Details

Request:

```
GET /analysis/cloud/aws/7abb627d-48db-4520-9e90-f46946ea6a24/res-
ults?recommendationType=Upsize&dataQuality="CPU Utilization,Disk I/O Oper-
ations"
```

Response:

```
[
    {
            "entityId": "49fb3629-1f2b-4039-b146-918eb8009184",
            "region": "us-east-1",
            "currentType": "t3.micro",
            "recommendationType": "Upsize",
            "recommendedType": "t3.small",
            "implementationMethod": "Manual",
            "predictedUptime": 92.29,
            "totalHoursRunning": 1329,
            "totalHours": 1440,
            "name": "SQL Express",
            "rptHref": "/systems/49fb3629-1f2b-4039-b146-918eb8009184/ana-
lysis-report",
            "approvalType": "na",
            "densifyPolicy": "Initial Assessment",
            "savingsEstimate": -5.4850235,
            "effortEstimate": "Very Low",
            "powerState": "Running",
            "recommendedHostEntityId": "9a67bfc4-7d30-4e75-9c4d-1ca501b0c4fd",
            "currentCost": 14.31,
            "recommendedCost": 28.62,
            "serviceType": "EC2",
            "currentHourlyRate": 0.02,
            "recommendedHourlyRate": 0.04,
            "currentRiCoverage": 0.0,
            "recommFirstSeen": 1579680587657,
            "recommLastSeen": 1589008760293,
            "recommSeenCount": 24,

        "auditInfo": {
            "dataCollection": {
                "dateFirstAudited": 1571949272133,
                "dateLastAudited": 1588997132907,
                "auditCount": 103
            },
            "workloadDataLast30": {
                "firstDate": 1687009600000,
                "lastDate": 1688910400000,
                "totalDays": 23,
                "seenDays": 7
            }
        },
        "dataQuality": [
            {
                "workloadName": "CPU Utilization",
                "firstSeen": 1666705600000,
                "lastSeen": 1688910400000,
```

```
                "completeDays": 161,
                "partialDays": 0
        },
        {
                "workloadName": "Disk I/O Operations",
                "firstSeen": 1666705600000,
                "lastSeen": 1673448400000,
                "completeDays": 78,
                "partialDays": 1
        }
    ]
}
...
]
```

Example: Returning ASG Downscale Recommendations

The following example shows you how to return ASG instances with "Downscale" recommendations.

Example: Returning ASG Downscale Recommendations

Request:

```
GET /analysis/cloud/aws/48db7abb627d-7abb-5602-9e9049f-
b36296a37/results?recommendationType=Downscale"
```

Response:

```
[
    {
                "entityId": "8c4719f5-0813-4943-82a8-9183c5994e6d",
                "resourceId": "8c22f5fb-8ed0-4ecd-ac91-8aa8e2f3856a",
                "accountIdRef": "209726931496",
                "region": "us-east-1",
                "currentType": "t2.micro",
                "recommendationType": "Downscale",
                "recommendedType": "t2.micro",
                "predictedUptime": 92.29,
                "totalHoursRunning": 1329,
                "totalHours": 1440,
                "name": "ec2containerservice-cluster-ecs-asg-k1yl0k4m",
                "rptHref": "/systems/8c4719f5-0813-4943-82a8-9183c5994e6d/ana-
lysis-report",
                "approvalType": "na",
                "densifyPolicy": "Initial Assessment",
                "savingsEstimate": 8.469575,
                "effortEstimate": "None",
                "powerState": "Running",
                "recommendedHostEntityId": "56b8437a-2eed-41e6-bd50-4d4734982785",
                "currentCost": 8.47,
```

```
                    "recommendedCost": 8.47,
                    "serviceType": "ASG",
                    "currentHourlyRate": 0.01,
                    "recommendedHourlyRate": 0.01,
                    "currentRiCoverage": 0.0,
                    "minGroupCurrent": "0",
                    "minGroupRecommended": "0",
                    "maxGroupCurrent": "4",
                    "maxGroupRecommended": "4",
                    "currentDesiredCapacity": "2",
                    "avgInstanceCountRecommended": 1.0,
                    "avgInstanceCountCurrent": 1.9999498,
                    "deferRecommendation": "no",
                    "recommFirstSeen": 1683131807753,
                    "recommLastSeen": 1686110400000,
                    "recommSeenCount": 10,
        }
        ...
    ]
```

Example: Returning AWS Recommendations with Low Effort in Terraform-map Form

The following example shows you how to return a collection of recommendations with Low effort in terraform-map form. The label of each recommendation (i.e. "asop-prepro-rdb-206", "ea-dev-asop-299" in the example below) is the provisioningId element.

Example: Return Low Effort Recommendations in Terraform-map

Request:

```
GET /analysis/cloud/aws/8bef9d74-94f7-414f-a032-5855258473a2/res-
ults?effortEstimate=Low
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Response:

```
densify_recommendations = {
  "asop-prepro-206" = {
    currentType = "t2.micro"
    recommendedType = "t3.micro"
    approvalType = "all"
```

```
      predictedUptime = "77.31"
      recommendationType = "Modernize"
      powerState = "Running"
      implementationMethod = "Self Optimization"
      savingsEstimate = "-34.31837"
      effortEstimate = "Low"
      densifyPolicy = "AWS General Prod"
      deferRecommendation = "no"
      minGroupCurrent = "0"
      minGroupRecommended = "0"
      maxGroupCurrent = "3"
      maxGroupRecommended = "3"
      avgInstanceCountCurrent = "0.99846673"
      currentDesiredCapacity = "0"
      avgInstanceCountRecommended = "0.99846673"
    }
    "ea-dev-asop-299" = {
      currentType = "c4.xlarge"
      recommendedType = "c5.xlarge"
      approvalType = "na"
      predictedUptime = "77.62"
      recommendationType = "Downsize"
      powerState = "Running"
      implementationMethod = "Self Optimization"
      savingsEstimate = "16.43216"
      effortEstimate = "Low"
      densifyPolicy = "AWS General Prod"
      deferRecommendation = "no"
    }
  ...
  }
```

Example: Returning an ASG Recommendation in Terraform-map Form

The following example shows you how to return a collection of ASG recommendations in terraform-map form.

Example: Return ASG Recommendations in Terraform-map

Request:

```
GET /analysis/aws/8bef9d74-94f7-414f-a032-5855258473a2/results?serviceType=ASG
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Response:

```
densify_recommendations = {
  "mobile-svc-asg-analysisGrid" = {
    currentType = "c4.large"
    recommendedType = "c5.large"
    approvalType = "na"
    predictedUptime = "81.53"
    recommendationType = "Downsize"
    powerState = "Running"
    savingsEstimate = "114.57644"
    effortEstimate = "Low"
    densifyPolicy = "AWS General Prod"
    minGroupCurrent = "4"
    minGroupRecommended = "2"
    maxGroupCurrent = "16"
    maxGroupRecommended = "8"
    avgInstanceCountCurrent = "2.9568965"
    currentDesiredCapacity = "4"
    avgInstanceCountRecommended = "1.6321839"
  }
  "pr000252-pro-bion-asg" = {
    currentType = "t2.xlarge"
    recommendedType = "c5.xlarge"
    approvalType = "na"
    predictedUptime = "67.23"
    recommendationType = "Downsize - Optimal Family"
    powerState = "Running"
    savingsEstimate = "302.53308"
    effortEstimate = "Low"
    densifyPolicy = "AWS General Prod"
    minGroupCurrent = "8"
    minGroupRecommended = "4"
    maxGroupCurrent = "12"
    maxGroupRecommended = "15"
    avgInstanceCountCurrent = "4.0170455"
    currentDesiredCapacity = "8"
    avgInstanceCountRecommended = "1.9479166"
  }
  ...
}
```

Example: Downloading an Impact Analysis and Recommendation Report

The following example shows you how to download a PDF Impact Analysis and Recommendation Report from the `rptHref` resource element provided in the instance recommendation output.

> **Note:** *HTTPS needs to be enabled to download the Impact Analysis and Recommendation Report PDF.*

Example: Download an Impact Analysis and Recommendation Report

Request:

```
GET /systems/7836335a-1942-4115-a65d-a298be1d390c/analysis-report
```

Headers:

```
Accept: application/octet-stream
Authorization: Bearer <apiToken>
```

Example: Returning AWS Systems with No Recommendations

The following example shows you how to return a collection of AWS systems without recommendations. These systems typically do not have adequate data for optimization analysis and have the "Not Analyzed" designation in the recommendationType element.

Example: Returning AWS Systems with No Recommendations in JSON

Request:

```
GET /analysis/cloud/aws/7abb627d-48db-4520-9e90-f46946ea6a24/res-
ults?recommendationType=Not Analyzed
```

Headers:

```
Accept: application/json
Authorization: Bearer <apiToken>
```

Response:

```
[
    {
        "entityId": "88bf8536-7f8e-4494-9930-5873ea982f7a",
        "resourceId": "i-06d390ca1a0c59d95",
        "accountIdRef": "555726931496",
        "region": "us-east-1",
        "currentType": "c4.large",
        "recommendationType": "Not Analyzed",
        "name": "Mark333",
        "densifyPolicy": "Initial Assessment",
        "powerState": "Stopped",
        "currentCost": 79.84,
        "serviceType": "EC2",
        "currentRiCoverage": 0.0,
        "recommFirstSeen": 1579680587673,
        "recommLastSeen": 1589008760337,
        "recommSeenCount": 24,
```

```
        "auditInfo": {
            "dataCollection": {
                "dateFirstAudited": 1571949272133,
                "dateLastAudited": 1588997132907,
                "auditCount": 103
            }
        }
    },
    {
        "entityId": "ef37339b-d400-4a8c-a080-44e6bfd8db19",
        "resourceId": "db-pfnzbc3ahhaqtt5gyhi7h4lhbi",
        "accountIdRef": "555726931496",
        "region": "us-east-1",
        "currentType": "db.t2.micro",
        "recommendationType": "Not Analyzed",
        "name": "autoscale-1",
        "densifyPolicy": "Initial Assessment",
        "powerState": "Offline",
        "currentCost": 12.41,
        "serviceType": "RDS",
        "currentRiCoverage": 0.0,
        "recommFirstSeen": 1579680587673,
        "recommLastSeen": 1589008760337,
        "recommSeenCount": 24,
        "auditInfo": {
            "dataCollection": {
                "dateFirstAudited": 1571949272133,
                "dateLastAudited": 1588997132907,
                "auditCount": 103
            },
            "workloadDataLast30": {
                "firstDate": 1587009600000,
                "lastDate": 1588910400000,
                "totalDays": 23,
                "seenDays": 7
            }
        }
    },
    ...
]
```

Example: Returning AWS Systems with No Recommendations in Terraform-Map

Request:

```
GET /analysis/cloud/aws/7abb627d-48db-4520-9e90-f46946ea6a24/res-
ults?recommendationType=Not Analyzed
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Response:

```
densify_recommendations = {
  "Mark333" = {
    currentType = "c4.large"
    recommendationType = "Not Analyzed"
    powerState = "Stopped"
    densifyPolicy = "Initial Assessment"
  }
  "autoscale-1" = {
    currentType = "db.t2.micro"
    recommendationType = "Not Analyzed"
    powerState = "Offline"
    densifyPolicy = "Initial Assessment"
  }
  ...
}
```

Analysis: AWS Systems

Description

The `/analysis/cloud/aws/<analysisId>/systems` resource is used to return a collection of all systems included in an AWS optimization analysis.

The recommendations from an AWS optimization analysis can be obtained using the `/analysis/cloud/aws/<analysisId>/results` resource. See Analysis: AWS Recommendations on page 83 for details on the AWS recommendations resource.

The number of analyzed systems (i.e. `/aws/<analysisId>/systems` entities) will always be greater than or equal to the number of system recommendations produced (i.e. `/aws/<analysisId>/results` entities), as some systems may not have any recommendations.

Resource

```
/analysis/cloud/aws/<analysisId>/systems
```

Supported Operations

Table: AWS Systems Supported Operations

Operation	HTTP Method	Input	Output	Description
List all systems included in an AWS analysis	`GET /analysis/cloud/aws/ <analysisId>/systems`	Path Parameter: ▪ analysisId	Collection of: ▪ resourceId ▪ powerState ▪ currentType ▪ displayName ▪ serviceType	Use this resource to return a list of all systems included in the AWS analysis. Example: Listing All Systems in an AWS

Operation	HTTP Method	Input	Output	Description
			• entityId • href	Analysis

Parameters

Path Parameters

Table: AWS System Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique referenced ID of the AWS analysis.

Response

Table: AWS System Response Schema

Element	Type	Filter/Sort	Description
serviceType	string		The cloud service type (e.g. EC2, RDS, ASG, SPOT). **Note:** *Unlike the Densify Console, ASGs with maximum group size of 1 (i.e. maxGroupCurrent=1) will have serviceType=ASG. However, the returned elements are similar to those for an EC2 service (i.e. min/max group size values are not returned).*
resourceId	string		The AWS identifier assigned to the instance.
powerState	string		The power state of the instance.
currentType	string		The current instance type of the AWS system.
displayName	string		The AWS name assigned to the system.
entityId	string		The Densify-assigned entity ID of the AWS system.
href	string		The referenced resource to the system entity. See Systems on page 249 for details of the `/systems` resource.

Examples

Example: Listing All Systems in an AWS Analysis

The following example shows you how to return all systems (instances) included in an AWS optimization analysis.

Example: Listing all Systems in an AWS Analysis

Request:

```
GET /analysis/cloud/aws/8b5d2d56-6d85-4fde-8beb-fcd3cdf8e5b8/systems
```

Response:

```
[
    {
        "serviceType": "EC2",
        "resourceId": "i-bc1037760797bc103",
        "powerState": "Running",
        "currentType": "t2.medium",
        "displayName": "ex-prod-ecds-216",
        "entityId": "008a502d-0a1a-40d4-bb83-42413289fe1e",
        "href": "/systems/008a502d-0a1a-40d4-bb83-42413289fe1e"
    },
    {
        "serviceType": "EC2",
        "resourceId": "i-00bc71015587bc83",
        "powerState": "Running",
        "currentType": "m4.2xlarge",
        "displayName": "ex-dev-abds-131",
        "entityId": "0090c272-7e02-4dd5-b7f3-ad4bc225fbd2",
        "href": "/systems/0090c272-7e02-4dd5-b7f3-ad4bc225fbd2"
    },
    ...
]
```

Analysis: Azure Analyze

Description

The `/analysis/azure/analyze` resource is used to collect Microsoft Azure infrastructure data and initiate optimization analysis with the collected data. The following processes occur when the first `/analysis/azure/analyze` request is triggered:

1. Set up and initiate data collection of the specified Azure subscription and schedule it to run automatically on a nightly basis.

2. Initiate analysis on the data collected using the default policy.

   - Subsequent analysis is scheduled to run on a nightly basis after the completion of data collection.

   - Optionally, you can configure the results to be sent to a webhook URI upon analysis completion. See Add webhook to an analysis on page 226 for details.

3. While data collection or analysis is in progress, you can check the status (using `/analysis/azure/<subscriptionId>/status` resource) or wait for the results to be published to an optional webhook URI.

4. The reporting database update is scheduled to run automatically on a nightly basis after the completion of the analysis. This process produces reports for each instance recommendation, which is useful for analysts or application owners. These reports are only created after the scheduled analysis is completed, and may therefore only be available on the following day for a new analysis. Exact timing depends on the size of your environment.

The `/analysis/cloud/azure` resource is also used to return a list of Microsoft Azure optimization analyses currently in the Densify.

Ad-Hoc Tasks

Generally you do not need to run once-off tasks as both data collection and analysis tasks are scheduled automatically. In cases where you need make an ad-hoc request in addition to the scheduled job, the functionality exists for this endpoint.

Historical Data Collection

When Densify initiates data collection, normally audits collect only the last 24 hours of data. You can optionally collect up to 30 days of historical data. The historical data provides a more representative set of data on which to base resizing and optimization recommendations. You can run an ad-hoc task to collect the historical data.

> *Note:* *Collection of historical data can take a significant amount of time, depending on the number of instances from which Densify is collecting data. Contact Support@Densify.com to enable historical data collection and details of the performance impact.*

The following settings define the range of historical data to be collected:

- Start date offset–This is the number of days from the 30-day maximum, used to define the start of the range.

- End date offset–This is number of days from yesterday, to end the range of data collected.

These parameters allow you to reduce the number of days of historical data to be collected. If, for example, the daily audit has been running for a few days before the historical audit can be executed then you can set the end offset to exclude the number of days that have already been collected. Thirty days is the maximum number of days that you can go back and collect historical data for Azure and GCP environments.

A connection to the specified cloud account must already exist before you can run an ad hoc audit. When you execute an ad hoc refresh an audit task will be configured but a new connection will not be created. If the cloud connection does not already exist and the API POST contains `triggerAdhocAudit=true`, then you will get an error message.

If there is more than one account associated with the specified account ID (i.e. a payer account with many linked accounts), the Densify API handles it in the same way that analyses are currently rerun using the POST operation.

Once the audit is complete you need to rerun the associated analyses as indicated below or you can wait for the next scheduled execution of the analyses and RDB populate.

Analysis Update

You can make an ad-hoc request to refresh an existing analysis, outside of the scheduled nightly run using `/analysis/cloud/<aws|azure|gcp>/analyze`. This manual, ad hoc analysis request does not perform data collection or reporting database updates. It only runs the analysis on the existing data collected with the following behavior:

- The analysis uses the policy that is configured for the analysis. Contact Support@Densify.com to change the configured policy.

- If a new webhook is provided, the analysis will send results to the new webhook URI. If no webhook is provided, the analysis will send results to the existing webhook, if configured.

- If the same analysis is already running, the request does not proceed and an appropriate message is returned.

- If the specified analysis has data collection scheduled within 30 minutes, the request does not proceed and an appropriate message is returned. For example, if data collection is scheduled to run at 12:05 AM, and you initiate a manual, ad hoc analyze request at 11:45 PM, then the analysis will not proceed and an error message is returned.

Prerequisite Configuration

Before you can collect Azure cloud infrastructure data in Densify, you need to create a service principle and configure a secret key. See *Microsoft Azure Data Collection Prerequisites for a Service Principal* (Help Topic ID 410010) for details on creating and configuring the service principle.

> **Note:** When using the Densify API only the Azure "Service Principal" can be used to connect to your Azure subscriptions.

If you are using the API, data collection and analysis are created and then refreshed daily on a per subscription basis (1-to-1). You can associate many subscriptions with a service principle, but when using the API to initiate data collection, you must specify a subscription ID and the audit and analysis are created for each subscription, separately.

When using the Connection Wizard in the Densify UI, you do not need the subscription ID, as all subscriptions that are associated with the service principle are collected and listed once the connection has been verified. You can then select one or more of the subscriptions that you want to analyze (1-to-Many). When using the Connection Wizard, data collection and analysis are created and then refreshed daily for all of the subscriptions that you selected when you created the connection.

> **Note:** When using the Densify API only one subscription will be processed per API request. This is the case, even if more than one subscription is associated with the service principle.

Changing Credentials

When you need to change the credentials for the subscription or the service principle, you need to delete the data collection audit and recreate it. When you delete the audit, only the audit and all associated scheduler entries are removed, so you can recreate the audit with the new credentials and continue without any loss of data.

Resource

```
/analysis/cloud/azure/analyze
```

```
/analysis/cloud/azure
```

Supported Operations

Table: Azure Analysis Supported Operations

Operation	HTTP Method	Input	Output	Description
Run Azure data collection and analysis	POST /analysis/azure/analyze	Application ID Secret Key Tenant ID Subscription ID serviceAcctJSON (optional) connectionName (optional) webHook (optional)	href message status	Use this resource to: 1. Collect Azure cloud data via API for the specified subscription. 2. Run the analyses if data has been collected. An analysis is created for the specified subscription. 3. (Optional) Send results to webhook receiving application. 4. Data collection and analysis processes are scheduled to run nightly, after the initial request.
Rerun Azure data analysis	POST /analysis/azure/analyze	Application ID Secret Key Tenant ID Subscription ID webHook (optional)	href message status	This resource operation is used to re-run an analysis that already exists. You can specify an updated policy and/or webhook to use for the analysis. Data collection is not run. Data collection only occurs during the first /analyze request, and is then scheduled to run nightly The updated webhook is saved and will be used in the next scheduled analyses. You cannot initiate a request if data collection or the

Operation	HTTP Method	Input	Output	Description
				analyses are in progress or within 30 minutes of the time that these tasks are scheduled to run.
Run the historical audit	POST /analysis/cloud/ azure/analyze	Request Body Parameter: • Application ID • Secret Key • Tenant ID • Subscription ID • triggerAdhocAudit • startDayOffset (optional) • endDayOffset (optional)	• href • message • status	This resource operation is used to re-run an audit for which a connection and daily, scheduled audit already exists. You can optionally specify the number of days of historical data to collect. If not specified the previous 30 days from yesterday's date are collected. If you initiate an audit request when data collection or analysis is already running or within 30 minutes of the time that these tasks are scheduled to run, then the request will fail and an error message is returned.
List all generated analyses	GET /analysis/cloud/azure/	Path Parameter: • N/A Request Body Parameter: • N/A	Lists all analyses that have been created, with their details.	This resource operation is used to obtain the analysis ID that is required for other operations.

Parameters

Azure Path Parameters

Table: Azure Analysis Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique referenced ID of the Azure analysis.

Azure Request Body Parameters

Table: Azure Analysis Parameters

Parameter Name	Type	Description
subscriptionId	string	The subscription ID of the Azure subscription from which to collect data. *Note: When using the Densify API only one subscription will be processed per API request. This is the case, even if more than one subscription is associated with the service principle.* If this is a new subscription ID (i.e. has not been audited before), the post request will trigger the initial 30-day audit of historical data, create and enable a schedule for the daily audit and create the corresponding cloud environment and analyze the data once data collection completes. If this is subscription ID has been audited and analyzed previously, then the post request will trigger an adhoc environment analysis refresh.
applicationID	string	The ID of the application created within your Active Directory. This application is associated with the user that creates it. It also contains the subscription ID for reference by other methods. See *Microsoft Azure Data Collection Prerequisites for a Service Principal* (Help Topic ID 410010) for details on setting up the service principle.
secretKey	string	The client secret for the above listed application. See *Microsoft Azure Data Collection Prerequisites for a Service Principal* (Help Topic ID 410010) for details on setting up the service principle.
tenantID	string	The tenant ID corresponds to the Azure Active Directory (AD). See *Microsoft Azure Data Collection Prerequisites for a Service Principal* (Help Topic ID 410010) for details on setting up the service principle.
serviceAcctJSON	file	The credentials and details as listed above, provided in a single JSON file. To update the JSON file, refer to Analysis: Azure Analyze.
connectionName *(optional)*	string	Use the connection name to clearly identify this connection within Densify. This name will appear in the **Saved Connections** list in the Densify UI. By default, the connection name is set to the Subscription ID. The connection name must be unique within the Azure connection type section, so if the name is already in use, the request fails with an error message. This connection name can be used for filtering. *Note: The Connection Name is limited to 32-characters.*
endDayOffset *(optional)*	string	Historical data end day offset. This parameter is optional and is used to configure the range of the historical audit. It is used in conjunction with the parameters, `Trigger ad-hoc audit` and `Start Day Offset` to set the end day of the range of historical days of data to collect. If no value is specified and the parameter, `Trigger ad-hoc audit` has been enabled, then the end date is set to yesterday. If you specify any number other than 0, then that number is used to offset

Parameter Name	Type	Description
		the range's end date from yesterday. i.e. if End Day Offset=5 and yesterday was Dec 1, then the end date will be Nov 25.
		When AFv2 is enabled, this setting is not used and the end date is always "yesterday". Do not pass this parameter in your API call.
startDayOffset *(optional)*	string	Historical data start day offset.
		This parameter is optional and is used to configure the historical audit. It is used in conjunction with the parameters, `Trigger ad-hoc audit` and `End Day Offset` to set the start day of the range of historical days of data to collect.
		If no value is specified and the parameter, `Trigger ad-hoc audit` has been enabled, then the start date is set to 30 days previous to yesterday's date.
		If you specify a number less than 30, that number is used to offset the start date from 30 days in the past. i.e. if Start Day Offset=10 then the start date will be 10 days offset from 30 days, so 20 days previous to yesterday.
		You can use these setting to define a smaller range (i.e. 20 days). You cannot collect more than 30 days of historical data.
triggerAdhocAudit *(optional)*	string	The flag to trigger an ad-hoc 30-day historical audit.
		This parameter is optional and used to run the historical audit immediately and once only.
		Typically the historical audit is run first, when data collection is initiated for the specified subscription /project. If the historical audit has been disabled for performance reasons, it can be run once to collect the historical data as compute resources become available.
		A connection to the specified subscription or project must already exist before you can use this flag.
		A once-off task will be configured but a new connection cannot be created. If the connection does not already exist AND the API post contains ad hoc=true, then you will see an error message.
webHook *(optional)*	• uri • authType • authValue	The webhook definition to an external application.
		Optimization results are sent to the webhook-defined application when the analysis is complete. See Parameters on page 228 for details of each parameter in the webhook definition.

Response

Table: Azure Analysis Response Elements

Element	Type	Filter/Sort	Description
analysisName	string		The analysis name corresponds to the Azure subscription ID for the collected infrastructure data. An analysis is created for each subscription associated with the service principle provided.
href	string		The referenced resource to the recommendations of the analysis.

Element	Type	Filter/Sort	Description
			See Analysis: Azure Recommendations on page 122 for details on Azure analysis recommendations. **Note:** *The Impact Analysis and Recommendation Report report is not currently available for VM Scale Sets with maximum size >1.*
completed	string		The date and time (in milliseconds) when processing of the last analysis completed.
analysisId	string		The Densify internal ID of the analysis entity.
phase	string		The current phase of the specified analysis. Possible phases include: ■ analyzing : *<percent completed>%* ■ not analyzing
message	string		The message for the analysis status. For errors, the message for the following status response is returned.
status	number		The HTTP response code of the request. Possible status values include: ■ 200–success with request; ■ 400–invalid parameters; ■ 401–authentication failed; ■ 404–resource not found; ■ 500–internal server error.

Examples

Example: Creating New Azure Data Collection and Analysis

The following example shows you how to initiate Azure data collection and analysis, and send the results to a WebHook.

Example: Running Azure Data Collection and Analysis

Request:

```
POST /analysis/azure/analyze
{
  "subscriptionId": "cc377154-9605-4cb0-8b41-1b39e1c4ac0f",
  "applicationId": "bd6529bf-85d8-410d-a207-ce963b4dd398",
  "secretKey": "w6Dvtd5wncPv7aLIXtSkzcIaIU8Y+MLwlcGuoVmFq4c=",
  "tenantId": "6c9190a7-bca6-4fcd-b35e-36378aadc695",
  "connectionName": "AzureEnvTest"

  "webHook": {
      "uri": "http://mywebhookserver/webhook/results",
      "authType": "basic",
```

```
        "authValue": "test:testpassword"
    }
}
```

Response:

```
{
  "href": "Not available",
  "message": "Analysis in progress",
  "status": 200
}
```

Example: Refreshing an Existing Analysis

The following example shows you how to run an adhoc analysis using the last set of collected data, the default policy and sending the results to a WebHook.

Example: Running Analysis Refresh

Request:

```
POST /analysis/azure/analyze
{
  "subscriptionId": "cc377154-9605-4cb0-8b41-1b39e1c4ac0f",
  "applicationId": "bd6529bf-85d8-410d-a207-ce963b4dd398",
  "secretKey": "w6Dvtd5wncPv7aLIXtSkzcIaIU8Y+ML",
  "tenantId": "6c9190a7-bca6-4fcd-b35e-36378aadc695",
  "connectionName": "AzureEnvTest"

  "webHook": {
        "uri": "http://mywebhookserver/webhook/results",
        "authType": "basic",
        "authValue": "test:testpassword"
  }
}
```

Response:

```
{
  "href": "/analysis/cloud/azure/cc377154-9605-4cb0-8b41-1b39e1c4ac0f"
  "message": "OK",
  "status": 200
}
```

Analysis: Azure Delete

Description

The `/analysis/cloud/azure/<analysisId>` resource can be used to delete the audit and all associated scheduler entries of the cloud analysis specified by `<analysisId>`.

- The single-day and 60-day historical audits are removed, if they exist.

- The audit and all associated scheduler entries, are deleted, if they exist.

When executing the delete operation in an environment observe the following:

- When deleting an audit for an environment associated with a single account, the operation proceeds as outlined above.

- When deleting an audit, if Densify determines that there is a many-to-one relationship for this connection, then the delete request fails. This type of multi-account connection can only be created through the Cloud Connection Wizard and must be deleted through the Cloud Connection Wizard.

During a delete operation, any attempt to access other associated endpoints related to the specified `<analysisId>` (i.e. GET, PUT, POST) will trigger a "400 Bad Request" error response.

While the delete operation is in progress, you can check the status (using the `/analysis/cloud/azure/<analysisId>/status` resource) or wait for the results to be published to an optional webhook URI.

Collected data is saved for the defined retention period and the environment and associated analysis structure are saved in case you want to generate reports on the historical data or restart data collection. Since the environment and analyses are not deleted, a GET call for the specified `<analysisId>` will always succeed regardless of whether the associated connection has been deleted.

If you use a GET call to list available environment and analyses, for which the audits have been deleted, they are still listed. You can reestablish the audit without have to re-create the analysis. You will still see the data in the Densify Console.

Resource

```
/analysis/cloud/azure/analysisId
```

Supported Operations

Table: Azure Supported Operations

Operation	HTTP Method	Input	Output	Description
Delete Azure data collection and analysis	DELETE /analysis/cloud/azure/<analysisId>	Request Body Parameter: • analysisId • webHook (optional)	• message • status	This resource operation is used to delete audit and audit-related items for the specified analysis. Example: Deleting Azure Data Collection

Parameters

Path Parameters

Table: Azure Analysis Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique, referenced ID of the Azure analysis.

Request Body Parameters

Table: Azure Analysis Request Body Parameters

Parameter Name	Type	Description
webHook (optional)	• uri • authType • authValue	The webhook definition to an external application. Status can be sent to the webhook-defined application when the delete operation is complete. See Parameters on page 228 for details of each parameter in the webhook definition.

Response

Table: Azure Analysis Response Schema

Element	Type	Filter/Sort	Description
message	string		The informational message for the status response that is returned.
status	number		The HTTP response code of the request. Possible status values include: ▪ 200–success with request; ▪ 400–invalid parameter; ▪ 401–authentication failed; ▪ 404–resource not found; ▪ 405–method not allowed; ▪ 500–internal server error.

Examples

Example: Deleting Azure Data Collection

The following example shows you how to delete the audit and send the results to a WebHook.

Example: Deleting Data Collection Audit

Request:

```
{
    DELETE /analysis/cloud/azure/fd9234a7b43c-7e4b-4084-8f46-b898c2c2

    "webHook":       {
        "uri": "http://mywebhookserver/webhook/results",
        "authType": basic",
        "authValue": "tester:testerpassword"
                    }
}
```

Response:

```
{
  "message": "OK",
  "status": 200
}
```

Example: Deleting an Audit that is in Progress

The following example shows you the response if you attempt to delete an audit that is in progress.

Example: Deleting Data Collection Audit

Request:

```
DELETE /analysis/cloud/azure/fd9234a7b43c-7e4b-4084-8f46-b898c2c2
```

Response:

```
{
  "message": "Connection has an audit in progress. It cannot be deleted at
this time.",
  "status": 500
}
```

Example: Deleting an Aggregate Audit

The following example shows you the response if you attempt to delete an audit that has more than one account associated with the specified analysis ID.

If Densify determines that there is a many-to-one relationship for this connection, then the delete request fails. This type of multi-subscription connection can only be created through the Cloud Connection Wizard and must be deleted through the Cloud Connection Wizard.

Example: Deleting a Many-to-1 Data Collection Audit

Request:

```
DELETE /analysis/cloud/azure/fd9234a7b43c-7e4b-4084-8f46-b898c2c2
```

Response:

```
{
   "message": "Account dfc04848-3848-44c0-b85a-02311951de36 was created via UI
and subscriptions [cc377154-9605-4cb0-8b41-1b39e1c4ac0f,3d4ba999-cbd8-40b8-
9998-574be6824a97] are incompatible with API use; please delete via UI",
  "status": 500
}
```

Analysis: Azure Recommendations

Description

The `/analysis/cloud/azure/<analysisId>/results` resource is used to return a collection of Azure Virtual Machine instance recommendations after the optimization analysis has been performed on your collected Azure infrastructure data.

For each instance recommendation, you can also download a PDF version of the Impact Analysis and Recommendation Report, which details system impact based on the recommendation. This report is useful to application owners who need to review recommended changes to their application resources. See the rptHref resource element for details on how to download this report.

To return a collection of all instances included in an Azure optimization analysis, see .

To see the catalog map for a selected instance, see

> **Note:** When using the Densify API only one subscription is processed per analysis This is the case, even if more than one subscription is associated with the service principle. If the connection was created through the Cloud Connection wizard in the UI, all subscriptions that were selected, when the connection was created, and associated with the `<analysisId>` are returned.

Resource

```
/analysis/cloud/azure/<analysisId>/results
```

Supported Operations

Returned recommendation details can be in either JSON or Terraform-map format. Specify your preference using `application/json` in the request header.

Table: Azure Recommendations Supported Operations

HTTP Method	Input	Output	Description
GET /analysis/cloud/az ure/*<analysisId>* /results	Path Parameter: - analysisId Query String Parameter Options: - Element Filters - includeAttributes - dataQuality Accept: - application/json	Collection of (JSON): - accountIdRef - approvalType - attributes - auditInfo - currentCost - currentHourlyRate - currentType - dataQuality - densifyPolicy - effortEstimate - entityId - implementationMethod - name - powerState - predictedUptime - region - resourceId> - recommendationType - recommendedCost - recommendedHostEntityId - recommendedHourlyRate - recommendedType - recommFirstSeen - recommLastSeen - recommSeenCount - rptHref - savingsEstimate - serviceType - totalHours - totalHoursRunning Collection of VMSS-specific in JSON format: - autoScaling - avgInstanceCountCurrent - avgInstanceCountRecommended - minGroupCurrent - minGroupRecommended - maxGroupCurrent - maxGroupRecommended - orchestrationMode - profiles - scalingPolicies	Returns a collection of recommendations for an Azure analysis. Specify `application/json` in the request header for recommendations to be returned in JSON format. Example: Returning Azure VM Scale Sets
GET /analysis/cloud/az	Path Parameter: - analysisId	Collection of the following details in Terraform-map format:	Returns a collection of recommendations for an

HTTP Method	Input	Output	Description

ure/
`<analysisId>`
`/results`
	Query String Parameter Options:
	
	• Element Filters
	
	Accept:
	
	• application/terraform-map
		• provisioningId–This provides the label for each terraform-map recommendation.
		• approvalType
		• currentType
		• densifyPolicy
		• effortEstimate
		• implementationMethod
		• powerState
		• recommendationType
		• predictedUptime
		• recommendedType
		• savingsEstimate
		
		Collection of VMSS-specific details in Terraform-map format:
		
		• avgInstanceCountCurrent
		• avgInstanceCountRecommended
		• minGroupCurrent
		• minGroupRecommended
		• maxGroupCurrent
		• maxGroupRecommended
			Azure analysis.
			
			Specify `application/terraform-map` in the request header forrecommendations to be returned in Terraform-map format.
			
			Example: Returning Azure Recommendations with Low Effort in Terraform-map Format

Parameters

Path Parameters

Table: Azure Recommendations Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique referenced ID of the Azure analysis. When the account grouping feature is enabled and you are working through the Densify API, with the cloud accounts that are part of a group, the workspace ID is replaced by the grouped account's primary key or a unique value that is used as the analysisId or as the ID in the href, that is used to obtain the Impact Analysis and Recommendation Report.

Query String Parameters

Table: Azure Recommendation Query String Parameters

Parameter Name	Type	Description
Element Filters	string	You can use element filters to return a targeted subset of the recommendations. See the "F" (filter) designation in the Response schema table for a list of elements that support filtering. Refer to

Parameter Name	Type	Description
		Filters on page 20 for a complete description of this common operation feature. Usage example: `../results?recommendationType=Upsize`
includeAttributes	▪ true ▪ false	Indicate whether or not to return system attributes: ▪ true—returns all the system attributes; ▪ false—(default) suppress system attributes from the response output. Usage example: `../results?includeAttributes=true`
dataQuality	string	Allows you to indicate which workload type data collection detail to return for the system. See dataQuality for the returned details. Specify a quoted list of workload type names for which data to be returned, separated by commas. Contact Support@Densify.com for a listing of the supported workload type names. Usage example: `../results?dataQuality="CPU Utilization (CINT2006 Rate),Memory Utilization in Percent"` *Note:* A "400-Bad Request" error message is returned if a non-supported workload type is specified.

Response

Table: Azure Recommendations Response Schema

Element	Type	Filter/Sort	Description
entityId	string	F	The Densify assigned entity ID of the cloud system.
resourceId	string	F	The Azure identifier assigned to the instance system.
accountIdRef	string	F	The Azure subscription identifier.
region	string	F	The region containing the instance.
currentType	string	F	The current instance type.
recommendationType	string	F	The recommended action for the system. This is also known as the Optimization Type in the Densify Console (see *Optimization Type Color-Coding Summary* in the topic *Understanding the Instance Optimization Details Report* (Help Topic ID 380390)). The following types of recommended actions are supported for this cloud platform: ▪ `"Just Right"`—this instance is optimally

Element	Type	Filter/Sort	Description
			sized for the workload; ▪ `"Upsize - Optimal Family"`—this instance should be upsized to a more optimal instance family; ▪ `"Upsize"`—this instance should be upsized to an instance within the same instance family; ▪ `"Terminate"`—this instance should be terminated; ▪ `"Downsize - Optimal Family"`—this instance should be downsized to an instance belonging to a more suited instance family; ▪ `"Downsize"`—this instance should be downsized to an instance within the same instance family; ▪ `"Modernize - Optimal Family"`—this instance should be modernized to an instance belonging to a more optimal instance family; ▪ `"Modernize"`—this instance should be modernized to an instance within the same instance family. ▪ `"Not Analyzed"`—this instance has no recommendation due to insufficient workload information. For VM Scale Sets, additional recommendation actions are supported: ▪ `Upscale`—increase compute capacity by adjusting the maximum group size; ▪ `Downscale`—decrease compute capacity by adjusting the minimum group size. Use *recommendationType* as a filter to retrieve only system actions of interest. Systems with insufficient information for analysis do not have a recommendation and are returned with limited elements. See Example: Returning Azure Systems with No Recommendations.
recommendedType	string	F	The recommended instance type after Densify optimization analysis.
implementationMethod	string	F	[Self Optimization|Manual|N/A] Specifies whether this system is configured for Self-Optimization or Manual actioning based on the recommended action (recommendationType) and on the Self-Optimizing Automation policies. **Note:** *Self-Optimization is not supported for VM Scale Sets with more than one VM instance, as a result this element is not returned for these VM Scale Sets.*
predictedUptime	percentage		The predicted uptime (%) for the system is based on the percentage of hours CPU

Element	Type	Filter/Sort	Description
			utilization data is present in the workload range specified in the policy settings.
			Predicted uptime % for new systems started mid-way within the workload range is calculated from the time/date that the system was started, as opposed to the beginning of the interval resulting, in more accurate prediction for the future.
totalHoursRunning	string		The total hours that the system has been running, in the workload range.
			Total hours for a scale group is the total time the scale group has been running and does not include cumulative hours for all of the in-service instances.
totalHours	string		The total hours since the instance was created, that are within the workload range.
			Total hours for a scale group is the total time since the scale group was created and does not include cumulative hours for all of the in-service instances.
			If the system is created in the middle of the workload range, then Densify uses the creation date to calculate total hours.
name	string	F	The Azure name assigned to the system.
rptHref	string		The reference resource to the Impact Analysis and Recommendation Report (also known as the Impact Analysis and Recommendation Report report). Use this reference to downoad the report.
			This report for the specified instance is available for download after the reporting database tables have been updated (i.e. after RDB populate has been executed). By default, the reporting database tables are updated once every night.
			You need to use the following in the request header to download the PDF file: `Accept : application/octet-stream`.
			Note: HTTPS must be enabled to download the Impact Analysis and Recommendation Report PDF.
			See Viewing the Impact Analysis and Recommendation Report _Viewing the Impact Analysis and Recommendation Report_ (Help Topic ID 380450) for details on the content of the report.
			Note: The Impact Analysis and Recommendation Report report is not

Element	Type	Filter/Sort	Description
			currently available for VM Scale Sets with maximum size >1. Example: Downloading an Impact Analysis and Recommendation Report
approvalType	string	F	The approval setting for the system recommendation. The value in this string is derived from the Self-Optimizing Automation policies in implementationMethod and the approval attribute `attr_ApprovalSetting`. Possible settings include: - `"na"`— not approved; - `"all"`—approve any change; - `"<recommended-instance-type>"`— approve changing the instance to the specified *<recommended-instance-type>*.
densifyPolicy	string	F	The Densify policy used for optimization analysis.
savingsEstimate	string		The value of savingestimate output parameter is the difference between the current and recommended instance type cost (this is the catalog cost). When using the API, the predicted uptime is NOT taken into consideration (i.e. [currentCost - recommendedCost]). The Impact Analysis and Recommendation Report report uses the predicted uptime % when calculating estimated savings regardless of whether the report is obtained through the UI or via API. This calculation also applies to scale groups with maximum group size = 1. In this case, the scale group is considered as a VM, with no running instance elements exposed. For scale groups with maximum group size greater than 1, the savings estimate is the difference between the current and recommended instance type cost (i.e. catalog cost), with the average group in-service instances taken into consideration (i.e. [avgInstanceCountCurrent * currentCost] - [avgInstanceCountRecommended* recommendedCost]).
effortEstimate	string	F	[Moderate|Low|Very Low|None|Impossible] This element describes the effort required to investigate and implement the Densify recommendations. Effort for each system is calculated by rule-driven analytics based on factors (such as instance family change, storage change, data quality checks, feature changes, etc.) that can be configured in the policy settings and rule sets which capture best practices.

Element	Type	Filter/Sort	Description
			"Impossible" effort is a result of a manual override for the instance.
			If a system is not 'Not Analyzed', Densify does not return an effortEstimate.
			Note: *When using the Subscription API, Densify returns an effortEstimate of "impossible" for systems that are 'Not Analyzed'.*
powerState	string	F	The power state of Azure VMs. The power state is not available for VM Scale Set.
			Power state is not shown in the UI. .
recommendedHostEntityId	string	F	The Densify entity ID of the catalog instance for the recommended instance type.
currentCost	string		The cost of the existing instance type (i.e. the instance catalog cost).
recommendedCost	string		The cost of the recommended instance type (i.e. the instance catalog cost) after Densify optimization analysis.
serviceType	string	F	The Azure cloud service type. Possible settings include: - Virtual Machine - VM Scale Set Scale groups with maximum group size of 1 (i.e. maxGroupCurrent=1) have serviceType=VM Scale Set. All other returned elements are similar to those for a virtual machine service (i.e. min/max group size values are not returned).
currentHourlyRate	string		The hourly rate for the current instance type (i.e. instance catalog cost / monthly hours). This value is rounded to the nearest penny.
recommendedHourlyRate	string		The hourly rate for the recommended instance type (i.e. instance catalog cost / monthly hours). This value is rounded to the nearest penny.
attributes	array of: - id - name - value		System attributes are properties set during the data collection process by a vendor platform (i.e. Azure) or by Densify for analytics. **Note:** *The attribute array is only returned when includeAttributes=true is added to the query string.*
provisioningId	string		This element is used to identify the terraform-map recommendations for a unique system and corresponds to the "Provisioning Id" Azure user-defined resource tag value. The "Provisioning Id" resource tag is used to uniquely identify a system, since its resourceId or system name could possibly change after an instance type update. See Example: Returning Azure

Element	Type	Filter/Sort	Description
			Recommendations with Low Effort in Terraform-map Format If the "Provisioning Id" user-defined resource tag value is not set, then the system name is used to identify the recommendations.
recommFirstSeen	Unix time (in milliseconds)		The first date and time the recommended instance type (i.e. the recommendedType element) was provided by Densify (Unix Epoch time, in milliseconds).
recommLastSeen	Unix time (in milliseconds)		The latest date and time the recommended instance type (i.e. the recommendedType element) was provided by Densify (Unix Epoch time, in milliseconds).
recommSeenCount	integer		The number of times Densify suggested the recommended instance type (i.e. the recommendedType element). This is the count of Densify analysis processes which produced the same recommended instance type from recommFirstSeen to recommLastSeen. This value is updated when the RDB populate task (i.e. the reporting database update process) is executed. The RDB populate task compares the current `recommendedType` with the new `recommendedType` to update the `recommSeenCount` counter. *Note:* *In typical production environments, where the RDB populate task is scheduled to run once daily post data collection and analysis, the* `recommSeenCount` *value will reflect exactly the number of times the recommended instance type was provided. If your environment executes the RDB populate task more than once daily, the* `recommSeenCount` *value will be inflated beyond the actual number of times the instance type was recommended. Contact Support@Densify.com if you have concerns about the* `recommSeenCount` *value.*
auditInfo	dataCollection: • dateFirstAudited • dateLastAudited • auditCount workloadDataLast30: • firstDate • lastDate • totalDays		The following system data collection details are returned: • `dateFirstAudited`—the first time data was collected for this system (Unix epoch time, in milliseconds); • `dateLastAudited`—the most recent data collection time (Unix epoch time, in milliseconds); • `auditCount`—the number of times that data was collected.

Element	Type	Filter/Sort	Description
	• seenDays		The following system workload collection details, for the last 30 days, are returned: ■ `firstDate`—the first time workload data was collected for this system (Unix epoch time, in milliseconds); ■ `lastDate`—the most recent workload data collection time for this system (Unix epoch time, in milliseconds); ■ `totalDays`—the difference in days between `firstDate` and `lastDate`; ■ `seenDays`—the number of days that at least one workload data was added into Densify for this system. ***Note:*** *If no workload data is collected for the system in the last 30 days, then the workloadDataLast30 element block is not returned.* ***Note:*** *The values in* `auditInfo` *are updated once a day, after the data collection and RDB populate processes are complete (i.e. the reporting tables have been updated with latest data collected).*
dataQuality	array of: ■ workloadName ■ firstSeen ■ lastSeen ■ completeDays ■ partialDays		The dataQuality array provides workload type data collection details for the system: ■ `workloadName`—the workload type name (see DCE Virtual Environment Workload Viewer to find a list of the supported workload types from the Analysis Console); ■ `firstSeen`—the first time this workload was collected (Unix Epoch time, in milliseconds); ■ `lastSeen`—the most recent time this workload was collected (Unix Epoch time, in milliseconds); ■ `completeDays`—the number of complete days that this workload data was collected; ■ `partialDays`—the number of partial days that this workload data was collected. ***Note:*** *The dataQuality array is only returned when the* dataQuality *query string is specified in the request. In addition, if you specify a workload type in the request, for which system workload data does not exist in Densify, then no data is returned.* ***Note:*** *The values in the dataQuality array are updated after data collection and subsequent updates to the reporting*

Element	Type	Filter/Sort	Description
			tables (i.e. RDB Populate process) are completed.
Scale Group Details			
The following details are returned for scale groups with maximum size > 1. Scale groups with maximum size = 1 are treated as VMs.			
orchestrationMode	string		The orchestration mode applied to this VM Scale Set. The orchestration mode can: • Uniform • Flexible Only Uniform scale sets, or Flexible scale sets running identical instance types are analyzed. See Mixed Mode Services for details.
autoScaling	string		The scaling mode applied to this VM Scale Set can be one of: • Manual • Autoscale If set to Flexible, the scale set must be running the same instance types.
profiles	integer		The number of profiles associated with the selected VM Scale Set. Profiles are only applicable to scale sets with scaling mode=Autoscale.
minGroupCurrent	integer		Specifies the current minimum group size of a scaling group.
minGroupRecommended	integer		Specifies the recommended minimum group size for a scaling group.
maxGroupCurrent	integer		Specifies the current maximum group size of a scale group.
maxGroupRecommended	integer		Specifies the recommended maximum group size for a scale group.
avgInstanceCountCurrent	float		The average of in-service instances over the historical interval (or workload range). This interval is defined by your policy (e.g. over the last 60 days).
avgInstanceCountRecommended	float		The predicted average instance count if the scale group recommendations are implemented.

scalingPolicies	string		This value indicates whether the scaling mode needs to be reviewed. Possible values depend on whether or not the VM Scale Set is managed by AKS:
			• VM Scale Set is managed by an AKS cluster:
			  • No Value- If the optimization type = Terminate, then:
			    ○ In the UI, a dash, "–" is displayed.
			    ○ In the API,"Terminate" is returned in the /results.
			  • Keep–If the current average group size is the same as the recommended average group size. The optimization type =Just Right.

Element	Type	Filter/Sort	Description
			- Review—If Densify recommends changing one or more of instance type, average, minimum or maximum group size. VM Scale Set is NOT managed by an AKS cluster: - No Value- If the optimization type = Terminate, then: - In the UI, a dash, "—" is displayed. - In the API,"Terminate" is returned in the `/results`. - Keep—If the current average group size, minimum and maximum group size are not changed. The optimization type =Just Right. - Review—If Densify recommends changing one or more of instance type, average, minimum or maximum group. - Enable Autoscale—If Densify recommends changing one or more of instance type, average, minimum or maximum group size and the VM Scale Set is not managed by AKS. - Adjust –If the current average group size is not the same as the predicted average group size AND instance type and minimum and maximum group size do not change AND the VM Scale Set is not managed by AKS. This element is only returned for scale groups with maximum group size greater than one.

Examples

Example: Returning Azure VM Scale Sets

The following example shows you how to return a collection of Azure recommendations for VM Scale Set.

Example: Returning Azure Recommendations for VM Scale Sets

```
GET /analysis/cloud/azure/b585963b-2f05-62a4-b146-1facd95a8f0d/res-
ults?serviceType=VM+Scale+Set
```

Response:

```
{
    "entityId": "0d2f5815-1dbe-439c-83e3-788180e41fb2",
    "resourceId": "5de3f259-1524-4242-8419-385d3944133b",
    "accountIdRef": "cc377154-9605-4cb0-8b41-1b39e1c4ac0f",
    "region": "eastus",
    "currentType": "standard_ds1_v2",
    "recommendationType": "Just Right",
    "recommendedType": "standard_ds1_v2",
    "predictedUptime": 100.0,
    "totalHoursRunning": 72,
    "totalHours": 72,
    "name": "vmss-test-set",
    "rptHref": "/systems/3bcb4ced-bd8f-3264-9c71/analysis-report",
    "approvalType": "na",
    "densifyPolicy": "API Initial Assessment",
    "savingsEstimate": 0.0,
    "effortEstimate": "Impossible",
    "powerState": "N/A",
    "recommendedHostEntityId": "ef2c3121-3bdd-4866-9708-581ca041c5c8",
    "currentCost": 53.29,
    "recommendedCost": 53.29,
    "serviceType": "VM Scale Set",
    "currentHourlyRate": 0.07,
    "recommendedHourlyRate": 0.07,
    "minGroupCurrent": "0",
    "minGroupRecommended": "0",
    "maxGroupCurrent": "3",
    "maxGroupRecommended": "3",
    "avgInstanceCountRecommended": 3.0,
    "avgInstanceCountCurrent": 3.0,
    "recommFirstSeen": 1708025580003,
    "recommLastSeen": 1709528400000,
    "recommSeenCount": 9,
    "auditInfo": {
        "dataCollection": {
            "dateFirstAudited": 1706731262210,
            "dateLastAudited": 1709530216270,
            "auditCount": 55
        },
        "workloadDataLast30": {
            "firstDate": 1707022800000,
            "lastDate": 1709442000000,
            "totalDays": 29,
            "seenDays": 29
        }
    },
    "orchestrationMode": "Uniform",
    "scalingMode": "Manual",
    "profiles": 1,
    "scalingPolicies": "Keep"
},
```

Example: Returning Azure Instances with Terminate Recommendations

The following example shows you how to return a collection of Azure instances with "Terminate" recommendations.

Example: Returning Azure Terminate Recommendations

Request:

```
GET /analysis/cloud/azure/b585963b-2f05-62a4-b146-1facd95a8f0d/res-
ults?recommendationType=Terminate
```

Response:

```
[
   {
        "entityId": "0d2f5815-1dbe-439c-83e3-788180e41fb2",
        "resourceId": "952d21d6-9ba2-4217-8df1-f5f6417ded45",
        "accountIdRef": "3d4ba999-cbd8-40b8-9998-574be6824a97",
        "region": "centralus",
        "currentType": "standard_d2",
        "recommendationType": "Terminate",
        "recommendedType": "idle.guest",
        "implementationMethod": "N/A",
        "predictedUptime": 18.33,
        "name": "st01-dev-est-125",
        "rptHref": "/systems/0d2f5815-1dbe-439c-83e3-788180e41fb2/analysis-
report",
        "approvalType": "na",
        "densifyPolicy": "Initial Assessment",
        "savingsEstimate": 95.107315,
        "effortEstimate": "Low",
        "powerState": "Running",
        "recommendedHostEntityId": "e4175f48-cf28-459a-b573-3da48948a82e",
        "currentCost": 112.42,
        "recommendedCost": 0.0,
        "serviceType": "Virtual Machine",
        "currentHourlyRate": 0.15,
        "recommendedHourlyRate": 0.00,
        "recommFirstSeen": 1579680587640,
        "recommLastSeen": 1589008760283,
        "recommSeenCount": 24,
        "auditInfo": {
            "dataCollection": {
                "dateFirstAudited": 1571949479093,
                "dateLastAudited": 1588997132800,
                "auditCount": 104
            },
            "workloadDataLast30": {
                "firstDate": 1587009600000,
```

```
                "lastDate": 1588910400000,
                "totalDays": 23,
                "seenDays": 6
            }
        }
    }
    ...
]
```

Example: Returning Azure Recommendations with Low Effort in Terraform-map Format

The following example shows you how to return a collection of recommendations with Low effort in terraform-map form. The label of each recommendation (i.e. "st01-dev-west-108", "st01-dev-west-110" in the example below) is the provisioningId element: "Provisioning Id" Azure user-defined resource tag or system name value.

Example: Return Low Effort Recommendations in Terraform-map Format

Request:

```
GET /analysis/cloud/azure/b585963b-2f05-62a4-b146-1facd95a8f0d/res-
ults?effortEstimate=Low
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Response:

```
densify_recommendations = {
  "st01-dev-west-108" = {
    currentType = "standard_d2"
    recommendedType = "standard_b2ms"
    approvalType = "na"
    predictedUptime = "93.56"
    recommendationType = "Modernize - Optimal Family"
    powerState = "Running"
    implementationMethod = "Self Optimization"
    savingsEstimate = "128.94438"
    effortEstimate = "Low"
    densifyPolicy = "Azure Assessement"
  }
  "st01-dev-west-110" = {
    currentType = "standard_a1_v2"
```

```
      recommendedType = "standard_b1ms"
      approvalType = "na"
      ...
    }
    ...
}
```

Example: Downloading an Impact Analysis and Recommendation Report

The following example shows you how to download a PDF Impact Analysis and Recommendation Report from the `rptHref` resource element provided in the instance recommendation output.

> *Note:* *HTTPS must be enabled to download the Impact Analysis and Recommendation Report PDF.*

Example: Download an Impact Analysis and Recommendation Report

Request:

```
GET /systems/97cfcb18-37a4-4f49-885b-7114eb2ceb30/analysis-report
```

Headers:

```
Accept: application/octet-stream
Authorization: Bearer <apiToken>
```

Example: Returning Azure Systems with No Recommendations

The following example shows you how to return a collection of Azure systems without recommendations. These systems typically do not have adequate data for optimization analysis and have the "Not Analyzed" designation in the recommendationType element.

Example: Returning Azure Systems with No Recommendations in JSON

Request:

```
GET /analysis/cloud/azure/6b8ab8e1-a026-4db9-a4ae-2d684446731f/res-
ults?recommendationType=Not Analyzed
```

Headers:

```
Accept: application/json
Authorization: Bearer <apiToken>
```

Response:

```
[
    {
        "entityId": "1123ea6b-0bdb-4382-9daa-5a597b9f2db4",
        "resourceId": "7a5316a0-4063-4123-a036-3625fea91033",
        "accountIdRef": "3d4ba999-cbd8-40b8-9998-574be6824a88",
        "region": "centralus",
        "currentType": "standard_ds3_v2",
        "recommendationType": "Not Analyzed",
        "name": "st08-est-catf-619",
        "densifyPolicy": "Azure Assess",
        "powerState": "Offline",
        "currentCost": 13.14,
        "serviceType": "Virtual Machine",
        "recommFirstSeen": 1579680587623,
        "recommLastSeen": 1589008760263,
        "recommSeenCount": 24,
        "auditInfo": {
            "dataCollection": {
                "dateFirstAudited": 1571949479093,
                "dateLastAudited": 1588997132800,
                "auditCount": 104
            }
        }
    },
    ...
]
```

Example: Returning Azure Systems with No Recommendations in Terraform-Map

Request:

```
GET /analysis/cloud/azure/6b8ab8e1-a026-4db9-a4ae-2d684446731f/res-
ults?recommendationType=Not Analyzed
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Response:

```
densify_recommendations = {
  "st08-est-catf-619" = {
    currentType = "standard_ds3_v2"
```

```
      recommendationType = "Not Analyzed"
      powerState = "Offline"
      densifyPolicy = "Azure Assess"
   }
   ...
}
```

Analysis: Azure Systems

Description

The `/analysis/cloud/azure/<analysisId>/systems` resource is used to return a collection of all the systems (Virtual Machines) that were included in the Azure optimization analysis.

The recommendations from an Azure optimization analysis can be obtained using the `/analysis/cloud/azure/<analysisId>/results` resource. See Analysis: Azure Recommendations on page 122 for details on the Azure recommendations resource. The set of analyzed systems in an analysis may be more than the set of recommendations produced, as there can be no recommendations for some analyzed systems.

Resource

```
/analysis/cloud/azure/<analysisId>/systems
```

Supported Operations

Table: Azure Systems Supported Operations

Operation	HTTP Method	Input	Output	Description
List all systems included in an Azure analysis	GET `/analysis/cloud/azure/<analysisId>/systems`	Path Parameter: ■ analysisId	Collection of: ■ serviceType ■ resourceId ■ powerState ■ currentType ■ displayName ■ entityId ■ href	Use this resource to return a list of all systems included in the Azure analysis. Example: Listing All Systems in an Azure Analysis

API Reference Guide for Public Cloud

Parameters

Path Parameters

Table: Azure System Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique referenced ID of the Azure analysis.

Response

Table: Azure System Response Schema

Element	Type	Filter/Sort	Description
serviceType	string		The cloud service type (i.e. Virtual Machine).
resourceId	string		The Azure unique identifier assigned to the instance (Virtual Machine).
powerState	string		The power state of the instance.
currentType	string		The current instance type of the Azure instance.
displayName	string		The name assigned to the instance. For Azure Virtual Machines, the displayName is the same as the unique identifier in resourceId.
entityId	string		The Densify assigned entity ID of the Azure system.
href	string		The referenced resource to the system entity. See Systems on page 249 for details of the /systems resource.

Examples

Example: Listing All Systems in an Azure Analysis

The following example shows you how to return all systems (Virtual Machine instances) included in an Azure optimization analysis.

Example: Listing all Systems in an Azure Analysis

Request:

```
GET /analysis/cloud/azure/8b8ab8e1-a026-4db9-a4ae-2d6878467e8f/systems
```

Response:

```
[
    {
        "serviceType": "Virtual Machine",
        "resourceId": "st02-db-edge-695",
        "powerState": "Running",
        "currentType": "standard_a2_v2",
        "displayName": "st02-db-edge-695",
        "entityId": "0137f635-eacd-44da-878f-a4482166f9c1",
        "href": "/systems/0137f635-eacd-44da-878f-a4482166f9c1"
    },
    {
        "serviceType": "Virtual Machine",
        "resourceId": "st01-io-asop-216",
        "powerState": "Running",
        "currentType": "standard_d1_v2",
        "displayName": "st01-io-asop-216",
        "entityId": "065c44d5-040a-4392-9d08-5f581bedb401",
        "href": "/systems/065c44d5-040a-4392-9d08-5f581bedb401"
    },
    ...
]
```

Analysis: Cloud

Description

The `/analysis/cloud` resource is used to return a list of all cloud optimization analysis entities currently in Densify. An analysis entity encompasses cloud resources within a predefined environment scope, specific to the cloud vendor. Typically, the scope of a cloud analysis entity corresponds to the resources within an account, subscription, or project discovered through Densify data collection.

Resource

```
/analysis/cloud
```

Supported Operations

Table: Cloud Analysis Supported Operations

Operation	HTTP Method	Input	Output	Description
List all cloud analyses in Densify	`GET /analysis/cloud`		Collection of: ■ analysisId ■ analysisName ■ analysisCompletedOn ■ href ■ analysisResults ■ analysisStatus	Use this resource to return a list of existing cloud analyses in Densify. Example: Listing All Cloud Analyses

Response

Table:Cloud Analysis Response Schema

Element	Type	Filter/Sort	Description
analysisId	string		The unique identifier for the cloud analysis entity in Densify.
analysisName	string	Filter by: • analysisName • analysisName_like	The name given to the cloud analysis entity. The analysis name typically corresponds to the account ID, subcription ID, or project ID from the infrastructure data collected. Use the `analysisName` and `analysisName_like` filter to return only cloud analyses you are interested in. For example, `/cloud/aws?analysisName_like=bob` will return all AWS analyses with a "bob" substring in the analysis name.
analysisCompletedOn	string		The date and time (in milliseconds) when the last analysis completed. If an analysis has never been completed, "0" is returned.
href	string		The referenced resource to the analysis entity. See Analysis: Entity on page 149 for details of the analysis entity resource.
analysisResults	string		The referenced resource to the recommendations of the analysis. For details of the various cloud analysis recommendations, see: • Analysis: AWS Recommendations on page 83 • Analysis: Azure Recommendations on page 122 • Analysis: GCP Recommendations on page 168
analysisStatus	string		The referenced resource to the status of the analysis. See Analysis: Status on page 206 for details.
message	string		For errors, the message for the status response is returned.
status	number		The HTTP response code of the request error. Possible status values include: • 200–success with request; • 400–invalid parameters; • 401–authentication failed; • 404–resource not found; • 500–internal server error.

Examples

Example: Listing All Cloud Analyses

The following example shows you how to list all cloud analyses in Densify.

Example: Listing all Cloud Analyses

Request:

```
GET /analysis/cloud
```

Response:

```
[
    {
        "analysisId": "9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf",
        "analysisName": "624756828528",
        "analysisCompletedOn": 1548083469463,
        "href": "/analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf",
        "analysisResults": "/analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0ce-
f8c5bf/results",
        "analysisStatus": "/analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0ce-
f8c5bf/status"
    },
    {
        "analysisId": "0930c31d-13ac-4a3f-892e-2a35c8a3f842",
        "analysisName": "DEN-prod-OPS-45",
        "analysisCompletedOn": 1548083655897,
        "href": "/analysis/cloud/gcp/0930c31d-13ac-4a3f-892e-2a35c8a3f842",
        "analysisResults": "/analysis/cloud/gcp/0930c31d-13ac-4a3f-892e-
2a35c8a3f842/results",
        "analysisStatus": "/analysis/cloud/gcp/0930c31d-13ac-4a3f-892e-
2a35c8a3f842/status"
    },
    {
        "analysisId": "6b8ab8e1-a026-4db9-a4ae-2d684446731f",
        "analysisName": "12d89cbc-bc00-4d00-bcf6-ce6ec08d45bd",
        "analysisCompletedOn": 1548340596930,
        "href": "/analysis/cloud/azure/6b8ab8e1-a026-4db9-a4ae-2d684446731f",
        "analysisResults": ""/analysis/cloud/azure/6b8ab8e1-a026-4db9-a4ae-
2d684446731f/results",
        "analysisStatus":"/analysis/cloud/azure/6b8ab8e1-a026-4db9-a4ae-
2d684446731f/status"
    }
]
```

Analysis: Containers

Description

The `/analysis/containers` resource is used to return a list of all optimized container-based analysis entities currently in your Densify system. A container analysis entity encompasses container resources within a predefined environment scope. Typically, the scope of a container analysis entity corresponds to the container resources within a cluster discovered through Densify data collection.

See Container Prerequisites*Container Prerequisites* (Help Topic ID 410140) for container data collection details.

Resource

```
/analysis/containers
```

Supported Operations

Table: Container Analysis Supported Operations

Operation	HTTP Method	Input	Output	Description
List all container analyses in Densify	`GET /analysis/containers`		Collection of: ■ analysisId ■ analysisName ■ analysisCompletedOn ■ href ■ analysisResults ■ analysisStatus	Use this resource to return a list of existing container analyses. Example: Listing All Container Analyses

Response

Table: Container Analysis Response Schema

Element	Type	Filter/Sort	Description
analysisId	string		The unique identifier for the container analysis entity in Densify.
analysisName	string	Filter by: - analysisName - analysisName_like	The name given to the container analysis entity. The analysis name typically corresponds to the cluster name from the container data collected.
analysisCompletedOn	string		The date and time (in milliseconds) when the last analysis completed. If an analysis has never been completed, "0" is returned.
href	string		The referenced resource to the analysis entity. See Analysis: Entity on page 149 for details of the analysis entity resource.
analysisResults	string		The referenced resource to the recommendations of the analysis. See Analysis: Kubernetes Container Recommendations on page 185 for details on container recommendations.
analysisStatus	string		The referenced resource to the status of the analysis. See Analysis: Status on page 206 for details.
message	string		For errors, the message for the status response is returned.
status	number		The HTTP response code of the request error. Possible status values include: - 200—success with request; - 400—invalid parameters; - 401—authentication failed; - 404—resource not found; - 500—internal server error.

Examples

Example: Listing All Container Analyses

The following example shows you how to list all container analyses in Densify.

Example: Listing all Container Analyses

Request:

```
GET /analysis/containers
```

Response:

```
[
    {
        "analysisId": "3d4ba999-cbd8-40b8-9998-574be6824a97",
        "analysisName": "DEN-east-321-45b",
        "analysisCompletedOn": 1510180908845,
        "href": "/analysis/containers/kubernetes/3d4ba999-cbd8-40b8-9998-
574be6824a97",
        "analysisResults": "/analysis/containers/kubernetes/3d4ba999-cbd8-40b8-
9998-574be6824a97/results",
        "analysisStatus": "/analysis/containers/kubernetes/3d4ba999-cbd8-40b8-
9998-574be6824a97/status"
    },
    {
        "analysisId": "97cfcb18-37a4-4f49-885b-8974eb2ceb31",
        "analysisName": "DEN-east-432-2",
        "analysisCompletedOn": 1543346324266,
        "href": "/analysis/containers/kubernetes/97cfcb18-37a4-4f49-885b-
8974eb2ceb31",
        "analysisResults": "/analysis/containers/kubernetes/97cfcb18-37a4-4f49-
885b-8974eb2ceb31/results",
        "analysisStatus": "/analysis/containers/kubernetes/97cfcb18-37a4-4f49-
885b-8974eb2ceb31/status"
    }
]
```

API Reference Guide for Public Cloud

Analysis: Entity

Description

The `/analysis/<platformType>/<platformSubType>` resource is used to return a list of analyses currently in the Densify system. The `/analysis/<platformType>/<platformSubType>/<analysisId>` resource is used to return the details of a specific analysis currently in Densify. An analysis entity encompasses infrastructure resources within a predefined environment scope, specific to the platform and cloud vendor. The infrastructure resources are analyzed to determine optimal recommendations.

See Analysis: Cloud on page 143 and Analysis: Containers on page 146 for the two platform types of analyses supported in Densify.

Resource

```
/analysis/<platformType>/<platformSubType>
/analysis/<platformType>/<platformSubType>/<analysisId>
```

Supported Operations

Table: Analysis Supported Operations

Operation	HTTP Method	Input	Output	Description
List all analyses for a particular platform	GET `/analysis/ <platformType>/ <platformSubType>`	Path Parameter: ▪ platformType ▪ platformSubType	Collection of: ▪ accountId ▪ accountName ▪ analysisCompletedOn ▪ analysisId	Use this resource to return a list of existing analyses for a particular platform and sub-platform

Operation	HTTP Method	Input	Output	Description
and vendor			analysisName analysisResults analysisStatus href policyInstanceId policyName	(vendor). Example: Listing All Kubernetes Container Analyses Example: Listing All AWS Analyses Example: Listing All Azure Analyses
Get a specific analysis	GET /analysis/ <platformType>/ <platformSubType> /<analysisId>	Path Parameter: platformType platformSubType analysisId	accountId accountName analysisCompletedOn analysisId analysisName href analysisResults analysisStatus policyInstanceId policyName	Use this resource to return a specific analysis with a known analysis ID. Example: Getting Details of a Specific Kubernetes Container Analysis

Parameters

Path Parameters

Table: Analysis Path Parameters

Parameter Name	Type	Description
platformType	string	[cloud|containers] The technology platform of the analysis.
platformSubType	string	The platform sub-type of the analysis. This is typically the vendor/flavor of the technology platform. Possible platform sub-types depend on the platformType specified: For platformType = cloud, platformSubType = [aws|gcp|azure] For platformType = containers, platformSubType = [kubernetes]
analysisId	string	The unique referenced ID of the analysis.

Response

Table: Analysis Response Schema

Element	Type	Filter/Sort	Description
accountId	string		The technology account identifier (i.e. AWS account number, Azure subscription ID,

Element	Type	Filter/Sort	Description
			GCP project ID, etc.).
			This element is not returned for container analyses.
accountName	string		The technology platform account name (e.g. AWS account name).
			This element is not returned for container analyses.
analysisCompletedOn	string		The date and time (in milliseconds) when the last analysis completed.
			If an analysis has never been completed, "0" is returned.
analysisId	string		The unique identifier for the analysis entity in Densify.
analysisName	string	Filter by: ▪ analysisName ▪ analysisName_like	The name given to the analysis entity. The analysis name typically corresponds to the cluster name, account ID, subscription ID, or project ID from the infrastructure data collected. Use the `analysisName` and `analysisName_like` filter to return only analyses you are interested in. For example, `/cloud/aws?analysisName_like=bob` will return all AWS analyses with a "bob" substring in the analysis name.
analysisResults	string		The referenced resource to the recommendations of the analysis. For details of the various analysis recommendations, see: ▪ Analysis: AWS Recommendations on page 83 ▪ Analysis: Azure Recommendations on page 122 ▪ Analysis: GCP Recommendations on page 168 ▪ Analysis: Kubernetes Container Recommendations on page 185
analysisStatus	string		The referenced resource to the status of the analysis. See Analysis: Status on page 206 for details.
href	string		The referenced resource to the analysis entity.
message	string		For errors, the message for the status response is returned.
policyInstanceId	string		The Densify policy instance GUID used for optimization analysis.
policyName	string		The Densify policy name used for optimization analysis.
status	number		The HTTP response code of the request error. Possible status values include: ▪ 200–success with request;

Element	Type	Filter/Sort	Description
			• 400–invalid parameters; • 401–authentication failed; • 404–resource not found; • 500–internal server error.

Examples

Example: Listing All Kubernetes Container Analyses

The following example shows you how to list all Kubernetes container analyses in Densify.

Example: Listing all Kubernetes Container Analyses

Request:

```
GET /analysis/containers/kubernetes
```

Response:

```
[
    {
        "policyInstanceId": "094d9c96-4802-4689-a260-ab4c08ee6123",
        "analysisCompletedOn": 1585725434757,
        "analysisResults": "/analysis/containers/kubernetes/8334887b-59ca-
42a4-a241-f970d7c306f1/results",
        "policyName": "Initial Assessment",
        "analysisId": "8334887b-59ca-42a4-a241-f970d7c306f1",
        "href": "/analysis/containers/kubernetes/8334887b-59ca-42a4-a241-
f970d7c306f1",
        "analysisStatus": "/analysis/containers/kubernetes/8334887b-59ca-42a4-
a241-f970d7c306f1/status",
        "analysisName": "mini-kube"
    }
]
```

Example: Listing All AWS Analyses

The following example shows you how to list all AWS analyses in Densify.

Example: Listing all AWS Analyses

Request:

```
GET /analysis/cloud/aws
```

Response:

```
[
  {
        "accountId": "922390019409",
        "policyInstanceId": "44b2ea5f-a3b7-44b2-9437-3c4e69650e58",
        "analysisCompletedOn": 1593606311863,
        "analysisResults": "/analysis/cloud/aws/0a87047f-e8dc-43f0-89bd-
769962b06b9d/results",
        "accountName": "General Services",
        "policyName": "AWS (Mobile_Dev)",
        "analysisId": "0a87047f-e8dc-43f0-89bd-769962b06b9d",
        "href": "/analysis/cloud/aws/0a87047f-e8dc-43f0-89bd-769962b06b9d",
        "analysisStatus": "/analysis/cloud/aws/0a87047f-e8dc-43f0-89bd-
769962b06b9d/status",
        "analysisName": "922390019409 (Mobile_Dev)"
    },
    {
        "accountId": "229192219122",
        "policyInstanceId": "ba022982-12da-48b8-b5f2-af6f0630952d",
        "analysisCompletedOn": 1585732332227,
        "analysisResults": "/analysis/cloud/aws/8bef9d74-94f7-414f-a032-
5855258473a2/results",
        "accountName": "General Services",
        "policyName": "AWS General Prod",
        "analysisId": "8bef9d74-94f7-414f-a032-5855258473a2",
        "href": "/analysis/cloud/aws/8bef9d74-94f7-414f-a032-5855258473a2",
        "analysisStatus": "/analysis/cloud/aws/8bef9d74-94f7-414f-a032-
5855258473a2/status",
        "analysisName": "229192219122"
    }
]
```

Example: Listing All Azure Analyses

The following example shows you how to list all Azure analyses in Densify.

Example: Listing all Azure Analyses

Request:

```
GET /analysis/cloud/azure
```

Response:

```
[
    {
        "accountId": "3d4ba999-cbd8-40b8-9998-574be6824a97",
        "policyInstanceId": "72aa4f1a-906a-4af4-94e9-f8fb606cdfe9",
        "analysisCompletedOn": 1589935057627,
        "analysisResults": "/analysis/cloud/azure/b3573fb3-0a37-406d-bb08-
```

```
24f088e8bc63/results",
        "accountName": "Test Subscription",
        "policyName": "Initial Assessment",
        "analysisId": "b3573fb3-0a37-406d-bb08-24f088e8bc63",
        "href": "/analysis/cloud/azure/b3573fb3-0a37-406d-bb08-24f088e8bc63",
        "analysisStatus": "/analysis/cloud/azure/b3573fb3-0a37-406d-bb08-
24f088e8bc63/status",
        "analysisName": "3d4ba999-cbd8-40b8-9998-574be6824a97"
    },
    {
        "accountId": "cc377154-9605-4cb0-8b41-1b39e1c4ac0f",
        "policyInstanceId": "72aa4f1a-906a-4af4-94e9-f8fb606cdfe9",
        "analysisCompletedOn": 1589978789707,
        "analysisResults": "/analysis/cloud/azure/c1e0bb64-6582-41e8-b7f6-
341797af4436/results",
        "policyName": "Initial Assessment",
        "analysisId": "c1e0bb64-6582-41e8-b7f6-341797af4436",
        "href": "/analysis/cloud/azure/c1e0bb64-6582-41e8-b7f6-341797af4436",
        "analysisStatus": "/analysis/cloud/azure/c1e0bb64-6582-41e8-b7f6-
341797af4436/status",
        "analysisName": "cc377154-9605-4cb0-8b41-1b39e1c4ac0f"
    },
    {
        "accountId": "00d89cbc-bc00-4d00-bcf6-ce6ec08d8fbc",
        "policyInstanceId": "129fa559-2db5-40ff-8a57-94c56900308a",
        "analysisCompletedOn": 1585726564073,
        "analysisResults": "/analysis/cloud/azure/6b8ab8e1-a026-4db9-a4ae-
2d684446731f/results",
        "policyName": "Azure Assessement",
        "analysisId": "6b8ab8e1-a026-4db9-a4ae-2d684446731f",
        "href": "/analysis/cloud/azure/6b8ab8e1-a026-4db9-a4ae-2d684446731f",
        "analysisStatus": "/analysis/cloud/azure/6b8ab8e1-a026-4db9-a4ae-
2d684446731f/status",
        "analysisName": "00d89cbc-bc00-4d00-bcf6-ce6ec08d8fbc"
    }
]
```

Example: Listing All GCP Analyses

The following example shows you how to list all GCP analyses in Densify.

Example: Listing all GCP Analyses

Request:

```
GET /analysis/cloud/gcp
```

Response:

```
[
```

API Reference Guide for Public Cloud

```
    {
        "accountId": "gcpAcctEast-production-op",
        "policyInstanceId": "83003bdd-389e-4210-8ef0-c47a6d025c4d",
        "analysisCompletedOn": 1585736399537,
        "analysisResults": "/analysis/cloud/gcp/0930c31d-13ac-4a3f-892e-
2a35c8a3f842/results",
        "accountName": "DVC_GCP",
        "policyName": "GCP Assessement",
        "analysisId": "0930c31d-13ac-4a3f-892e-2a35c8a3f842",
        "href": "/analysis/cloud/gcp/0930c31d-13ac-4a3f-892e-2a35c8a3f842",
        "analysisStatus": "/analysis/cloud/gcp/0930c31d-13ac-4a3f-892e-
2a35c8a3f842/status",
        "analysisName": "gcpAcctEast-production-op"
    }
]
```

Example: Getting Details of a Specific Kubernetes Container Analysis

The following example shows you how to get the details of a Kubernetes container analysis in Densify with a known analysis ID.

Example: Getting Kubernetes Container Analysis Details

Request:

```
GET /analysis/containers/kubernetes/519a1b9f-aae0-4b65-bb63-1180ff68d27a
```

Response:

```
{
    "policyInstanceId": "094d9c96-4802-4689-a260-ab4c08ee6123",
    "analysisCompletedOn": 1585725429013,
    "analysisResults": "/analysis/containers/kubernetes/519a1b9f-aae0-4b65-
bb63-1180ff68d27a/results",
    "policyName": "Initial Assessment",
    "analysisId": "519a1b9f-aae0-4b65-bb63-1180ff68d27a",
    "href": "/analysis/containers/kubernetes/519a1b9f-aae0-4b65-bb63-
1180ff68d27a",
    "analysisStatus": "/analysis/containers/kubernetes/519a1b9f-aae0-4b65-
bb63-1180ff68d27a/status",
    "analysisName": "kube-master"
}
```

Analysis: GCP Analyze

Description

The `/analysis/gcp` resource is used to return a list of GCP optimization analyses currently in the Densify system.

An analysis is generated for each GCP project associated with the provided GCP service account used during data collection.

The `/analysis/gcp/analyze` resource is used to collect Google Cloud Platform infrastructure data and initiate optimization analysis with the data collected. Below are the series of processes that occur when the initial `/analysis/gcp/analyze` request is triggered:

1. Set up and initiate data collection of the GCP projects based on the associated GCP service account and schedule it to run automatically on a nightly basis.

2. Initiate analysis on the data collected using the default policy.

   - Subsequent analysis is scheduled to run on a nightly basis after data collection.

   - You can optionally configure the results to be sent to a webhook URI upon analysis completion. See for details.

3. While data collection or analysis is in progress, you can check for status (using `/analysis/gcp/<projectId>/status` resource) or wait for the results to be published to an optional webhook URI.

4. The reporting database update is scheduled to run automatically on a nightly basis after the completion of the analysis. This process produces reports for each instance recommendation, which is useful for analysts or application owners. These reports are only created after the scheduled analysis is completed, and may therefore only be available on the following day for a new analysis. Exact timing depends on the size of your environment.

Ad-Hoc Tasks

Generally you do not need to run once-off tasks as both data collection and analysis tasks are scheduled automatically. In cases where you need make an ad-hoc request in addition to the scheduled job, the functionality exists for this endpoint.

Historical Data Collection

When Densify initiates data collection, normally audits collect only the last 24 hours of data. You can optionally collect up to 30 days of historical data. The historical data provides a more representative set of data on which to base resizing and optimization recommendations. You can run an ad-hoc task to collect the historical data.

> *Note:* *Collection of historical data can take a significant amount of time, depending on the number of instances from which Densifyis collecting data. Contact Support@Densify.com to enable historical data collection and details of the performance impact.*

The following settings define the range of historical data to be collected:

- Start date offset—This is the number of days from the 30-day maximum, used to define the start of the range.
- End date offset—This is number of days from yesterday, to end the range of data collected.

These parameters allow you to reduce the number of days of historical data to be collected. If, for example, the daily audit has been running for a few days before the historical audit can be executed then you can set the end offset to exclude the number of days that have already been collected. Thirty days is the maximum number of days that you can go back and collect historical data for Azure and GCP environments.

A connection to the specified cloud account must already exist before you can run an ad hoc audit. When you execute an ad hoc refresh an audit task will be configured but a new connection will not be created. If the cloud connection does not already exist and the API POST contains `triggerAdhocAudit=true`, then you will get an error message.

If there is more than one account associated with the specified account ID (i.e. a payer account with many linked accounts), the Densify API handles it in the same way that analyses are currently rerun using the POST operation.

Once the audit is complete you need to rerun the associated analyses as indicated below or you can wait for the next scheduled execution of the analyses and RDB populate.

Analysis Update

You can make an ad-hoc request to refresh an existing analysis, outside of the scheduled nightly run using `/analysis/cloud/<aws|azure|gcp>/analyze`. This manual, ad hoc analysis request does not perform data collection or reporting database updates. It only runs the analysis on the existing data collected with the following behavior:

- The analysis uses the policy that is configured for the analysis. Contact Support@Densify.com to change the configured policy.

- If a new webhook is provided, the analysis will send results to the new webhook URI. If no webhook is provided, the analysis will send results to the existing webhook, if configured.

- If the same analysis is already running, the request does not proceed and an appropriate message is returned.

- If the specified analysis has data collection scheduled within 30 minutes, the request does not proceed and an appropriate message is returned. For example, if data collection is scheduled to run at 12:05 AM, and you initiate a manual, ad hoc analyze request at 11:45 PM, then the analysis will not proceed and an error message is returned.

Prerequisite Configuration

Before you can collect GCP cloud infrastructure data in Densify, you need to create a GCP service account with services enabled and permissions configured. See *Google Cloud Platform Data Collection Prerequisites* (Help Topic ID 380300) for details.

Note: *When using the Densify API only one project will be processed per API request. This is the case, even if more than one project is associated with the service account.*

Resource

```
/analysis/gcp/analyze
/analysis/gcp
```

Supported Operations

Table: GCP Analysis Supported Operations

Operation	HTTP Method	Input	Output	Description
Run GCP data collection and analysis	POST /analysis/gcp/ analyze	credential projectId connectionName (optional) webHook (optional)	href message status	Use this resource to: 1. Collect GCP cloud data via API for the project associated with the provided service account. 2. Run analysis on the GCP project

Operation	HTTP Method	Input	Output	Description
				data collected. The analysis is created for each project associated with the service account provided. 3. (Optional) Send results to webhook receiving application. 4. Data collection and analysis processes are scheduled to run nightly, subsequent to the initial request.
Rerun GCP data analysis	`POST /analysis/gcp/ analyze`	▪ credential ▪ projectId ▪ webHook *(optional)*	▪ href ▪ message ▪ status	This resource operation is used to re-run an analysis that already exists. You can specify an updated policy and/or webhook to use for the analysis. Data collection is not run. Data collection only occurs during the first `/analyze` request, and is then scheduled to run nightly The updated webhook is saved and will be used in the next scheduled analyses. You cannot initiate a request if data collection or the analyses are in progress or within 30 minutes of the time that these

Operation	HTTP Method	Input	Output	Description
				tasks are scheduled to run.
Run the historical audit	POST /analysis/cloud/gcp/analyze	Request Body Parameter: • credential • projectId • triggerAdhocAudit • startDayOffset *(optional)* • endDayOffset *(optional)*	• href • message • status	This resource operation is used to re-run an audit for which a connection and daily, scheduled audit already exists. You can optionally specify the number of days of historical data to collect. If not specified the previous 30 days from yesterday's date are collected. If you initiate an audit request when data collection or analysis is already running or within 30 minutes of the time that these tasks are scheduled to run, then the request will fail and an error message is returned.
List all generated analyses	GET /analysis/cloud/gcp	Path Parameter: • N/A Request Body Parameter: • N/A	Lists all analyses that have been created with details.	This resource operation is used to obtain the analysis ID that is required for other operations.

Parameters

GCP Path Parameters

Table: GCP Analysis Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique referenced ID of the GCP analysis.

GCP Body Request Parameters

Table: GCP Analysis Parameters

Parameter Name	Type	Description
credential	string	The GCP service account credential details. You must specify the details from the .JSON credential file, inline when making this request *Google Cloud Platform Data Collection Prerequisites* (Help Topic ID 380300) for details.
projectID	string	The projectid for which to create the environment. You must provide the projectid even if the credential file is linked to multiple projects. A multi-project list is not derived from the credential file.
connectionName *(optional)*	string	Use the connection name to clearly identify this connection within Densify. This name will appear in the **Saved Connections** list in theDensify UI. By default, the connection name is set to the GCP project. The connection name must be unique within the GCP connection type section, so if the name is already in use, the request fails with an error message. This connection name can be used for filtering. ***Note:*** *The Connection Name is limited to 32-characters.*
endDayOffset *(optional)*	string	Historical data end day offset. This parameter is optional and is used to configure the range of the historical audit. It is used in conjunction with the parameters, `Trigger ad-hoc audit` and `Start Day Offset` to set the end day of the range of historical days of data to collect. If no value is specified and the parameter, `Trigger ad-hoc audit` has been enabled, then the end date is set to yesterday. If you specify any number other than 0, then that number is used to offset the range's end date from yesterday. i.e. if End Day Offset=5 and yesterday was Dec 1, then the end date will be Nov 25. When AFv2 is enabled, this setting is not used and the end date is always "yesterday". Do not pass this parameter in your API call.
startDayOffset *(optional)*	string	Historical data start day offset. This parameter is optional and is used to configure the historical audit. It is used in conjunction with the parameters, `Trigger ad-hoc audit` and `End Day Offset` to set the start day of the range of historical days of data to collect. If no value is specified and the parameter, `Trigger ad-hoc audit` has been enabled, then the start date is set to 30 days previous to yesterday's date. If you specify a number less than 30, that number is used to offset the start date from 30 days in the past. i.e. if Start Day Offset=10 then the start date will be 10 days offset from 30 days, so 20 days previous to yesterday. You can use these setting to define a smaller range (i.e. 20 days). You cannot collect more than 30 days of historical data.
triggerAdhocAudit	string	The flag to trigger an ad-hoc 30-day historical audit.

Parameter Name	Type	Description
(optional)		This parameter is optional and used to run the historical audit immediately and once only.
		Typically the historical audit is run first, when data collection is initiated for the specified subscription /project. If the historical audit has been disabled for performance reasons, it can be run once to collect the historical data as compute resources become available.
		A connection to the specified subscription or project must already exist before you can use this flag.
		A once-off task will be configured but a new connection cannot be created. If the connection does not already exist AND the API post contains ad hoc=true, then you will see an error message.
webHook *(optional)*	▪ uri ▪ authType ▪ authValue	The webhook definition to an external application. Optimization results are sent to the webhook-defined application when the analysis is complete. See Parameters on page 228 for details of each parameter in the webhook definition.

Responce

Table: GCP Analysis Responce Elements

Element	Type	Filter/Sort	Description
analysisName	string		The analysis name corresponds to the GCP project ID for the collected infrastructure data. An analysis is created for each project associated with the service account credential provided.
href	string		The referenced resource to the recommendations of the analysis. See Analysis: GCP Recommendations on page 168 for details on GCP analysis recommendations.
completed	string		The date and time (in milliseconds) when the last analysis completed.
analysisId	string		The Densify internal ID of the analysis entity.
phase	string		The current phase of the specified analysis. Possible phases include: ▪ analyzing : *<percent completed>%* ▪ not analyzing
message	string		The message for the analysis status. For errors, the message for the following status response is returned.
status	number		The HTTP response code of the request. Possible status values include: ▪ 200–success with request; ▪ 400–invalid parameters; ▪ 401–authentication failed; ▪ 404–resource not found; ▪ 500–internal server error.

Examples

Example: Running GCP Data Collection and Analysis

The following example shows you how to initiate GCP data collection and analysis, and send the results to a WebHook.

Example: Initiating GCP Data Collection and Analysis

Request:

```
POST /analysis/gcp/analyze
{
  "connectionName": "gcp-testing",
  "policyInstanceId": "583e0068-45a3-4c4e-baec-473d4daa4095",
  "projectId": "pm-testing-608378",
  "credential": {
        "type": "service_account",
        "project_id": "engineering-183318",
        "private_key_id": "b768acaecb36312ffe771dadc68534e0f1cdeeb9",
        "private_key": "-----BEGIN PRIVATE KEY-----
        IIEvQIBADANBgkqhkiG9w0BAQEFAASCBKcwggSjAgEAAoIBAQCy0JZTceQh8ThwUab0P7G5O
Of7jWnJKGov/BXHp1xYp3NFGzQH+NAnib5wajYd3eg3Vo5Jpipf87yeztcN0AQ0G8iqaCnOQqMVa26T
2ZzJT4CATUJ7+08RO4h8kL8DiO9BD5L0B9hPbTWXgSG2ModNmag5OG+c
        -----END PRIVATE KEY-----",
         "client_email": "min-163416@eng-163.iam.gserviceaccount.com",
         "client_id": "28884131006232124",
         "auth_uri": "https://accounts.google.com/o/oauth2/auth"
         "token_uri": "https://oauth2.googleapis.com/token",
         "auth_provider_x509_cert_url":
"https://www.googleapis.com/oauth2/v1/certs",
         client_x509_cert_url":
"https://www.googleapis.com/robot/v1/metadata/x509/min-permission-
1833engineering-183318.iam.gserviceaccount.com"

  "webHook": {
        "uri": "http://mywebhookserver/webhook/results",
        "authType": "basic",
        "authValue": "tester:testerpassword"
        }
}
```

Response:

```
{
  "href": "Not available"
  "message": "Analysis in progress",
  "status": 200
}
```

Analysis: GCP Delete

Description

The `/analysis/cloud/gcp/<analysisId>` resource can be used to delete the audit and all associated scheduler entries of the cloud analysis specified by `<analysisId>`.

- The single-day and 60-day historical audits are removed, if they exist.
- The audit and all associated scheduler entries, are deleted, if they exist.

When executing the delete operation in an environment observe the following:

- When deleting an audit for an environment associated with a single account, the operation proceeds as outlined above.
- When deleting an audit, if Densify determines that there is a many-to-one relationship for this connection, then the delete request fails. This type of multi-account connection can only be created through the Cloud Connection Wizard and must be deleted through the Cloud Connection Wizard.

During a delete operation, any attempt to access other associated endpoints related to the specified `<analysisId>` (i.e. GET, PUT, POST) will trigger a "400 Bad Request" error response.

While the delete operation is in progress, you can check the status (using the `/analysis/cloud/gcp/<analysisId>/status` resource) or wait for the results to be published to an optional webhook URI.

Collected data is saved for the defined retention period and the environment and associated analysis structure are saved in case you want to generate reports on the historical data or restart data collection. Since the environment and analyses are not deleted, a GET call for the specified `<analysisId>` will always succeed regardless of whether the associated connection has been deleted.

If you use a GET call to list available environment and analyses, for which the audits have been deleted, they are still listed. You can reestablish the audit without have to re-create the analysis. You will still see the data in the Densify Console.

Resource

```
/analysis/cloud/gcp/analysisId
```

Supported Operations

Table: GCP Supported Operations

Operation	HTTP Method	Input	Output	Description
Delete GCP data collection and analysis	DELETE /analysis/cloud/gcp/<analysisId>	Request Body Parameter: ■ analysisId ■ webHook *(optional)*	■ message ■ status	This resource operation is used to delete audit and audit-related items for the specified analysis. Example: Deleting GCP Data Collection

Parameters

Path Parameters

Table: GCP Analysis Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique, referenced ID of the GCP analysis.

Request Body Parameters

Table: GCP Analysis Request Body Parameters

Parameter Name	Type	Description
webHook *(optional)*	■ uri ■ authType ■ authValue	The webhook definition to an external application. Status can be sent to the webhook-defined application when the delete operation is complete. See Parameters on page 228 for details of each parameter in the webhook definition.

Response

Table: GCP Analysis Response Schema

Element	Type	Filter/Sort	Description
message	string		The informational message for the status response that is returned.
status	number		The HTTP response code of the request. Possible status values include: ■ 200–success with request; ■ 400–invalid parameter; ■ 401–authentication failed; ■ 404–resource not found; ■ 405–method not allowed; ■ 500–internal server error.

Examples

Example: Deleting GCP Data Collection

The following example shows you how to delete the audit and send the results to a WebHook.

Example: Deleting Data Collection Audit

Request:

```
{
    DELETE /analysis/cloud/gcp/fd9234a7b43c-7e4b-4084-8f46-b898c2c2

    "webHook": {
        "uri": "http://mywebhookserver/webhook/results",
        "authType": basic,
        "authValue": "tester:testerpassword"
                }
}
```

Response:

```
{
  "message": "OK",
  "status": 200
}
```

Example: Deleting an Audit that is in Progress

The following example shows you the response if you attempt to delete an audit that is in progress.

Example: Deleting Data Collection Audit

Request:

```
DELETE /analysis/cloud/gcp/fd9234a7b43c-7e4b-4084-8f46-b898c2c2
```

Response:

```
{
  "message": "Connection has an audit in progress. It cannot be deleted at
this time.",
  "status": 500
}
```

Example: Deleting an Aggregate Audit

The following example shows you the response if you attempt to delete an audit that has more than one audit associated with the specified analysis ID.

If Densify determines that there is a many-to-one relationship for this connection, then the delete request fails. This type of multi-subscription connection can only be created through the Cloud Connection Wizard and must be deleted through the Cloud Connection Wizard.

Example: Deleting a Many-to-1 Data Collection Audit

Request:

```
DELETE /analysis/cloud/gcp/fd9234a7b43c-7e4b-4084-8f46-b898c2c2
```

Response:

```
{
    "message": "Account dfc04848-3848-44c0-b85a-02311951de36 was created via UI
and subscriptions [cc377154-9605-4cb0-8b41-1b39e1c4ac0f,3d4ba999-cbd8-40b8-
9998-574be6824a97] are incompatible with API use; please delete via UI",
  "status": 500
}
```

Analysis: GCP Recommendations

Description

The `/analysis/cloud/gcp/<analysisId>/results` resource is used to return a collection of Compute Engine instance recommendations after optimization analysis has been performed on your collected GCP infrastructure data.

For each instance recommendation, you can also download a PDF version of the Impact Analysis and Recommendation Report, which details system impact based on the recommendations. This report is useful to application owners who need to review recommended changes to their application resources. See the rptHref resource element for details on how to download this report.

To return a collection of all instances included in a GCP optimization analysis, see Analysis: GCP Systems on page 182.

Resource

```
/analysis/cloud/gcp/<analysisId>/results
```

Supported Operations

Table: GCP Recommendations Supported Operations

HTTP Method	Input	Output	Description
GET	Path Parameter:	Collection of (JSON):	Returns a collection of

HTTP Method	Input	Output	Description

`/analysis/cloud/gcp/`
`<analysisId>`
`/results`
	analysisId
	
	Query String Parameter Options:
	Element Filters
	includeAttributes
	dataQuality
	
	Accept:
	application/json
		entityId
		resourceId
		accountIdRef
		currentType
		recommendationType
		recommendedType
		implementationMethod
		predictedUptime
		Analysis: GCP Recommendations
		Analysis: GCP Recommendations
		name
		rptHref
		approvalType
		densifyPolicy
		savingsEstimate
		effortEstimate
		powerState
		recommendedHostEntityId
		currentCost
		recommendedCost
		serviceType
		currentHourlyRate
		recommendedHourlyRate
		attributes
		recommFirstSeen
		recommLastSeen
		recommSeenCount
		auditInfo
		dataQuality
			recommendations for a GCP analysis.
			
			Specify `application/json` in the request header for returned recommendations in JSON format.
			
			Note: *The returned recommendations can be in either JSON or Terraform-map format.*
			
			Example: Returning GCP Instances with Downsize Recommendations

`GET`
`/analysis/cloud/gcp/`
`<analysisId>`
`/results`
	Path Parameter:
	analysisId
	
	Query String Parameter Options:
	Element Filters
	
	Accept:
	application/terraform-map
		Collection of (Terraform-map):
		provisioningId (label of each terraform-map recommendation)
		currentType
		recommendedType
		approvalType
		predictedUptime
		recommendationType
		powerState
		implementationMethod
		savingsEstimate
		effortEstimate
		densifyPolicy
			Returns a collection of recommendations for a GCP analysis in Terraform-map form. Specify `application/terraform-map` in the request header.
			
			Note: *The returned recommendations can be in either JSON or Terraform-map format.*
			
			Example: Returning GCP Recommendations with Low Effort in Terraform-map Form

Parameters

Path Parameters

Table: GCP Recommendations Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique referenced ID of the GCP analysis. When the account grouping feature is enabled and you are working through the Densify API, with the cloud accounts that are part of a group, the workspace ID is replaced by the grouped account's primary key or a unique value that is used as the analysisId or as the ID in the href, that is used to obtain the Impact Analysis and Recommendation Report.

Query String Parameters

Table: GCP Recommendation Query String Parameters

Parameter Name	Type	Description
Element Filters	string	You can use element filters to return a targeted subset of the recommendations. See the "F" (filter) designation in the Response schema table for a list of elements that support filtering. Refer to Filters on page 20 for a complete description of this common operation. Usage example: `../results?recommendationType=Upsize`
includeAttributes	• true • false	Indicate whether or not to return system attributes: • true—returns all system attributes; • false—(default) suppress system attributes from the response output. Usage example: `../results?includeAttributes=true`

Parameter Name	Type	Description
dataQuality	string	Allows you to indicate which workload type data collection detail to return for the system. See dataQuality for the returned details. Specify a quoted list of workload type names for which data to be returned, separated by commas. Contact Support@Densify.com for a listing of the supported workload type names. Usage example: `../results?dataQuality="CPU Utilization (CINT2006 Rate),Memory Utilization in Percent"` *Note:* *A "400-Bad Request" error message is returned if a non-supported workload type is specified.*

Response

Table: GCP Recommendations Response Schema

Element	Type	Filter/Sort	Description
entityId	string	F	The Densify assigned entity ID of the cloud system.
resourceId	string	F	The GCP identifier assigned to the Compute Engine system .
accountIdRef	string	F	The GCP Project identifier.
currentType	string	F	The current instance type of the Compute Engine system.
recommendationType	string	F	The recommended action for the system. This is also known as the Optimization Type in the Densify Console (see *Optimization Type Color-Coding Summary* in the topic *Understanding the Instance Optimization Details Report* (Help Topic ID 380390)). The following types of recommended actions are supported for this cloud platform: - `"Just Right"`—this instance is optimally sized for the workload; - `"Upsize - Optimal Family"`—this instance should be upsized to a more optimal instance family; - `"Upsize"`—this instance should be upsized to an instance within the same instance family; - `"Terminate"`—this instance should be terminated; - `"Downsize - Optimal Family"`—this instance should be downsized to an

Element	Type	Filter/Sort	Description
			instance belonging to a more suited instance family; ■ `"Downsize"`—this instance should be downsized to an instance within the same instance family; ■ `"Modernize - Optimal Family"`—this instance should be modernized to an instance belonging to a more optimal instance family; ■ `"Modernize"`—this instance should be modernized to an instance within the same instance family. ■ `"Not Analyzed"`—this instance has no recommendation due to insufficient workload information. Systems with insufficient information for analysis do not have a recommendation and are returned with limited elements. See Example: Returning GCP Systems with No Recommendations.
recommendedType	string	F	The recommended instance type after Densify optimization analysis.
implementationMethod	string	F	[Self Optimization|Manual|N/A] Specifies whether this system is configured for Self-Optimization or Manual actioning based on the recommended action (recommendationType) and on the Self-Optimizing Automation policies.
predictedUptime	percentage		The predicted uptime (%) for the system is based on the percentage of hours CPU utilization data is present in the workload range specified in the policy settings. Predicted uptime % for new systems started mid-way within the workload range is calculated from the time/date that the system was started, as opposed to the beginning of the interval resulting, in more accurate prediction for the future.
name	string	F	The GCP name assigned to the system.
rptHref	string		The reference resource to the Impact Analysis and Recommendation Report (also known as the Impact Analysis and Recommendation Report report). Use this reference to downoad the report. This report for the specified instance is available for download after the reporting database tables have been updated (i.e. after RDB populate has been executed). By default, the reporting database tables are updated once every night. You need to use the following in the request header to download the PDF file:

Element	Type	Filter/Sort	Description
			`Accept : application/octet-stream`.
			Note: *HTTPS must be enabled to download the Impact Analysis and Recommendation Report PDF.*
			See Viewing the Impact Analysis and Recommendation Report *Viewing the Impact Analysis and Recommendation Report* (Help Topic ID 380450) for details on the content of the report. Example: Downloading an Impact Analysis and Recommendation Report
approvalType	string	F	The approval setting for the system recommendation. The value in this string is derived from the Self-Optimizing Automation policies in implementationMethod and the approval attribute `attr_ApprovalSetting`. Possible settings include: • `"na"`– not approved; • `"all"`–approve any change; • `"<recommended-instance-type>"`– approve changing the instance to the specified *<recommended-instance-type>*.
densifyPolicy	string	F	The Densify policy used for optimization analysis.
savingsEstimate	string		The value of savingestimate output parameter is the difference between the current and recommended instance type cost (this is the catalog cost). When using the API, the predicted uptime is NOT taken into consideration (i.e. [currentCost - recommendedCost]). The Impact Analysis and Recommendation Report report uses the predicted uptime % when calculating estimated savings regardless of whether the report is obtained through the UI or via API.
effortEstimate	string	F	[Moderate|Low|Very Low|None|Impossible] This element describes the effort required to investigate and implement the Densify recommendations. Effort for each system is calculated by rule-driven analytics based on factors (such as instance family change, storage change, data quality checks, feature changes, etc.) that can be configured in the policy settings and rule sets which capture best practices.

Element	Type	Filter/Sort	Description
			"Impossible" effort is a result of a manual override for the instance. If a system is not 'Not Analyzed', Densify does not return an effortEstimate. *Note:* *When using the Subscription API, Densify returns an effortEstimate of "impossible" for systems that are 'Not Analyzed'.*
powerState	string	F	The power state of the system.
recommendedHostEntityId	string	F	The Densify entity ID of the catalog instance for the recommended instance type.
currentCost	string		The cost of the existing instance type (i.e. the instance catalog cost).
recommendedCost	string		The cost of the recommended instance type (i.e. the instance catalog cost) after Densify optimization analysis.
serviceType	string	F	The GCP service type: 🔹 Compute Engine
currentHourlyRate	string		The hourly rate for the current instance type (i.e. instance catalog cost / monthly hours). This value is rounded to the nearest penny.
recommendedHourlyRate	string		The hourly rate for the recommended instance type (i.e. instance catalog cost / monthly hours). This value is rounded to the nearest penny.
attributes	array of 🔹 id 🔹 name 🔹 value		System attributes are properties set during the data collection process by a vendor platform (i.e. GCP) or by Densify for analytics. *Note:* *The attribute array is only returned when includeAttributes=true is added to the query string.*
provisioningId	string		This element is used to identify the terraform-map recommendations for a unique system and corresponds to the "Provisioning Id" GCP user-defined label value. The "Provisioning Id" label is used to uniquely identify a system, since its resourceId or system name could possibly change after an instance type update. See Example: Returning GCP Recommendations with Low Effort in Terraform-map Form. If the "Provisioning Id" label value is not set, then the system name is used to identify the recommendations.
recommFirstSeen	Unix time (in milliseconds)		The first date and time the recommended instance type (i.e. the recommendedType

Element	Type	Filter/Sort	Description
			element) was provided by Densify (Unix Epoch time, in milliseconds).
recommLastSeen	Unix time (in milliseconds)		The latest date and time the recommended instance type (i.e. the recommendedType element) was provided by Densify (Unix Epoch time, in milliseconds).
recommSeenCount	integer		The number of times Densify suggested the recommended instance type (i.e. the recommendedType element). This is the count of Densify analysis processes which produced the same recommended instance type from recommFirstSeen to recommLastSeen. This value is updated when the RDB populate task (i.e. the reporting database update process) is executed. The RDB populate task compares the current `recommendedType` with the new `recommendedType` to update the `recommSeenCount` counter. *Note: In typical production environments, where the RDB populate task is scheduled to run once daily post data collection and analysis, the `recommSeenCount` value will reflect exactly the number of times the recommended instance type was provided. If your environment executes the RDB populate task more than once daily, the `recommSeenCount` value will be inflated beyond the actual number of times the instance type was recommended. Contact Support@Densify.com if you have concerns about the `recommSeenCount` value.*
auditInfo	dataCollection: • dateFirstAudited • dateLastAudited • auditCount workloadDataLast30: • firstDate • lastDate • totalDays • seenDays		The following system data collection details are returned: • `dateFirstAudited`—the first time data was collected for this system (Unix epoch time, in milliseconds); • `dateLastAudited`—the most recent data collection time (Unix epoch time, in milliseconds); • `auditCount`—the number of times that data was collected. The following system workload collection details, for the last 30 days, are returned: • `firstDate`—the first time workload data was collected for this system (Unix epoch time, in milliseconds); • `lastDate`—the most recent workload

Element	Type	Filter/Sort	Description
			data collection time for this system (Unix epoch time, in milliseconds); `totalDays`—the difference in days between `firstDate` and `lastDate`; `seenDays`—the number of days that at least one workload data was added into Densify for this system.
			Note: If no workload data is collected for the system in the last 30 days, then the workloadDataLast30 element block is not returned.
			Note: The values in `auditInfo` are updated once a day, after the data collection and RDB populate processes are complete (i.e. the reporting tables have been updated with latest data collected).
dataQuality	array of: • workloadName • firstSeen • lastSeen • completeDays • partialDays		The dataQuality array provides workload type data collection details for the system: `workloadName`—the workload type name (see DCE Virtual Environment Workload Viewer to find a list of the supported workload types from the Analysis Console); `firstSeen`—the first time this workload was collected (Unix Epoch time, in milliseconds); `lastSeen`—the most recent time this workload was collected (Unix Epoch time, in milliseconds); `completeDays`—the number of complete days that this workload data was collected; `partialDays`—the number of partial days that this workload data was collected.
			Note: The dataQuality array is only returned when the *dataQuality* query string is specified in the request. In addition, if you specify a workload type in the request, for which system workload data does not exist in Densify, then no data is returned.
			Note: The values in the dataQuality array are updated after data collection and subsequent updates to the reporting tables (i.e. RDB Populate process) are

Element	Type	Filter/Sort	Description
			completed. See Example: Returning GCP Instances with Downsize Recommendations.

Examples

Example: Returning GCP Instances with Downsize Recommendations

The following example shows you how to return a collection of GCP instances with "Downsize" recommendations. In addition, this example also returns the "Disk I/O Bytes" workload data collection timestamps.

Example: Returning GCP Downsize Recommendations with "Disk I/O Bytes" Workload

Request:

```
GET /analysis/cloud/gcp/d230c31d-13ac-543f-892e-2a35c8a3f232/res-
ults?recommendationType=Downsize&dataQuality="Disk I/O Bytes"
```

Response:

```
[
  {
      "entityId": "1679679b-9726-4a6b-9c33-a00f9261b6c1",
      "resourceId": "4090409965214718888",
      "accountIdRef": "east-prod-gidz",
      "currentType": "n1-standard-8",
      "recommendationType": "Downsize",
      "recommendedType": "n1-standard-4",
      "implementationMethod": "Self Optimization",
      "predictedUptime": 13.82,
      "totalHoursRunning": 230,
      "name": "gl04.doop_xtr-32",
      "rptHref": "/systems/1679679b-9726-4a6b-9c33-a00f9261b6c1/analysis-
report",
      "approvalType": "na",
      "densifyPolicy": "GCP-Assess-Prod",
      "savingsEstimate": 132.91621,
      "effortEstimate": "Very Low",
      "powerState": "Running",
      "recommendedHostEntityId": "4e8309c5-cf70-42d9-ab70-4aa46516ff62",
      "currentCost": 277.4,
      "recommendedCost": 138.16,
```

```
        "serviceType": "Compute Engine",
        "currentHourlyRate": 0.38,
        "recommendedHourlyRate": 0.24,
        "recommFirstSeen": 1579680587570,
        "recommLastSeen": 1589008760110,
        "recommSeenCount": 24,
        "auditInfo": {
             "dataCollection": {
                 "dateFirstAudited": 1571951513223,
                 "dateLastAudited": 1588997765350,
                 "auditCount": 101
             },
             "workloadDataLast30": {
                 "firstDate": 1587009600000,
                 "lastDate": 1588910400000,
                 "totalDays": 23,
                 "seenDays": 5
             }
        },
        "dataQuality": [
             {
                 "workloadName": "Disk I/O Bytes",
                 "firstSeen": 1569297600000,
                 "lastSeen": 1588910400000,
                 "completeDays": 116,
                 "partialDays": 0
             }
        ]
    },
    ...
]
```

Example: Returning GCP Recommendations with Low Effort in Terraform-map Form

The following example shows you how to return a collection of recommendations with Low effort in terraform-map form. The label of each recommendation (i.e. "gl02.camp_gas-340", "gl02.camp_gas-432" in the example below) is the provisioningId element: "Provisioning Id" GCP label or system name value.

Example: Return Low Recommendations in Terraform-map

Request:

```
GET /analysis/cloud/gcp/d230c31d-13ac-543f-892e-2a35c8a3f232/res-
ults?effortEstimate=Low
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Response:

```
densify_recommendations = {
  "gl02.camp_gas-340" = {
    currentType = "n1-standard-8"
    recommendedType = "n1-highmem-4"
    approvalType = "na"
    predictedUptime = "64.01"
    recommendationType = "Downsize - Optimal Family"
    powerState = "Running"
    implementationMethod = "Manual"
    savingsEstimate = "98.27805"
    effortEstimate = "Low"
    densifyPolicy = "GCP-Assess-Prod"
  }
  "gl02.camp_gas-432" = {
    currentType = "n1-standard-8"
    recommendedType = "n1-highmem-4"
    approvalType = "na"
    predictedUptime = "88.0"
    recommendationType = "Downsize - Optimal Family"
    powerState = "Running"
    implementationMethod = "Manual"
    savingsEstimate = "98.27805"
    effortEstimate = "Low"
    densifyPolicy = "GCP-Assess-Prod"
  }
   ...
}
```

Example: Downloading an Impact Analysis and Recommendation Report

The following example shows you how to download a PDF Impact Analysis and Recommendation Report from the rptHref resource element provided in the instance recommendation output.

Note: *HTTPS needs to be enabled to download the Impact Analysis and Recommendation Report PDF.*

Example: Download an Impact Analysis and Recommendation Report

Request:

```
GET /systems/1679679b-9726-4a6b-9c33-a00f9261b6c1/analysis-report
```

Headers:

```
Accept: application/octet-stream
Authorization: Bearer <apiToken>
```

Example: Returning GCP Systems with No Recommendations

The following example shows you how to return a collection of GCP systems without recommendations. These systems typically do not have adequate data for optimization analysis and have the "Not Analyzed" designation in the recommendationType element.

Example: Returning GCP Systems with No Recommendations In JSON

Request:

```
GET /analysis/cloud/gcp/6b8ab8e1-a026-4db9-a4ae-2d684446731f/res-
ults?recommendationType=Not Analyzed
```

Headers:

```
Accept: application/json
Authorization: Bearer <apiToken>
```

Response:

```
[
    {
        "entityId": "1679679b-9726-4a6b-9c33-a00f9261b6c1",
        "resourceId": "4090409723758657994",
        "accountIdRef": "Prod9873318",
        "currentType": "n1-standard-8",
        "recommendationType": "Not Analyzed",
        "name": "g105.doop_xtr-32",
        "densifyPolicy": "GCP Assess",
        "powerState": "Offline",
        "currentCost": 27.47,
        "serviceType": "Compute Engine",
        "recommFirstSeen": 1579680587570,
        "recommLastSeen": 1589008760110,
        "recommSeenCount": 24,
        "auditInfo": {
            "dataCollection": {
                "dateFirstAudited": 1571951513223,
                "dateLastAudited": 1588997765350,
                "auditCount": 101
            }
        }
```

```
      },
      ...
  ]
```

Example: Returning GCP Systems with No Recommendations in Terraform-Map

Request:

```
GET /analysis/cloud/gcp/6b8ab8e1-a026-4db9-a4ae-2d684446731f/res-
ults?recommendationType=Not Analyzed
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Response:

```
densify_recommendations = {
  "g105.doop_xtr-32" = {
    currentType = "n1-standard-8"
    recommendationType = "Not Analyzed"
    powerState = "Offline"
    densifyPolicy = "GCP Assess"
  }
  ...
}
```

Analysis: GCP Systems

Description

The `/analysis/cloud/gcp/<analysisId>/systems` resource is used to return a collection of all the systems that were included in the Google Cloud Platform (GCP) optimization analysis.

The recommendations from an GCP optimization analysis can be obtained using the `/analysis/cloud/gcp/<analysisId>/results` resource. See Analysis: GCP Recommendations on page 168 for details on the GCP recommendations resource. The set of analyzed systems may be more than the set of recommendations produced, as there can be potentially no recommendations for some analyzed systems.

Resource

```
/analysis/cloud/gcp/<analysisId>/systems
```

Supported Operations

Table: GCP Systems Supported Operations

Operation	HTTP Method	Input	Output	Description
List all systems included in a GCP analysis	GET `/analysis/cloud/gcp/<analysisId>/systems`	Path Parameter: analysisId	Collection of: • serviceType • resourceId • powerState • currentType • displayName • entityId • href	Use this resource to return a list of all systems included in the GCP analysis. Example: Listing All Systems in a GCP Analysis

Parameters

Path Parameters

Table: GCP Systems Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique referenced ID of the GCP analysis.

Response

Table: GCP Systems Response Schema

Element	Type	Filter/Sort	Description
serviceType	string		The cloud service type (i.e. Compute Engine).
resourceId	string		The GCP unique identifier assigned to the instance system.
powerState	string		The power state of the system.
currentType	string		The current instance type of the system.
displayName	string		The GCP name assigned to the system.
entityId	string		The Densify assigned entity ID of the system.
href	string		The referenced resource to the system entity. See Systems on page 249 for details of the `/systems` resource.

Examples

Example: Listing All Systems in a GCP Analysis

The following example shows you how to return all systems (instances) included in an GCP optimization analysis.

Example: Listing all Systems in a GCP Analysis

Request:

```
GET /analysis/cloud/gcp/d930c31d-13ac-4a3f-892e-2a35c8a3f232/systems
```

Response:

```
[
    {
        "serviceType": "Compute Engine",
        "resourceId": "4390409739570762220",
        "powerState": "Running",
        "currentType": "n1-standard-8",
        "displayName": "gl08.mln_gas-286",
        "entityId": "00d60166-b9a5-4825-990e-a2efeb3caa32",
        "href": "/systems/00d60166-b9a5-4825-990e-a2efeb3caa32"
    },
    {
        "serviceType": "Compute Engine",
        "resourceId": "4090399486108714123",
        "powerState": "Running",
        "currentType": "n1-highmem-8",
        "displayName": "gl08.hosa_far-348",
        "entityId": "02cbfdb5-3a0d-4ecb-8732-c6f04cfcf7f9",
        "href": "/systems/02cbfdb5-3a0d-4ecb-8732-c6f04cfcf7f9"
    },
    ...
]
```

Analysis: Kubernetes Container Recommendations

Description

The `/analysis/containers/kubernetes/<analysisId>/results` resource is used to return a collection of recommendations after optimization analysis has been performed on your Kubernetes container data in Densify.

To return a collection of all the Kubernetes containers included in the optimization analysis, see Analysis: Kubernetes Container Systems on page 197. The set of analyzed containers may be more than the set of container recommendations due to the possibility of no recommendations for some analyzed containers.

Resource

```
/analysis/containers/kubernetes/<analysisId>/results
```

Supported Operations

Table: Kubernetes Container Recommendations Supported Operations

HTTP Method	Input	Output	Description
`GET /analysis/containers/kubernetes/<analysisId>/results`	Path Parameter: - analysisId Query String Parameter Options: - Element Filters - dataQuality Accept: - application/json	Collection of (JSON): - container - cluster - hostName - predictedUptime - displayName - recommLastSeen - podService - auditInfo - recommendedCpuLimit - currentCount - recommSeenCount - currentMemLimit - recommendedMemLimit - recommendationType - recommendedCpuRequest - recommFirstSeen - controllerType - currentMemRequest - entityId - currentCpuLimit - dataQuality - recommendedMemRequest - currentCpuRequest - namespace	Returns a collection of recommendations for a Kubernetes container analysis. Specify `application/json` in the request header for returned recommendations in JSON format. **Note:** *The returned recommendations can be in either JSON or Terraform-map format.* Example: Returning Kubernetes Container Recommendations
`GET /analysis/containers/kubernetes/<analysisId>/results`	Path Parameter: - analysisId Query String Parameter Options: - Element Filters Accept: - application/terraform-map	Collection of (Terraform-map): - container - cluster - hostName - predictedUptime - controllerType - displayName (the label of each terraform-map recommendation) - currentMemRequest - currentCpuLimit - podService - recommendedCpuLimit - recommendedMemRequest - currentCpuRequest - currentCount	Returns a collection of recommendations for a Kubernetes container analysis. Specify `application/terraform-map` in the request header for Terraform-map output. **Note:** *The returned recommendations can be in either JSON or Terraform-map format.* Example: Returning Kubernetes Container Recommendations in Terraform-map Form

HTTP Method	Input	Output	Description
		- currentMemLimit - namespace - recommendedMemLimit - recommendationType - recommendedCpuRequest	

Parameters

Path Parameters

Table: Kubernetes Container Recommendations Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique referenced ID of the Kubernetes analysis.

Query String Parameters

Table: Kubernetes Container Recommendations Query String Parameters

Parameter Name	Type	Description
Element Filters	string	You can use element filters to return a targeted subset of the recommendations. See the "F" (filter) designation in the Response schema table for a list of elements that support filtering. Refer to Filters on page 20 for a complete description of this common operation feature. Usage example: `../results?recommendationType=Resize`
dataQuality	string	Allows you to indicate which workload type data collection detail to return for the system. See dataQuality for the returned details. Specify a quoted list of workload type names for which data to be returned, separated by commas. Contact Support@Densify.com for a listing of the supported workload type names. Usage example: `../results?dataQuality="CPU Utilization (CINT2006 Rate),Memory Utilization in Percent"` **Note:** *A "400-Bad Request" error message is returned if a non-supported workload type is specified.*

Response

Table: Kubernetes Container Recommendations Response Schema

Element	Type	Filter/Sort	Description
container	string	F	The container name.
cluster	string	F	The container's cluster name.
hostName	string		The Densify internal reference used to ensure that containers are uniquely identified across multiple clusters, even if they potentially have the same namespace, pod and/or container names.
predictedUptime	percentage		The predicted uptime (%) for the container is based on the percentage of hours that CPU utilization data is present in the historical interval specified in the policy settings.
displayName	string	F	The display name is a combination of the pod and container name.
recommLastSeen	Unix time (in milliseconds)		The latest date and time that the following set of recommendations were provided by Densify(Unix Epoch time, in milliseconds): • recommendedMemLimit • recommendedCpuLimit • recommendedMemRequest • recommendedCpuRequest on page 191 Note: If any recommendation value was updated, then Densify would considered that as a new set of recommendations and the recommLastSeen value would not be updated.
podService	string	F	The pod or service name for the container.
auditInfo	▪ dataCollection: • dateFirstAudited • dateLastAudited • auditCount ▪ workloadDataLast30: • firstDate • lastDate • totalDays • seenDays		The following system data collection details are returned: ▪ dateFirstAudited—the first time data was collected for this container (Unix epoch time, in milliseconds); ▪ dateLastAudited—the most recent data collection time (Unix epoch time, in milliseconds); ▪ auditCount—the number of times that data was collected. The following container workload collection details, for the last 30 days, are returned:

Element	Type	Filter/Sort	Description
			• `firstDate`—the first time workload data was collected for this container (Unix epoch time, in milliseconds); • `lastDate`—the most recent workload data collection time for this container (Unix epoch time, in milliseconds); • `totalDays`—the difference in days between `firstDate` and `lastDate`; • `seenDays`—the number of days that at least one workload data was added into Densify for this container. *Note:* *If no workload data is collected for the container in the last 30 days, then the workloadDataLast30 element block is not returned.* *Note:* *The values in* `auditInfo` *are updated once a day (after the data collection and RDB populate processes are complete).*
recommendedCpuLimit	float		The recommended CPU limit for the container after Densify optimization analysis.
currentCount	int		The number of copies running in the Controller. This value comes form the container data collection. If size of the Controller is not found, then this value is set to 1.
recommSeenCount	integer		The number of times Densify recommended the same values for the following elements: • recommendedMemLimit • recommendedCpuLimit • recommendedMemRequest • recommendedCpuRequest on page 191 This is the count of Densify analysis processes which produced the same recommendation values from recommFirstSeen to recommLastSeen. This value is updated when the RDB populate task (i.e. the reporting database update process) is executed. *Note:* *In typical production*

Element	Type	Filter/Sort	Description
			environments, where the RDB populate task is scheduled to run once daily post data collection and analysis, the `recommSeenCount` *value will reflect exactly the number of times the same recommendations were provided after data collection and analysis. If your environment executes the RDB populate task more than once daily, the* `recommSeenCount` *value will be inflated beyond the actual number of times the analysis produced the same recommendations. Contact Support@Densify.com if you have concerns about the* `recommSeenCount` *value.*
currentMemLimit	float		The current memory limit configured for the container.
recommendedMemLimit	float		The recommended memory limit for the container after Densify optimization analysis.
recommendationType	string	F	The recommended action for this container: - `"Just Right"`—this container manifest is optimally configured for the workload; - `"Upsize"`—increase one or more of CPU Request, CPU Limit, Memory Request, or Memory Limit settings; - `"Downsize"`—decrease one or more of CPU Request, CPU Limit, Memory Request, or Memory Limit settings; - `"Resize"`— resize at least two of CPU Request, CPU Limit, Memory Request, or Memory Limit settings (i.e. at least one recommendation is a size increase and the other one is a size decrease); - `"Resize from Unspecified"`— resize recommendations for this container manifest are made without current CPU Request, CPU Limit, Memory Request, or Memory Limit values; - `"Not Analyzed"`— there is insufficient data to recommend CPU Request, CPU Limit, Memory

Element	Type	Filter/Sort	Description
			Request, or Memory Limit values.
			Containers with insufficient information for analysis do not have recommendations and are returned with limited elements. See Example: Returning Kubernetes Containers with No Recommendations.
recommendedCpuRequest	float		The recommended CPU request for the container after Densify optimization analysis.
recommFirstSeen	Unix time (in milliseconds)		The first date and time that the following set of recommendations were provided by Densify (Unix Epoch time, in milliseconds): • recommendedMemLimit • recommendedCpuLimit • recommendedMemRequest • recommendedCpuRequest on page 191 *Note:* *If any recommendation value (above) is different than the previous set of recommFirstSeen recommendation values, then Densify would considered that as a new set of recommendations and the recommFirstSeen value would be updated.*
controllerType	string		The type of controller (i.e. "ReplicatSet", "ReplicationController", "DaemonSet", "StatefulSet", "Deployment", "Job", etc.).
currentMemRequest	float		The current memory request configured for the container.
entityId	string		The Densify assigned entity ID of the container system.
currentCpuLimit	float		The current CPU limit configured for the container.
dataQuality	array of: • workloadName • firstSeen • lastSeen • completeDays • partialDays		The dataQuality array provides workload type data collection details for the system: • `workloadName`—the workload type name (see DCE Virtual Environment Workload Viewer to find a list of the supported workload types from the Analysis Console); • `firstSeen`—the first time this workload was collected (Unix Epoch time, in milliseconds); • `lastSeen`—the most recent time this workload was collected (Unix

Element	Type	Filter/Sort	Description
			Epoch time, in milliseconds); `completeDays`—the number of complete days that this workload data was collected; `partialDays`—the number of partial days that this workload data was collected.
			Note: *The dataQuality array is only returned when the* [dataQuality](#) *query string is specified in the request. In addition, if you specify a workload type in the request, for which system workload data does not exist in Densify, then no data is returned.*
			Note: *The values in the dataQuality array are updated after data collection and subsequent updates to the reporting tables (i.e. RDB Populate process) are completed.*
recommendedMemRequest	float		The recommended memory request for the container after Densify optimization analysis.
currentCpuRequest	float		The current CPU request configured for the container.
namespace	string	F	The container's namespace.

Examples

Example: Returning Kubernetes Container Recommendations

The following example shows you how to return a collection of Kubernetes Container recommendations in JSON format.

Example: Returning Kubernetes Container Recommendations

Request:

```
GET /analysis/containers/kubernetes/e9298ac3-a143-41ed-b7d7-62f659f2a4c6/res-
ults
```

Headers:

```
Accept: application/json
Authorization: Bearer <apiToken>
```

Response:

```
[
    {
        "container": "kube-apiserver",
        "cluster": "kube-ctrl",
        "hostName": "e89c16ee-1f7c-3e40-bdfa-75f33de36405",
        "predictedUptime": 0.08,
        "displayName": "kube-apiserver-kube-ctrl.in.densify.com__kube-
apiserver",
        "recommLastSeen": 1597104000000,
        "podService": "kube-apiserver-kube-ctrl.in.densify.com",
        "auditInfo": {
          "workloadDataLast30": {
                "totalDays": 29,
                "seenDays": 29,
                "firstDate": 1594612800000,
                "lastDate": 1597104000000
            },
            "dataCollection": {
                "auditCount": 40,
                "dateFirstAudited": 1593648000000,
                "dateLastAudited": 1597104000000
            }
        },
        "recommendedCpuLimit": 80,
        "currentCount": 4,
        "recommSeenCount": 6,
        "currentMemLimit": 630,
        "recommendedMemLimit": 630,
        "recommendationType": "Just Right",
        "recommendedCpuRequest": 80,
        "recommFirstSeen": 1593648000000,
        "controllerType": "ReplicationController",
        "currentMemRequest": 628,
        "entityId": "06df1b52-a4ac-4328-94c5-67bfa0ea95b4",
        "currentCpuLimit": 80,
        "recommendedMemRequest": 628,
        "currentCpuRequest": 80,
        "namespace": "kube-system"
    },
    {
        "container": "kube-scheduler",
        "cluster": "kube-ctrl",
        "hostName": "56d85c60-94f2-388f-9e2b-74c5afa75beb",
        "predictedUptime": 0.08,
        "displayName": "kube-scheduler-kube-ctrl.in.densify.com__kube-sched-
uler",
        "recommLastSeen": 1597104000000,
        "podService": "kube-scheduler-kube-ctrl.in.densify.com",
```

```
        ...
    },
    ...
]
```

Example: Returning Kubernetes Container Recommendations in Terraform-map Form

The following example shows you how to return a collection of recommendations in terraform-map format.

Example: Return Kubernetes Container Recommendations in Terraform-map

Request:

```
GET /analysis/containers/kubernetes/e7298ac3-a143-41ed-b7d7-62f659f2a4c6/res-
ults
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Response:

```
densify_recommendations = {
  "kube-controller-manager-kube-ctrl.in.densify.com__kube-controller-manager"
= {
    container = "kube-controller-manager"
    cluster = "kube-ctrl"
    hostName = "3e021430-5b65-307c-b049-3c0edca9be43"
    predictedUptime = "0.08"
    controllerType = "ReplicationController"
    displayName = "kube-controller-manager-kube-ctrl.int.densify.com__kube-con-
troller-manager"
    currentMemRequest = "152.0"
    currentCpuLimit = "30.0"
    podService = "kube-controller-manager-kube-ctrl.in.densify.com"
    recommendedCpuLimit = "30.0"
    recommendedMemRequest = "152.0"
    currentCpuRequest = "30.0"
    currentCount = "2"
    currentMemLimit = "152.0"
    namespace = "kube-system"
    recommendedMemLimit = "152.0"
    recommendationType = "Just Right"
    recommendedCpuRequest = "30.0"
  }
```

```
    "kubernetes-dashboard-845747bdd4__kubernetes-dashboard" = {
      container = "kubernetes-dashboard"
      cluster = "kube-ctrl"
      hostName = "93d0c4a7-a915-3cc8-80cf-19729e8b091c"
      predictedUptime = "0.08"
      controllerType = "ReplicaSet"
      displayName = "kubernetes-dashboard-845747bdd4__kubernetes-dashboard"
      currentMemRequest = "68.0"
      currentCpuLimit = "10.0"
      podService = "kubernetes-dashboard-845747bdd4"
      recommendedCpuLimit = "10.0"
      recommendedMemRequest = "68.0"
      currentCpuRequest = "10.0"
      currentCount = "1"
      currentMemLimit = "68.0"
      namespace = "kube-system"
      recommendedMemLimit = "68.0"
      recommendationType = "Just Right"
      recommendedCpuRequest = "10.0"
    }
     ...
  }
```

Example: Returning Kubernetes Containers with No Recommendations

The following example shows you how to return a collection of Kubernetes containers without recommendations. These containers typically do not have adequate data for optimization analysis and have the "Not Analyzed" designation in the recommendationType element.

Example: Returning Kubernetes Containers with No Recommendations In JSON

Request:

```
GET /analysis/containers/kubernetes/8334887b-59ca-42a4-a241-f970d7c306f1/res-
ults?recommendationType=Not Analyzed
```

Headers:

```
Accept: application/json
Authorization: Bearer <apiToken>
```

Response:

```
[
    {
```

```
            "container": "data-forwarder",
            "recommFirstSeen": 1594080000000,
            "cluster": "kube-ctrl",
            "controllerType": "ReplicaSet",
            "displayName": "densify1__data-forwarder",
            "recommLastSeen": 1597104000000,
            "currentMemRequest": 2048,
            "entityId": "b1253c3b-16fe-47b5-b4db-6a72f541ab7e",
            "podService": "densify1",
            "currentCpuRequest": 520,
            "currentCount": 1,
            "recommSeenCount": 4,
            "currentMemLimit": 3584,
            "namespace": "default",
            "recommendationType": "Not Analyzed"
        }
    ]
```

Example: Returning Kubernetes Containers with No Recommendations in Terraform-Map

Request:

```
GET /analysis/containers/kubernetes/8334887b-59ca-42a4-a241-f970d7c306f1/res-
ults?recommendationType=Not Analyzed
```

Headers:

```
Accept: application/terraform-map
Authorization: Bearer <apiToken>
```

Response:

```
densify_recommendations = {
 "densify1__data-forwarder" = {
    container = "data-forwarder"
    cluster = "kube-master"
    controllerType = "ReplicaSet"
    currentCpuRequest = "520.0"
    displayName = "densify1__data-forwarder"
    currentCount = "1"
    currentMemLimit = "3584.0"
    namespace = "default"
    currentMemRequest = "2048.0"
    recommendationType = "Not Analyzed"
    podService = "densify1"
  }
}
```

Analysis: Kubernetes Container Systems

Description

The `/analysis/containers/kubernetes/<analysisId>/systems` resource is used to return a collection of all container systems that were included in the Kubernetes optimization analysis.

The number of analyzed systems (i.e. `/containers/kubernetes/<analysisId>/systems` entities) will always be greater than or equal to the number of system recommendations produced (i.e. `/containers/kubernetes/<analysisId>/results` entities), as some systems may not have any recommendations. See Analysis: Kubernetes Container Recommendations on page 185 for details of the Kubernetes container recommendation resource.

Resource

```
/analysis/containers/kubernetes/<analysisId>/systems
```

Supported Operations

Table: Kubernetes Container Systems Supported Operations

HTTP Method	Input	Output	Description
GET `/analysis/containers/kubernetes/<analysisId>/systems`	Path Parameter: analysisId	Collection of: ▪ hostName ▪ displayName	Use this resource to return a list of all container systems included in a Kubernetes

HTTP Method	Input	Output	Description
		◼ entityId ◼ href	container analysis. Example: Listing All Systems in a Kubernetes Container Analysis

Parameters

Path Parameters

Table: Kubernetes Container Systems Path Parameters

Parameter Name	Type	Description
analysisId	string	The unique referenced ID of the Kubernetes container analysis.

Response

Table: Kubernetes Container Systems Response Schema

Element	Type	Filter/Sort	Description
hostName	string		The Densify internal reference used to ensure that containers are uniquely identified across multiple clusters, even if they potentially have the same namespace, pod and/or container names.
displayName	string		The display name is a combination of the pod and container name.
entityId	string		The Densify assigned entity ID of the container system.
href	string		The referenced resource to the Densify system entity representing this container. See Systems on page 249 for details of the `/systems` resource.

Examples

Example: Listing All Systems in a Kubernetes Container Analysis

The following example shows you how to return all systems (container) included in a Kubernetes container optimization analysis.

Example: Listing all Systems in a Kubernetes Container Analysis

Request:

```
GET /analysis/containers/kubernetes/d7298ac3-a143-41bb-b7d7-62f659f2a8c5/sys-
tems
```

Response:

```
[
    {
        "hostName": "d3cf25f4-01b1-4c0b-66a8-86bd1ea771d3",
        "displayName": "kube-abcserver-kube-ctrl.densify.com__kube-abcserver",
        "entityId": "06770825-988a-4aa4-b047-41bda995a69e",
        "href": "/systems/06770825-988a-4aa4-b047-41bda995a69e"
    },
    {
        "hostName": "e9422b59-97ef-40b1-b589-587a23a761d3",
        "displayName": "kube-scheduler-kube-ctrl.densify.com__kube-scheduler"
        "entityId": "1f7981f5-dd90-416b-88f9-67f032188536",
        "href": "/systems/1f7981f5-dd90-416b-88f9-67f032188536"
    },
    ...
]
```

Analysis: Policy

Description

The following resources are used to return a list of Densify platform-specific cloud policies available for analysis:

- `/analysis/cloud/aws/policy`
- `/analysis/cloud/gcp/policy`
- `/analysis/cloud/azure/policy`
- `/analysis/containers/kubernetes/policy`

Policies are used to define the business and technical behaviors and requirements for optimization. Densify optimization analysis utilizes the platform-specific default policy employing industry best practices. Contact Densify support (Support@Densify.com) to review your policy settings and to adjust or enable additional cloud policies for your environment.

You can also verify the specific details of the cloud policy used during an analysis through the Densify user interface, see Cloud Optimization Policies*Cloud Optimization Policies* in the topic *Viewing Policy Settings*(Help Topic ID 120200) for details.

Contact Support@Densify.com to review the specific details of the cloud policy used for your analysis.

Resource

```
/analysis/cloud/aws/policy
/analysis/cloud/gcp/policy
/analysis/cloud/azure/policy
/analysis/containers/kubernetes/policy
```

Supported Operations

Table: Policy Supported Operations

HTTP Method	Input	Output	Description
GET /analysis/cloud/aws/policy	Query String Parameter: details=true	• policyId • policyInstanceId • name • description (*optional*)	Used to return a list of AWS policies available for analysis purposes. Example: List All the AWS Cloud Policies Available Example: Listing AWS Cloud Policies with Descriptions
GET /analysis/cloud/gcp/policy	Query String Parameter: details=true	• policyId • policyInstanceId • name • description (*optional*)	Used to return a list of GCP policies used for analysis. Example: Listing GCP Cloud Policies with Descriptions
GET /analysis/cloud/azure/policy	Query String Parameter: details=true	• policyId • policyInstanceId • name • description (*optional*)	Used to return a list of Azure policies used for analysis.
GET /analysis/containers/kubernetes/policy	Query String Parameter: details=true	• policyId • policyInstanceId • name • description (*optional*)	Used to return a list of Kubernetes Container policies available for analysis purposes. *Note:* *DevOps policies are not supported for Container analysis. Contact support (* support@densify.com *) to review your policy settings for Container resource optimization.*

Parameters

Query String Parameters

Table: Analysis Policy Query String Parameters

Parameter Name	Type	Description
details=true *(optional)*	string	This option returns additional policy description details.

Response

Table: Analysis Policy Response Schema

Element	Type	Sort By	Filter	Description
policyId	string			The analysis policy type ID.
policyInstanceId	string			The entity ID of the policy instance.
name	string			The policy name.
description *(optional)*	string			A description of the policy. This element is only returned when the `?details=true` option is added to the request.

Examples

Example: List All the AWS Cloud Policies Available

The following example shows you how to list all the available AWS cloud policies.

Example: Listing AWS Cloud Policies

Request:

```
GET /analysis/cloud/aws/policy
```

Response:

```
[
  {
    "policyId": "4a63f651-a583-4157-97ff-35651370ffbe",
    "policyInstanceId": "69fa4c99-1be2-4048-94a7-36fd83d07f37",
    "name": "DevOps-Default"
  },
```

```
      {
        "policyId": "4a63f651-a583-4157-97ff-35651370ffbe",
        "policyInstanceId": "0c0ef18b-9367-4071-b733-396f63e51925",
        "name": "DevOps-Automation"
      },
      {
        "policyId": "4a63f651-a583-4157-97ff-35651370ffbe",
        "policyInstanceId": "70a2ef4a-2ebb-4209-8ec8-9f6c70f77a74",
        "name": "DevOps-Efficiency"
      }
    ]
```

Example: Listing AWS Cloud Policies with Descriptions

The following example shows you a request to list all the available AWS cloud policies with description.

Example: Listing AWS Cloud Policies with Description

Request:

```
GET /analysis/cloud/aws/policy?details=true
```

Response:

```
[
    {
      "policyId": "4a63f651-a583-4157-97ff-35651370ffbe",
      "policyInstanceId": "0c0ef18b-9367-4071-b733-396f63e51925",
      "name": "DevOps-Automation",
      "description": "This policy is intended for generating instance sizing and
instance family optimization recommendations that require little or no review
before being implemented. CLOE utilizes rule-driven analytics to predict the
effort of changing instance type from current to recommended, and this policy
favors Low effort recommendations, producing higher automation at the expense
of lower cost savings.\r\nThe resource utilization of each system is modeled
using a minimum of 7 days and up to 90 days of historical workload.\r\nWhen
optimizing instance sizes and families, the predicted CPU and memory usage must
not exceed 65% and 85%, respectively.\r\nThis policy will not specify burstable
(T-series) instance families for workloads unless they are already running in a
burstable family and will not change CPU Architecture from Intel to AMD or vice
versa.\r\nWhen memory usage metrics are not available, the analysis assumes the
existing memory allocation is required and will not change the memory
configuration."
    },
    {
      "policyId": "4a63f651-a583-4157-97ff-35651370ffbe",
      "policyInstanceId": "69fa4c99-1be2-4048-94a7-36fd83d07f37",
      "name": "DevOps-Default",
      "description": "This policy reflects best practices for generating
instance sizing and instance family optimization recommendations.\r\nThe
resource utilization of each system is modeled using a minimum of 7 days and up
to 60 days of historical workload.\r\nWhen optimizing instance sizes and
families, the predicted CPU and memory usage levels must not exceed 70% and
90%, respectively.\r\nWhen memory usage metrics are not available, the analysis
effectively assumes the existing memory allocation of the instance is required
and will not change the memory configuration."
```

```
      }
  ]
```

Example: Listing GCP Cloud Policies with Descriptions

The following example shows you a request to list all the available GCP cloud policies with description.

Example: Listing GCP Cloud Policies with Description

Request:

```
GET /analysis/cloud/gcp/policy?details=true
```

Response:

```
[
    {
      "policyId": "4fe32fc6-6067-4647-8a87-9bd1dc74389e",
      "policyInstanceId": "43636c78-851b-4f25-8733-5debbbb6856b",
      "name": "DevOps-Automation",
      "description": "This policy is intended for generating instance sizing and
instance family optimization recommendations that require little or no review
before being implemented. CLOE utilizes rule-driven analytics to predict the
effort of changing instance type from current to recommended, and this policy
favors Low effort recommendations, producing higher automation at the expense
of lower cost savings.\r\nThe resource utilization of each system is modeled
using a minimum of 7 days and up to 90 days of historical workload.\r\nWhen
optimizing instance sizes and families, the predicted CPU and memory usage must
not exceed 65% and 85%, respectively.\r\nThis policy will not specify burstable
(T-series) instance families for workloads unless they are already running in a
burstable family and will not change CPU Architecture from Intel to AMD or vice
versa.\r\nWhen memory usage metrics are not available, the analysis assumes the
existing memory allocation is required and will not change the memory
configuration."
    },
    {
      "policyId": "4fe32fc6-6067-4647-8a87-9bd1dc74389e",
      "policyInstanceId": "6131f154-8453-48c3-9747-edb327ec0bed",
      "name": "DevOps-Efficiency",
      "description": "This policy is intended for generating instance sizing and
family optimization that provide maximum operational efficiency (and lowest
operating cost). This policy also includes recommendations rated as Moderate
effort, generating higher cost reduction than the Densify Defaults policy.
Because of this, any recommendations should be reviewed before
implementing.\r\nThe resource utilization of each system is modeled using a
minimum of 3 day and up to 30 days of historical workload.\r\nWhen optimizing
instance sizes and families, the predicted CPU and memory usage levels must not
exceed 75% and 95%, respectively.\r\nWhen memory usage metrics are not
available, the analysis effectively assumes the existing memory allocation of
the instance is required and will not change the memory configuration."
    },
    {
      "policyId": "4fe32fc6-6067-4647-8a87-9bd1dc74389e",
      "policyInstanceId": "9b8cb5db-5045-4dbe-a6ee-740038c6b0b0",
      "name": "DevOps-Default",
```

```
    "description": "This policy reflects best practices for generating
instance sizing and instance family optimization recommendations.\r\nThe
resource utilization of each system is modeled using a minimum of 7 days and up
to 60 days of historical workload.\r\nWhen optimizing instance sizes and
families, the predicted CPU and memory usage levels must not exceed 70% and
90%, respectively.\r\nWhen memory usage metrics are not available, the analysis
effectively assumes the existing memory allocation of the instance is required
and will not change the memory configuration."
  }
]
```

Analysis: Status

Description

The `/analysis/<platformType>/<platformSubType>/<analysisId>/status` resource is used to return the current status of an analysis in Densify.

Resource

```
/analysis/<platformType>/<platformSubType>/<analysisId>/status
```

Supported Operations

Table: Analysis Status Supported Operations

Operation	HTTP Method	Input	Output	Description
Check for analysis status	`GET /analysis/<platformType>/<platformSubType>/<analysisId>/status`	Path Parameter: ■ platformType ■ platformSubType ■ analysisId	■ analysisStage ■ webHookStatus ■ statusMessage For errors: ■ message ■ status	Use this resource to check the status of an analysis by providing the analysis ID. Example: Checking for Kubernetes Container Analysis Status Example: Checking for AWS Analysis Status Example: Checking for GCP Analysis Status

Parameters

Path Parameters

Table: Analysis Status Path Parameters

Parameter Name	Type	Description
platformType	string	[cloud|containers] The technology platform of the analysis.
platformSubType	string	The platform sub-type of the analysis. This is typically the vendor/flavor of the technology platform. Possible platform sub-types depend on the platformType specified: For platformType = cloud, platformSubType = [aws|gcp|azure] For platformType = containers, platformSubType = [kubernetes]
analysisId	string	The unique referenced ID of the analysis.

Response

Table: Analysis Status Response Schema

Element	Type	Filter/Sort	Description
analysisStage	string		The current stage of the specified analysis. Possible stages include: Analyzing : {*percent*} Completed Not analyzed Completed
webHookStatus	string		The status of the last webhook POST request. Recommendations are pushed to the webhook URI via a POST request after analysis completion.
statusMessage	string		The message for the analysis status.
message	string		For errors, the message for the status response is returned.
status	number		The HTTP response code of the request error. Possible status values include: 200–success with request; 400–invalid parameters; 401–authentication failed; 404–resource not found; 500–internal server error.

Examples

Example: Checking for Kubernetes Container Analysis Status

The following example shows you how to check for a specific Kubernetes container analysis status.

Example: Checking for Kubernetes Container Analysis Status

Request:

```
GET /analysis/containers/kubernetes/6e4da559-cbd8-40b8-5498-644be6843a93/status
```

Response:

```
{
  "analysisStage": "Completed",
  "webHookStatus": "",
  "statusMessage": "Analysis ABC-east-665-b was last completed on Wed Jan 30
18:08:45 EST 2019."
}
```

Example: Checking for AWS Analysis Status

The following example shows you how to check for a specific AWS analysis status.

Example: Checking for AWS Analysis Status

Request:

```
GET /analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf/status
```

Response:

```
{
  "analysisStage": "Completed",
  "webHookStatus": "",
  "message": "Analysis 231345225455 was last completed on Mon Jan 21 10:11:09
EST 2019."
}
```

Example: Checking for Azure Analysis Status

The following example shows you how to check for a specific Azure analysis status.

Example: Checking for Azure Analysis Status

Request:

```
GET /analysis/cloud/azure/6b8ab8e1-a026-4db9-a4ae-2d684446731f/status
```

Response:

```
{
  "analysisStage": "Completed",
```

```
  "webHookStatus": "Success",
  "statusMessage": "Analysis 00d89cbc-bc00-4d00-bcf6-ce6ec09d8adc was last
completed on Thu Jan 24 09:36:36 EST 2019."
}
```

Example: Checking for GCP Analysis Status

The following example shows you how to check for a specific GCP analysis status.

Example: Checking for GCP Analysis Status

Request:

```
GET /analysis/cloud/gcp/0930c31d-13ac-4a3f-892e-2a35c8a3f842/status
```

Response:

```
{
  "analysisStage": "Analyzing : 38% Completed",
  "webHookStatus": "",
  "statusMessage": "Analysis gcpAcctEast-608378 is currently running."
}
```

Kubernetes Clusters

Description

This resource is used to return all clusters, as specified in the data forwarder.

No filtering is applied.

This endpoint does not require the analysis ID that is required when using the `/systems` and `/results` endpoints.

Resources

- `/kubernetes/clusters`
- `/kubernetes/clusters/<clusterName>`

Supported Operations

Table: Kubernetes Container Systems Supported Operations

HTTP Method	Input	Output	Description
`GET / kubernetes/clusters`	Path Parameter: N/A	Collection of: - cluster - firstCollectionTime - lastCollectionTime - nodeCount - nodeGroupCount - containerCount - kubexAgentVersion - kubernetesVersion - prometheusVersion	Use this resource to return a list of clusters from which data is being collected.

HTTP Method	Input	Output	Description
GET / kubernetes/clusters/< clusterName>	Path Parameter: clusterName	Collection of: - cluster - firstCollectionTime - lastCollectionTime - nodeCount - nodeGroupCount - containerCount - kubexAgentVersion - kubernetesVersion - prometheusVersion - namespace - podOwnerName - containers	Use this resource to retrieve additional details for a specific Kubernetes cluster. This endpoint returns a structured view, including namespaces, pods, and containers.

Parameters

Path Parameters

Table: Kubernetes Clusters Path Parameters

Parameter Name	Type	Description
clusterName	string	The unique cluster name obtained from the output of /kubernetes/clusters or from the Kubex UI Connections tab.

Response

Table: Kubernetes Container Clusters Response Schema

Element	Type	Filter/Sort	Description
cluster	string		The cluster name as specified in the config.yaml files.
firstCollectionTime	date		The date and time on which data collection started, in the format: YYYY-MM-DDTHH:mm:ssZ.
lastCollectionTime	date		The date and time of the most recent data collection, in the format: YYYY-MM-DDTHH:mm:ssZ.
nodeCount	string		The number of nodes discovered in this cluster.
nodeGroupCount	string		The number of node groups discovered in this cluster.
containerCount	string		The number of containers discovered in this cluster.
kubexAgentVersion	string		The data forwarder version.
kubernetesVersion	string		The Kubernetes version.
prometheusVersion	string		The Prometheus version.
namespace	string		The name of the namespace.
podOwnerName	string		The pod owner's name.
containers	string		The names of the containers in the selected pod.

Examples

Example: Listing All Kubernetes Clusters

The following example shows you how to return all clusters included in data collection.

Example: Listing All Clusters

Request:

```
GET {{scheme}}://{{host}}{{port}}/api/v2/kubernetes/clusters
```

Response:

Example: Listing All Clusters

```
 1  {
 2      "cluster": "eks_oss_dev",
 3      "firstCollectionTime": "2025-02-03T09:00:00Z",
 4      "lastCollectionTime": "2025-03-18T011:00:00Z",
 5      "nodeCount": 10",
 6      "nodeGroupCount": 5",
 7      "ContainerCount": 100",
 8      "kubexAgentVersion": "v4.1.2",
 9      "kubernetesVersion": "v1.31.5",
10      "prometheusVersion": "2.54.1"
11
12  },
13
14  {
15      "cluster": "gke-arm64",
16      "firstCollectionTime": "2025-02-12T19:00:00Z",
17      "lastCollectionTime": "2025-03-18T011:00:00Z",
18      "nodeCount": 3",
19      "nodeGroupCount": 2",
20      "ContainerCount": 33",
21      "kubexAgentVersion": "v4.1.2",
22      "kubernetesVersion": "v1.31.6-gke.100",
23      "prometheusVersion": "2.54.1"
24  },
25
26  {
27  ...
28  },
29
```

Example: Listing Specific Cluster Details

The following example shows you how to return details for a specific named cluster.

Example: Listing Specific Cluster Details

Request:

```
GET {{scheme}}://{{host}}{{port}}/api/v2/kubernetes/clusters/eks_oss_dev
```

Response:

Example: Listing Specific Cluster Details

```
 1  {
 2      "cluster": "eks_oss_dev",
 3      "firstCollectionTime": "2025-02-03T09:00:00Z",
 4      "lastCollectionTime": "2025-03-18T011:00:00Z",
 5      "nodeCount": 10",
 6      "nodeGroupCount": 5",
 7      "ContainerCount": 100",
 8      "kubexAgentVersion": "v4.1.2",
 9      "kubernetesVersion": "v1.32.2-eks-bc803b4",
10      "prometheusVersion": "2.54.1",
11      "namespaces": [
12          {
13              "namespace": "production",
14              "pods": [
15                  {
16                      "podOwnerName": "webserver-1",
17                      "containers": [
18                              "apache", "security-agent"
19                      ]
20                  }
21              ]
22          },
23          {
24              "namespace": "kube-system",
25              "pods": [
26                  {
27                      "podOwnerName": "aws-node",
28                      "containers": [
29                              "aws-node",
30                              "aws-eks-nodeagent"
31                      ]
32                  },
33                  {
34                      "podOwnerName": "densify-job",
35                      "containers": [
36                      "data-forwarder"
37                      ]
38                  }
39              ]
40          },
41              ....
42          }
43      ]
44  }
45
```

Kubernetes Clusters Results

Description

This resource is used to return container recommendations at the cluster level. This endpoint shows all analyzed containers in the specified cluster.

Resources

- `/kubernetes/clusters/<clusterName>/containers`
- `/analysis/containers/kubernetes/clusters/<clusterName>/containers?details=true`

Supported Operations

Table: Kubernetes Clusters Results Supported Operations

HTTP Method	Input	Output	Description
`GET / kubernetes/clusters/<clusterName>/containers`	Path Parameter: ClusterName	Collection of: - cluster - namespace - podOwnerName - podOwnerKind - container - entityId - containerId - avgContainerCount	Use this resource to retrieve details of container recommendation for a specific Kubernetes cluster.

HTTP Method	Input	Output	Description
		• currentCpuRequestmCores • currentCpuLimitBytes • currentMemRequestmCores • currentMemLimitBytes • recommendedCpuRequestmCores • recommendedCpuLimitmCores • recommendedMemRequestBytes • recommendedMemLimitBytes • estimatedSavingsPerContainer • analyzedOn	

`GET /`
`kubernetes/clusters/<`
`clusterName`
`>/containers?details=true`
	Path Parameter:
	
	ClusterName
		In addition to the above-listed properties the following additional details are included:
		
		• predictedUptime
		• configLastChangedOn
		• nodeGroup
		• oomKills_last7days
		• dateFirstAudited
		• dateLastAudited
			Use this resource to retrieve additional details for a specific Kubernetes cluster.

Parameters

Path Parameters

Table: Kubernetes Clusters Path Parameters

Parameter Name	Type	Description
ClusterName	string	The unique cluster name obtained from the output of `/kubernetes/clusters` or from the Kubex UI Connections tab.

Response

If a property or value is not set or not available, it is not included in the response.

Table: Kubernetes Cluster Response Schema

Element	Type	Filter/Sort	Description
cluster	string		The cluster name as specified in the config.yaml files.
namespace	string		The name of the namespace associated with the selected container.
podOwnerName	string		This is the name of the controller or pod. You can have one or more containers per pod.
podOwnerKind	string		The controller type that monitors and maintains the state of your cluster. This can be one of DaemonSet, Deployment, ReplicaSet,

Element	Type	Filter/Sort	Description
			ReplicationController, StatefulSet, etc.
container	string		This is the name of the container manifest.
entityId	string		A unique identifier, for the selected cluster assigned by Densify.
containerId	string		A unique identifier for the listed container.
avgContainerCount	integer		This is the number of containers specified by the selected manifest. The number is determined using an average number of containers, based on the metric, "In Service Instances".
currentCpuRequestmCores	integer		The current CPU request value for the listed container.
currentCpuLimitmCores	integer		The current CPU limit value for the listed container.
currentMemRequestBytes	integer		The current memory request value for the listed container.
currentMemLimitBytes	integer		The current memory limit value for the listed container.
recommendedCpuRequestmCores	integer		The recommended CPU request value for the listed container.
recommendedCpuLimitmCores	integer		The recommended CPU limit value for the listed container.
recommendedMemRequestBytes	integer		The recommended memory request value for the listed container.
recommendedMemLimitBytes	integer		The recommended memory limit value for the selected container.
estimatedSavingsPerContainer	integer		The estimated savings that can be achieved by using the recommended CPU and memory settings for this container.
analyzedOn	date		The date and time of the most recent analysis refresh, in the format: YYYY-MM-DDTHH:mm:ssZ

Table: Kubernetes Cluster Response Schema - Additional Details

Element	Type	Filter/Sort	Description
predictedUptime	string		The predicted uptime (%) for the container, is based on the percentage of hours CPU utilization data is present in the historical interval, as specified in the policy settings for the entity.
configLastChangedOn	date		A value in this column indicates the instance request/limit values have been changed recently. The date and time are in the format: YYYY-MM-DDTHH:mm:ssZ. The workload for this instance, on which the recommendation is based, includes all historical data for this instance and is not included in the analysis.
nodeGroup	string		This contains the names of the associated node groups.
oomKills_last7days	string		The number of out of memory (oom) events that occurred in the last 7 days.
dateFirstAudited	date		The date and time on which this container was last audited, in the format: YYYY-MM-DDTHH:mm:ssZ. The value is from the workload history instead of the audit history, which

Element	Type	Filter/Sort	Description
			may be truncated.
dateLastAudited	date		The date and time on which this container was first audited, in the format: YYYY-MM-DDTHH:mm:ssZ. The value is from the workload history instead of the audit history, which may be truncated.

Examples

Example: Listing Recommendations for Pods in a Specified Cluster

The following example shows you how to retrieve recommendation details for the container manifests in the specified cluster. This command shows all analyzed containers in the specified cluster.

Example: Listing Recommendations for Containers in a Specified Cluster

Request:

```
GET {{scheme}}://{{host}}{{port}}/api/v2/kubernetes/clusters/eks_oss_dev/-
containers
```

Response:

Example: Listing Recommendations for Containers in a Specified Cluster

```
 1
 2   [
 3        {
 4            "cluster": "eks_oss_dev",
 5            "namespace": "dss",
 6            "podOwnerName": "main-entity-operator",
 7            "podOwnerKind": "Deployment",
 8            "container": "user-operator",
 9            "entityId": "aa5c013d-dfb8-45e2-a05a-cf6a35005079",
10            "containerId": "583719fc-9fb4-32fa-8d38-19aa539840c6",
11            "avgContainerCount": 2,
12            "currentCpuRequestmCores": 10,
13            "currentCpuLimit": 2500,
14            "currentMemRequestBytes": 40000000,
15            "currentMemLimitBytes": 256000000,
16            "recommendedCpuRequestmCores": 10,
17            "recommendedCpuLimitmCores": 10,
18            "recommendedMemRequestBytes": 20000000,
19            "recommendedMemLimitBytes": 30000000,
20            "estimatedSavingsPerContainer": 33.34,
```

Example: Listing Recommendations for Containers in a Specified Cluster

```
21          "analyzedOn": "2025-03-19T04:33:40Z,
22      },
23
24      {
25          "cluster": "eks_oss_dev",
26          "namespace": "prometheus",
27          "podOwnerName": "prometheus-server",
28          "podOwnerKind": "Deployment",
29          "container": "prometheus-server-configmap-reload",
30          "entityId": "05e67e8c-8e45-4a3f-b4f4-07e1ac718275",
31          "containerId": "e5dd3d73-40c7-31b2-9877-903a6e6e3929",
32          "avgContainerCount": 1,
33          "recommendedCpuRequestmCores": 10,
34          "recommendedCpuLimitmCores": 10,
35          "recommendedMemRequestBytes": 20000000,
36          "recommendedMemLimitBytes": 80000000,
37          "estimatedSavingsPerContainer": -0.29,
38          "analyzedOn": "2025-03-19T04:33:40Z"
39      },
40
41      {
42          …
43      }
44  ]
45
```

Example: Listing Recommendations for Pods in a Specified Cluster with Additional Details

The following example shows you how to return details for a specific cluster.

Example: Listing Recommendations for Pods in a Specified Cluster with Additional Details

Request:

```
GET {{scheme}}://{{host}}{{port}}/api/v2/kubernetes/clusters/eks_oss_dev
```

Response:

Example: Listing Recommendations for Pods in a Specified Cluster with Additional Details

```
1  [
2      {
3          "cluster": "eks_oss_dev",
4          "namespace": "densify",
5          "podOwnerName": "kubex-prometheus-node-exporter",
6          "podOwnerKind": "DaemonSet",
7          "container": "node-exporter",
```

Example: Listing Recommendations for Pods in a Specified Cluster with Additional Details

```
 8            "entityId": "11aeafc0-4d1f-4f2c-a69f-cd68cd9c4abb",
 9            "containerId": "7552dd71-d546-3dae-b709-fad40d729a6f",
10            "avgContainerCount": 3,
11            "recommendedCpuRequestmCores": 10,
12            "recommendedCpuLimitmCores": 10,
13            "recommendedMemRequestBytes": 20000000,
14            "recommendedMemLimitBytes": 30000000,
15            "estimatedSavingsPerContainer": -0.06,
16            "analyzedOn": "2025-03-19T04:33:40Z",
17            "predictedUptime": 21.54,
18            "nodeGroup": [
19                    "test_node_group",
20                    "test_node_group_2"
21            ],
22            "oomKills_last7days": 0,
23            "dataCollection": {
24                "dateFirstAudited": "2024-12-27T12:00:00Z",
25                "dateLastAudited": "2025-01-30T17:00:00Z"
26            }
27
28        },
29
30        {
31        ...
32        }
33        ]
34
```

Kubernetes Clusters Automation

Description

This resource is used to return container recommendations for Kubernetes cluster which have been selected for automation via the UI.

Resources

```
/kubernetes/clusters/<clusterName>/automation
```

Supported Operations

Table: Kubernetes Container Systems Supported Operations

HTTP Method	Input	Output	Description
`GET / kubernetes/clusters/< clusterName >/containers`	Path Parameter: ClusterName	Collection of: • cluster • namespace • podOwnerName • podOwnerKind • container • entityId • containerId • avgContainerCount • currentCpuRequestmCores • currentCpuLimitmCores	Use this resource to retrieve details of container recommendation for a specific Kubernetes cluster.

HTTP Method	Input	Output	Description
		• currentMemRequestBytes • currentMemLimitBytes • recommendedCpuRequestmCores • recommendedCpuLimitmCores • recommendedMemRequestBytes • recommendedMemLimitBytes • estimatedSavingsPerContainer • analyzedOn • kubexAutomation • automationPolicy • hpaMetricName • hpaMetricThreshold	

Parameters

Path Parameters

Table: Kubernetes Clusters Path Parameters

Parameter Name	Type	Description
ClusterName	string	The unique cluster name obtained from the output of `/kubernetes/clusters` or from the Kubex UI Connections tab.

Response

If a property or value is not set or not available, it is not included in the response.

Table: Kubernetes Container Response Schema

Element	Type	Filter/Sort	Description
cluster	string		The cluster name as specified in the config.yaml files.
namespace	string		The name of the namespace associated with the selected container.
podOwnerName	string		This is the name of the controller or pod. You can have one or more containers per pod.
podOwnerKind	string		The controller type that monitors and maintains the state of your cluster. This can be one of DaemonSet, Deployment, ReplicaSet, ReplicationController, StatefulSet, etc.
container	string		This is the name of the container manifest.
entityId	string		A unique identifier, for the selected cluster assigned by Densify.
containerId	string		A unique identifier for the listed container.
avgContainerCount	integer		This is the number of containers specified by the

Element	Type	Filter/Sort	Description
			selected manifest. The number is determined using an average number of containers, based on the metric, "In Service Instances".
currentCpuRequestmCores	integer		The current CPU request value for the listed container.
currentCpuLimitmCores	integer		The current CPU limit value for the listed container.
currentMemRequestBytes	integer		The current memory request value for the listed container.
currentMemLimitBytes	integer		The current memory limit value for the listed container.
recommendedCpuRequestmCores	integer		The recommended CPU request value for the listed container.
recommendedCpuLimitmCores	integer		The recommended CPU limit value for the listed container.
recommendedMemRequestBytes	integer		The recommended memory request value for the listed container.
recommendedMemLimitBytes	integer		The recommended memory limit value for the selected container.
estimatedSavingsPerContainer	integer		The estimated savings that can be achieved by using the recommended CPU and memory settings for this container.
analyzedOn	date		The date and time of the most recent analysis refresh, in the format: YYYY-MM-DDTHH:mm:ssZ
kubexAutomation	string		Indicates whether automation has been enabled for this cluster.
automationPolicy	string		The number of node groups discovered in this cluster.
hpaMetricName	string		Indicates the metric used to trigger autoscaling.
hpaMetricThreshold	string		Indicates the metric threshold value used to trigger autoscaling.

Table: Kubernetes Container Response Schema - Additional Details

Element	Type	Filter/Sort	Description
predictedUptime	string		The predicted uptime (%) for the container, is based on the percentage of hours CPU utilization data is present in the historical interval, as specified in the policy settings for the entity.
configLastChangedOn	date		A value in this column indicates the instance request/limit values have been changed recently. The date and time are in the format: YYYY-MM-DDTHH:mm:ssZ. The workload for this instance, on which the recommendation is based, includes all historical data for this instance and is not included in the analysis.
nodeGroup	string		This is the name of the associated node group.
oomKills_last7days	string		The number of out of memory (oom) events that occurred in the last 7 days.
dateFirstAudited	date		The date and time on which this container was last audited, in the format: YYYY-MM-DDTHH:mm:ssZ. The value is

Element	Type	Filter/Sort	Description
			from the workload history instead of the audit history, which may be truncated.
dateLastAudited	date		The date and time on which this container was first audited, in the format: YYYY-MM-DDTHH:mm:ssZ. The value is from the workload history instead of the audit history, which may be truncated.

Examples

Example: Listing Kubernetes Clusters with Containers Enabled for Automation

The following example shows you how to retrieve cluster details for containers that have been enabled for automation.

Example: Listing Kubernetes Clusters with Containers Enabled for Automation

Request:

```
GET {{scheme}}://{{host}}{{port}}/api/v2/kubernetes/clusters/eks_oss_dev/-
containers
```

Response:

Example: Listing Kubernetes Clusters with Containers Enabled for Automation

```
 1  {
 2      "checksum": "d8d10f84c2829a3d-
    b166064ce2377af984f93254855a9e486633bf4cff1399f3",
 3      "results": [
 4          {
 5              "cluster": "eks_oss_dev",
 6              "namespace": "apps",
 7              "podOwnerName": "nginx-deployment",
 8              "podOwnerKind": "Deployment",
 9              "container": "nginx",
10              "entityId": "51d4f114-9de5-4639-b810-a6749edb19dc",
11              "containerId": "d8121526-9757-371b-8bd7-abc3cb83c6b7",
12              "avgContainerCount": 3,
13              "recommendedCpuRequestmCores": 10,
14              "recommendedCpuLimitmCores": 10,
15              "recommendedMemRequestBytes": 10000000,
16              "recommendedMemLimitBytes": 10000000,
17              "estimatedSavingsPerContainer": -0.25,
18              "analyzedOn": "2025-03-19T04:40:22Z",
```

Example: Listing Kubernetes Clusters with Containers Enabled for Automation

```
19              "kubexAutomation": "true",
20              "automationPolicy": "true"
21
22          },
23
24          {
25              "cluster": "eks_oss_dev",
26              "namespace": "kube-system",
27              "podOwnerName": "ama-metrics-ksm",
28              "podOwnerKind": "Deployment",
29              "container": "ama-metrics-ksm",
30              "entityId": "f06b2383-cf25-4c55-a9b0-ba0782daaf67",
31              "containerId": "aaef5810-2aed-3729-99f4-0796f3de921e",
32              "avgContainerCount": 1,
33              "currentCpuRequestmCores": 5,
34              "currentCpuLimitmCores": 1000,
35              "currentMemRequestBytes": 50000000,
36              "currentMemLimitBytes": 5120000000,
37              "recommendedCpuRequestmCores": 10,
38              "recommendedCpuLimitmCores": 110,
39              "recommendedMemRequestBytes": 40000000,
40              "recommendedMemLimitBytes": 520000000,
41              "estimatedSavingsPerContainer": -0.05,
42              "analyzedOn": "2025-03-19T04:40:22Z",
43              "kubexAutomation": "true",
44              "hpaMetricName": "cpu utilization"
45          },
46
47          {
48              ...
49          }
50  ]
51
```

Analysis: Webhook

Description

The `/webhook/analysis` resource is used to manage the webhook definition for a Densify analysis. The webhook definition is the path and authentication to a third-party application, where recommendation results are sent when the analysis completes. The receiving application can process the resulting data for downstream consumption. For example, the receiving application can route recommendations to an orchestration engine or distribute reports to instance owners.

You can define the webhook when you initiate data collection and analysis with the `/analysis/<platformType>/<platformSubType>/analyze` request. See the example, Run AWS data collection and analysis on page 72 for details of specifying webhook input parameters. Results are posted to the webhook location when the analysis is complete. You can edit or delete a webhook at any time.

Resource

```
/webhook/analysis
```

Supported Operations

Table: Webhook Supported Operations

Operation	HTTP Method	Input	Output	Description
List all analyse	`GET /webhook/analysis/<platformType>`	Path Parameter: 🔖 platformType	Collection of: 🔖 analysisId	Use this resource to

Operation	HTTP Method	Input	Output	Description
s webhooks for a specific platform type			analysisName href uri webHookStatus	list all analysis webhook definitions for a particular platform type in Densify. Example: Retrieving All Cloud Webhook Definitions Example: Retrieving All Container Webhook Definitions
Get a webhook definition for an analysis	`GET /webhook/analysis/` `<platformType>/<platformSubType>/<analysisId>`	Path Parameters: platformType platformSubType analysisId	analysisId analysisName href uri webHookStatus	Use this resource to return the webhook details of a specific analysis in Densify. Example: Getting a Webhook Definition for an AWS Analysis
Add webhook to an analysis	`POST /webhook/analysis/` `<platformType>/<platformSubType>/<analysisId>`	Path Parameters: platformType platformSubType analysisId Request Body Parameters: uri authType authValue	message status	Use this resource to add a webhook definition to an existing analysis in Densify. You can only add a webhook to an analysis without a webhook. To update a webhook,

Operation	HTTP Method	Input	Output	Description
				use the `PUT` method. See Update a webhook for an analysis for details. Example: Adding a Webhook to an Existing AWS Analysis
Update a webhook for an analysis	`PUT /webhook/analysis/` `<platformType>/<platformSubType>/<analysisId>`	Path Parameters: ■ platformType ■ platformSubType ■ analysisId Request Body Parameters: ■ uri ■ authType ■ authValue	■ message ■ status	Use this resource to update a webhook definition of an existing analysis in Densify. Example: Updating an AWS Analysis Webhook
Delete a webhook from an analysis	`DELETE /webhook/analysis/` `<platformType>/<platformSubType>/<analysisId>`	Path Parameters: ■ platformType ■ platformSubType ■ analysisId	■ message ■ status	Use this resource to delete a webhook definition from an existing analysis. Example: Deleting a Webhook from an AWS Analysis

Parameters

Path Parameters

Table: Webhook Path Parameters

Parameter Name	Type	Description
platformType	string	[cloud|containers] The technology platform of the analysis.
platformSubType	string	The platform sub-type of the analysis. This is the vendor for the cloud platform. Possible platform sub-types depend on the platformType specified: ▪ For platformType = cloud, platformSubType = [aws|gcp|azure] ▪ For platformType = containers, platformSubType = [kubernetes]
analysisId	string	The referenced ID of the analysis.

Request Body Parameters

Table: Webhook Request Body Parameters

Parameter Name	Type	Description
uri	string	The URI to the webhook application. *Note:* *You must ensure that the webhook URI is accessible from the Densify system with a fully qualified domain name or an IP address.*
authType	string	The authorization type (i.e. Basic, Bearer) for the webhook application.
authValue	string	The authorization value (i.e. username, password, token) for the webhook application. Densify assumes that the authorization value is in standard Base64 encoding, but if a colon (":") is present in the value, then Densify assumes that the value is in plain text with `username:password` format.

Response

Table: Webhook Response Schema

Element	Type	Sort By/Filter	Description
analysisId	string		The Densify assigned ID for the analysis entity.
href	string		The analysis webhook resource reference.
uri	string		The webhook URI.

Element	Type	Sort By/Filter	Description
webHookStatus	string		The status of the last webhook POST request. Recommendations are pushed to the webhook URI via a POST request after analysis completion.
analysisName	string		The name assigned to the analysis. The default analysis name is the platform-specific cluster, account, project, or subscription ID.
message	string		The message for the status response. For example, the following messages are returned for the error codes below: ▪ 400–"Analysis webhook already exists." ▪ 404–"Analysis webhook not found."
status	number		The HTTP response code of the request. Possible status values include: ▪ 200–success with webhook request; ▪ 400–object already exists or invalid parameters; ▪ 401–authentication failed; ▪ 404–resource not found.

Examples

Example: Retrieving All Cloud Webhook Definitions

The following example shows you how to list all cloud webhook definitions in Densify.

Example: Listing All Cloud Webhook Definitions

Request:

```
GET /webhook/analysis/cloud
```

Response:

```
[
    {
      "analysisId": "f73e13ca-09d3-4395-8e32-894669a6d1eb",
      "analysisName": "551345225429"
      "href": "/webhook/analysis/cloud/aws/f73e13ca-09d3-4395-8e32-
894669a6d1eb",
      "uri": "https://webhookdensify:443/results/display/",
      "webHookStatus": "success"
    },
    {
      "analysisId": "329959e9-3641-4677-a665-dff754943944",
      "analysisName": "330076083786"
      "href": "/webhook/analysis/cloud/aws/329959e9-3641-4677-a665-
dff754943944",
      "uri": "https://webhooknow:443/api/results/test",
      "webHookStatus": "An error occured executing POST request"
    }
```

```
]
```

Example: Retrieving All Container Webhook Definitions

The following example shows you how to list all container webhook definitions in Densify.

Example: Listing All Container Webhook Definitions

Request:

```
GET /webhook/analysis/containers
```

Response:

```
[
    {
        "analysisId": "e8298ac3-a143-41bb-b7d7-62f659f43ec5",
        "analysisName": "kube-prod-341ops",
        "href": "/webhook/analysis/containers/kubernetes/e8298ac3-a143-41bb-b7d7-
62f659f43ec5",
        "uri": "https://webhookdensify:443/results/display/",
        "webHookStatus": "success"
    },
    {
        "analysisId": "d32959e9-3641-4677-a665-dff754943945",
        "analysisName": "kube-dev-543",
        "href": "/webhook/analysis/containers/kubernetes/d32959e9-3641-4677-a665-
dff754943945",
        "uri": "https://mycontainerresources:443/api/results/",
        "webHookStatus": "An error occured executing POST request"
    }
]
```

Example: Getting a Webhook Definition for an AWS Analysis

The following example shows you how to return the details of a webhook definition for an AWS analysis.

Example:Retrieving Webhook Definition for an AWS Analysis

Request:

```
GET /webhook/analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf
```

Response:

```
{
  "analysisId": "9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf",
  "analysisName": "775525148221",
  "href": "/webhook/analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf",
  "uri": "https://Dwebhooks:443/api/test/webhook",
  "webHookStatus": "success"
}
```

Example: Adding a Webhook to an Existing AWS Analysis

The following example shows you how to add a webhook definition to an existing AWS analysis without a webhook.

Example: Adding a Webhook Definition to an Existing AWS Analysis

Request:

```
POST /webhook/analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf
{
  "uri": "https://mycallbackServer:443/api/test/webhook",
  "authType": "basic",
  "authValue": "saasUser:password1"
}
```

Response:

```
{
  "message": "ok",
  "status": 200
}
```

Example: Updating an AWS Analysis Webhook

The following example shows you how to update a webhook definition for an AWS analysis.

Example:Updating a Webhook Definition for an AWS Analysis

Request:

```
PUT /webhook/analysis/cloud/aws/9a5d2d55-6d85-4fde-8bab-fcd0cef8c5bf
{
  "uri": "https://myOtherServer:443/api/webhook",
  "authType": "bearer ",
  "authValue":
"eyJhbGciOiJIUzUxMiJ9.eyJqdGkiOiI0MjRhMGziOC0xMDRlopRiMjItOeg0ZS1lZTJhNmEzZDBlN
GIiLCJpYXQiOjE1NDM2MDg1OTMsInN1YiI6ImFkbWluIiwiaXNzIjoiRGVuc2lmeS5jb20iLCJleHAi
OjE1NDM2MDg4OTN9.F9VgD918C6WufBajr0ezLd61T6d9ij8z4BmHFzfNmMqCS-
9JTDDaxfPmQVZDjeSTo0C-dYWcllwPTcMSUfRnYQ"
}
```

Response:

```
{
  "message": "ok",
  "status": 200
}
```

Example: Deleting a Webhook from an AWS Analysis

The following example shows you how to delete a webhook definition from an AWS analysis.

Example:Deleting a Webhook Definition

Request:

```
DELETE /webhook/analysis/cloud/aws/329959e9-3641-4677-a665-dff754943944
```

Response:

```
{
  "message": "ok",
  "status": 200
}
```

Receive Metrics Jobs

Description

The Receive Metrics Jobs resource is part of a series of data management tools using the Data Ingestion API framework for users on SaaS deployments to transfer utilization data into Densify.

The Data Ingestion API framework allows you to customize and extend data transfer capabilities (such as transferring metrics, configuration, attributes, or commands) for Densify SaaS deployments in a secure job execution framework. The framework encapsulates a custom data management tool into a Job entity in Densify, where you can create, list, schedule, execute, and delete. When you schedule the Densify job for execution, the custom data management tool is invoked. The framework also allows you to upload and download job artifacts, such as data or log files.

By default, Densify is installed with the Metrics data management tool which allows SaaS users to invoke a standard Receive Metrics Jobs resource from the Densify REST API. This resource allows you to upload a metrics file into a Densify job and schedule the job for processing, which transfers the metrics data to their respective services for analysis. You can also use the Receive Metrics Jobs resource to download any result files or logs of the job execution.

Resource

```
/receive/metrics/jobs/
```

Supported Operations

Table: Receive Metrics Jobs Supported Operations

Operation	HTTP Method	Request Input Parameters	Response Elements	Description
Create a job	POST /receive/metrics/jobs	Request Body Parameters: ■ [name] ■ [parameters]	■ jobId ■ name ■ job_status	Creates a job in Densify with the support structure for uploading utilization metrics. If the [name] parameter is not supplied, then the job's name will be set to the job GUID. See Example: Create a Job.
Get all jobs	GET /receive/metrics/jobs	None	Collection of jobs: ■ jobId ■ name	Returns a list of all jobs in Densify. See Example: Get All Jobs.
Get details of an individual job	GET /receive/metrics/jobs/ <jobId>?[lines_to_tail_ in_logs=n]	Path Parameter: ■ jobId Query String Parameter: ■ [lines_to_ tail_in_logs]	■ jobId ■ name ■ parameters ■ audit_info ■ input_files ■ lines_to_tail_ in_logs ■ logs ■ job_status	Returns the details of the job specified by job ID provided in the request. See Example: Get Specific Job Details.
Get job input file details	GET /receive/metrics/jobs/ <jobId>/input	Path Parameter: ■ jobId	■ jobId ■ name ■ parameters ■ input_files	Returns a list of input files for the job. See Example: Get Job Input File Details.
Get job log file details	GET /receive/metrics/jobs/ <jobId>/logs?[lines_to_ tail_in_logs=n]	Path Parameter: ■ jobId Query String Parameter: ■ [lines_to_ tail_in_logs]	■ jobId ■ name ■ parameters ■ logs	Returns a list of log files for the job. See Example: Get Job Log File Details.
Get job audit details	GET /receive/metrics/jobs/ <jobId>/audit_info	Path Parameter: ■ jobId	■ jobId ■ name ■ parameters ■ audit_info	Returns audit information for the job specified. See Example: Get Job Audit Information.
Upload a file to a job	POST /receive/metrics/jobs/ <jobId>?[execute=true]& [time=HH:MM]	Path Parameter: ■ jobId Query String Parameters:	■ jobId ■ name ■ job_status	Uploads a file into an existing job. See Example: Upload a File.

Operation	HTTP Method	Request Input Parameters	Response Elements	Description
		[execute =true] [time =HH:MM] Request Body Parameter: file		
Download files in a job	GET /receive/metrics/jobs/ <jobId>/download/files? [file]	Path Parameter: jobId Query String Parameters: [file]	octet-stream zipped file containing the downloaded files See Download Files on page 240.	This request returns all files designated for download of the specified job. You also have the option to download a specific file, if the file name is known. See Example: Download Files.
Download logs in a job	GET /receive/metrics/jobs/ <jobId>/logs/files? [file]	Path Parameter: jobId Query String Parameters: [file]	octet-stream zipped file containing log files See Download Files on page 240.	This request returns all log files of the specified job. You also have the option to download a specific log file if the filename is known. See Example: Download Logs.
Delete job input files	DELETE /receive/metrics/jobs/ <jobId>/contents/input	Path Parameter: jobId	message status	Delete all input files associated with the specified job. See Example: Delete Input Files.
Delete job log files	DELETE /receive/metrics/jobs/ <jobId>/contents/logs	Path Parameter: jobId	message status	Delete all log files associated with the specified job. Use this request to clean up the job log files on the Densify server. See Example: Delete Log Files.
Delete a job	DELETE /receive/metrics/jobs/ <jobId>	Path Parameter: jobId	message status	Deletes all content associated with the job (i.e. input, download, and log files) and removes the job from the scheduled job list. See Example: Delete Job.
Update parameters of an existing job	PUT /receive/metrics/jobs/ <jobId>/parameters	Path Parameter: jobId Request Body	jobId name parameters input_files	Updates the parameter attributes of an existing job.

Operation	HTTP Method	Request Input Parameters	Response Elements	Description
		Parameters: ■ [parameters]	■ lines_to_tail_in_logs ■ logs ■ job_status	See Example: Update Job Parameters.

Parameters

The following is a complete list of possible parameters for the `/receive/metrics/jobs` endpoint. Path, query string, and/or request body parameters are required depending on the method requested.

Path Parameters

Table: Receive Metrics Jobs - Path Parameters

Parameter Name	Type	Description
jobId	string	Specify the job GUID to identify the job.

Request Body Parameters

Table: Receive Metrics Jobs - Request Body Parameters

Parameter Name	Type	Description

name
 	
(optional)
	string	The name of the job.
		
		During job creation, if the `name` parameter is not set, then `name` will be automatically set to the job globally unique identifier (GUID).
		
		Example of setting the job name:
		
		```{"name": "my-sample-job"}```
		
		If you do not want to specify any parameters for creating a job, you must still provide an empty JSON body element when using the POST operation:
		
		```POST /receive/metrics/jobs```
		```{ }```

parameters
 	
(optional, depending on the metrics custom endpoint)
	Array of name-value pairs	The job `parameters` element is an array of `"name"`, `"value"` pairs that you can provide in the request body.
		
		The `"parameters"` element needs to be provided in the following JSON format:
		
		```"parameters":```
		```[```
		``` {"name": <string>,```
		``` "value": <string>},```
		``` ...```
		```]```
		
		The parameters required during job creation is dependent on the

Parameter Name	Type	Description

		metrics custom endpoint used. Below is an example of setting optional parameters during job creation:
		
		``json
		{
		  "name" : "My new job",
		  "parameters":
		  [
		    {
		      "name": "JobPriority",
		      "value": "urgent"
		    },
		    {
		      "name": "Licence",
		      "value": "true"
		    },
		    {
		      "name": "Area",
		      "value": "NewArea"
		    }
		  ]
		}
		``
		
		In an update parameters request, you need to provide the complete list of parameters, including the updated ones and the non-updated ones. The new list of parameters used for the update command overwrites the entire old list of parameters.
		
		Below is an example of the request body for updating the "Area" parameter. Notice that the entire parameters list is provided.
		
		``json
		[
		  {
		    "name": "JobPriority",
		    "value": "urgent"
		  },
		  {
		    "name": "Licence",
		    "value": "true"
		  },
		  {
		    "name": "Area",
		    "value": "west-2b"
		  }
		]
		``

file	multipart/form-data	To upload a file into an existing job, attach the file into the `file` form-data key in the body of the POST request.

Query String Parameters

Table: Receive Metrics Jobs - Query String Parameters

Parameter Name	Type	Description
lines_to_tail_in_logs *(optional)*	integer	Specify the number of lines from the bottom of the log files to display. For example, to display the last 100 lines of logs, you would specify the following:

Parameter Name	Type	Description
		`lines_to_tail_in_logs=100`
		The default value of `-1` denotes that the entire log files will be displayed.
execute *(optional)*	string	The job `execute` option specifies whether the job should be executed with the uploaded file. Possible values for the `execute` option: ▪ `true`—The job is executed at the scheduled time provided by the [time] parameter. If [time] is not set and you specify `execute=true`, then the job is executed immediately. ▪ `false`—The file is uploaded with no job execution. This is the default behavior if no `execute` option is specified. You can use the `execute=false` option to upload multiple files to the job before executing the bulk job.
time *(optional)*	string	The job `time` option is used in conjunction with the [execute] option to specify when the job is to be executed next. The `time` value must be in `HH:mm` 24-hr format.

file
 	
(optional)
	string	The `file` option allows you to download a specific file.
		
		You must specify the exact download filename in order for this operation to succeed. For example, to download a file named "output.txt", the following call is made:
		`GET /receive/metrics/jobs/455fa7bb-10fb-41a7-96a9-f4b13bd7a05c/download/files?file=output.txt`
		
		Here is an example to download a log file named "output.log":
		`GET /receive/metrics/jobs/455fa7bb-10fb-41a7-96a9-f4b13bd7a05c/logs/files?file=output.log`

Response

The following is a complete list of possible response elements that are returned for the `/receive/metrics/jobs` resource. If the response element does not apply to the API request, then the element is not displayed in the results.

Table: Receive Metrics Jobs Response Schema

Element	Type	Description
jobId	string	The globally unique identifier (GUID) assigned to the job.
name	string	The name of the job. During job creation, if the `[name]` parameter is not set, then it will be automatically set to the job globally unique identifier (GUID).
parameters	Array of ▪ `name` ▪ `value`	The job `parameters` element is an array of `"name"`, `"value"` pairs that is dependent on the data management tool used. The Metrics tool is the default data management tool for the Receive Metrics Jobs resource.
audit_info	▪ `audit name` ▪ `audit date` ▪ `audit`	If the job has uploaded metric files which have been audited, the last audit details are displayed. The `audit_info` element displays the following information: `"audit_info": {`

Element	Type	Description

	path
	▪ target
	  audit
	▪ target
	  failed
	▪ audit
	  end date
	▪ load
	  date
	▪ load end
	  date
	▪ load
	  status
		``` "audit_name": <string>, "audit_date": <string>, "audit_path": <string>, "target_audit": <int>, "targets_failed": <int>, "audit_end_date": <string>, "load_date": <string>, "load_end_date": <string>, "load_status": <string>, } ```

input_files	Array of: ▪ name ▪ size	If the job has uploaded files, details of those files are displayed: ```"input_files": [{ "name": <string>, // name of the metrics file "size": <int>, // size of the file, in bytes }, ... { "name": <string>, "size": <int>, }] ```
lines_to_tail_in_logs	integer	Specifies the number of lines from the bottom of the log file to display. The default value of `-1` denotes that the entire log file will be displayed.
logs	▪ name ▪ contents	Contains the details of each file in the job's log directory. For each log file available, the following information is displayed: ▪ `name`–displays the log filename; ▪ `contents`–displays the tail end contents of the log file, depending on the lines_to_tail_in_logs parameter.

job_status	▪ code
	▪ message
	▪ files
		An element that provides the status of the executed Metrics job operation, which contains the following items:
		▪ `code`–Displays the status code from the statuscode.txt file in the job status folder. If the statuscode.txt file does not exist, then the last audit load code is displayed:
		  • `0`–successful operation;
		  • `-1`–an error with loading the job.
		▪ `message`–Displays the contents of the statusmessage.txt file in the job status folder. If the statusmessage.txt file does not exist, then the last audit load status message is displayed (e.g. Loaded, Loaded_Error).
		▪ `files`–Displays the details of each non-status file from the job status folder.
		
		The `job_status` element is displayed in the following form:
		
		```{"code": <int>, "message": <string>, "files": [{ "name": <string>, // filename "size": <int>, /size in bytes }, ```

Element	Type	Description
| | | ```
    ...
    ]
}
``` |
| | | Below is a response example if there are additional non-status files in the job status folder: |
| | | ```
"job_status": {
    "code": 0,
    "message": "File Loaded Successfully.",
    "files": [
        {
        "name": "production_status.zip",
        "size": 908756
        },
        {
        "name": "test_status.zip",
        "size": 453903
        }
    ]
}
``` |
		The default Metrics data management tool produces a statusmessage.txt file, a corresponding statuscode.txt file, and other status files in the job status directory for each job execution. The Data Ingestion API framework provides flexibility for a custom data management tool to produce their own status message, code, and files by generating files with the exact name (i.e. statusmessage.txt, statuscode.txt) in the custom tool's job status directory. *Note:* *File names are case sensitive. Both message and code files must exist in the job status directory for the framework to override the job_status element.*
Download Files Response		

Download Files	octet-stream zipped file containing files	The response for a file or log download request is an octet-stream zipped file containing all the files produced when executing the job. For a single file download, the zipped file will contain only one file. The suggested filename (in the response Content-Disposition) is in the following format:
		■ for download logs request:
		`{jobId}_logs.zip`
		■ for download file request:
		`{jobId}_download.zip`
		
		Where `{jobId}` is the job GUID.

Messages and Error Handling		
message	string	A response message for the request from the `job_status` element. The `job_status: message` string can be one of the following: ■ AccessDeniedException: "Access Denied! Unable to create temporary file or directory"–Check file permissions on file system. ■ IngestionJobDirectoryDoesNotExistException–Job file structure does not exist in the system. ■ IngestionJobDirectoryNotReadableException–Job file structure permission is possibly set to read-only. ■ IngestionJobInvalidArgumentException–Invalid arguments in the file upload request. ■ IOException: "There is not enough space on the disk. IO Exception"–Check disk space.

Element	Type	Description
		- IOException—Any I/O exception that occurred on the system, such as when the specified filename cannot be found. - Exception—Any other error that occurred during the processing of the upload request.
status	integer	The status code of the job request. Possible "`status`" values are: `200`—job action successful (i.e. delete action was successful); `404`—the job does not exist; `409`—the job is currently running, request action canceled; `400`—the job files are not accessible or permissions are invalid; `500`—the job is corrupt or any other exception.

HTTP Code 400	HTTP error	Request is malformed. For example, when the time format is invalid, the following error message is displayed:
		
		``{
		  "message": "Invalid schedule time format <HH:MM>.
		Schedule task cannot be updated or created.",
		  "status": 400
		}```

HTTP Code 403	HTTP error	"Limit number of active jobs has been reached." The number of active jobs on the server exceeds the Densify configuration setting of maximum job limit.
HTTP Code 404	HTTP error	The job does not exist in the system.
HTTP Code 500	HTTP error	"Working directory does not exist."—The Data Ingestion API root working directory does not exist. "OS error message."—Failed to create directory structure due to insufficient storage..

Examples

Example: Create a Job

The following example shows you how to create a job:

Example: Create a Job

Request:

```
POST /receive/metrics/jobs/
```

JSON Body:

```
{ "name": "sample-job" }
```

Response:

```
{
  "jobId": "483e2a46-1f40-4e8a-9ee2-532f66dc549b",
  "name": "sample-job",
  "job_status": {
      "code": 0,
      "message": "Job created successfully"
```

```
        }
}
```

Example: Get All Jobs

The following example shows you how retrieve all jobs:

Example: Get All Jobs

Request:

```
GET /receive/metrics/jobs
```

Response:

```
[
  {
  "jobId": "e79518b7-fe0f-4bd4-ae5a-2ca51974d77b",
  "name": "sample-job"
  },
  {
  "jobId": "58419075-6009-4c73-8783-54a49e347384",
  "name": "my-job"
  }
]
```

Example: Get Specific Job Details

The following example shows you how retrieve details of a specific job:

Example: Get Job Details

Request:

```
GET /receive/metrics/jobs/58419075-6009-4c73-8783-54a49e347384?lines_to_tail_
in_logs=2
```

Response:

```
{
  "jobId": "58419075-6009-4c73-8783-54a49e347384",
  "name": "CPU utilization - upload job",
  "parameters": [{
        "name": "hint",
        "value": "Sample"
      },
      {
        "name": "license",
        "value": "true"
      }
  ],
  "audit_info": {
      "audit_name": "58419075-6009-4c73-8783-54a49e347384",
      "audit_date": "2018-02-13 09:23:37.000",
      "audit_path": "output\00001518_M0500_58419075-6009-4c73-8783-
54a49e347384",
      "targets_audited": 323,
      "targets_failed": 0,
      "audit_end_date": "2018-02-13 09:45:11.333",
```

```
        "load_date": "2018-02-13 09:52:23.930",
        "load_end_date": "2018-02-13 09:52:26.577",
        "load_status": "Loaded"
    },
    "input_files": [{
        "name": "allmetrics_production.csv",
        "size": 203945
    }, {
        "name": "h2metrics_test.csv",
        "size": 19932
    }
    ],
    "lines_to_tail_in_logs": 2,
    "logs": [
        {
        "name": "main.log",
        "contents": "[INFO] [2017-10-10 17:37:33,682] 1)****************** end
testing RMI connectivity to thedocsmachine/192.168.2.123\n[INFO] [2017-10-10
17:37:33,682] \n"
        }
    ],
    "job_status": {
      "code": 0,
      "message": "Success"
    }
}
```

Example: Get Job Input File Details

The following example shows you how retrieve input file details of a specific job:

Example: Get Job Input File Details

Request:

```
GET /receive/metrics/jobs/483e2a46-1f40-4e8a-9ee2-532f66dc549b/input
```

Response:

```
{
  "jobId": "483e2a46-1f40-4e8a-9ee2-532f66dc549b",
  "name": "My Upload Job"
  "parameters": [{
        "name": "Priority",
        "value": "High"
    },
    {
        "name": "License",
        "value": "True"
    }
    ],
  "input_files": [{
        "name": "allmetrics_production.csv",
        "size": 203945
    }, {
        "name": "Eastmetrics_test.csv",
        "size": 28635
    }, {
        "name": "Normetrics.csv",
```

```
            "size": 27812
        }
    ],
}
```

Example: Get Job Log File Details

The following example shows you how retrieve log file details for a specific job:

Example: Get Job Log File Details

Request:

```
GET /receive/metrics/jobs/58419075-6009-4c73-8783-54a49e347384/logs?lines_to_
tail_in_logs=2
```

Response:

```
{
  "jobId": "58419075-6009-4c73-8783-54a49e347384",
  "name": "CPU utilization - upload job",
  "parameters": [{
        "name": "hint",
        "value": "Sample"
      },
      {
        "name": "license",
        "value": "true"
      }
  ],
  "logs": [
      {
        "name": "main.log",
        "contents": "[INFO] [2018-05-10 17:37:33,682] 1)****************** end
testing RMI connectivity to thedocsmachine/192.168.2.123\n[INFO] [2018-05-10
17:37:33,682] \n"
      },
      {
        "name": "logs_load.txt",
        "contents": "[INFO] [2018-05-10 09:45:10,403] Process
host:notApplicable, converter type:DataImporter\n[INFO] [2018-05-10
09:45:11,355] Audit Converter complete.\n"
      }
  ]
}
```

Example: Get Job Audit Information

The following example shows you how retrieve details of a specific job:

Example: Get Job Audit Information

Request:

```
GET /receive/metrics/jobs/483e2a46-1f40-4e8a-9ee2-532f66dc549b/audit_info
```

Response:

```
{
```

API Reference Guide for Public Cloud

```
  "jobId": "483e2a46-1f40-4e8a-9ee2-532f66dc549b",
  "name": "CPU utilization - manual upload",
  "parameters": [{
        "name": "region",
        "value": "north"
      },
      {
        "name": "license",
        "value": "true"
      }
  ],
  "audit_info": {
      "audit_name": "483e2a46-1f40-4e8a-9ee2-532f66dc549b",
      "audit_date": "2018-05-13 09:23:37.000",
      "audit_path": "output\002215185311_M0511_483e2a46-1f40-4e8a-9ee2-
532f66dc549b",
      "targets_audited": 323,
      "targets_failed": 0,
      "audit_end_date": "2018-05-13 09:45:11.333",
      "load_date": "2018-05-13 09:52:23.930",
      "load_end_date": "2018-05-13 09:52:26.577",
      "load_status": "Loaded"
  },
}
```

Example: Upload a File

The following example shows you how to upload a file to an existing job:

Example: Upload a File

Request:

```
POST /receive/metrics/jobs/8e5fafe4-1709-46b5-9bdb-
f0a12ed4910d?execute=false&time=00:00
```

The request body contains a multipart/form-data file with the `file` key:

```
{
  "file": <file content>
}
```

Response:

```
{
  "jobId": "8e5fafe4-1709-46b5-9bdb-f0a12ed4910d",
  "name": "sample-job",
  "job_status": {
      "code": 0,
      "message": "File uploaded successfully."
   }
}
```

Example: Download Files

The following example shows you how to download files from an existing job:

Example: Download Files

Request:

```
GET /receive/metrics/jobs/8e5fafe4-1709-46b5-9bdb-f0a12ed4910d/download/files
```

Response:

The response is an octet-stream with the Content-Disposition suggested filename in the following format:

```
<jobId>_download.zip
```

The downloaded zipped file contains all the files, designated as "download", for the specified job.

Example: Download Logs

The following example shows you how to download log files from an existing job:

Example: Download Logs

Request:

```
GET /receive/metrics/jobs/8e5fafe4-1709-46b5-9bdb-f0a12ed4910d/logs/files
```

Response:

The response is an octet-stream with the Content-Disposition suggested filename in the following format:

```
<jobId>_logs.zip
```

Example: Delete Input Files

The following example shows you how to delete the input files from an existing job:

Example: Delete Input Files

Request:

```
DELETE /receive/metrics/jobs/483e2a46-1f40-4e8a-9ee2-
532f66dc549b/contents/input
```

Response:

```
{
  "message": "Contents of directory deleted successfully.",
  "status": 200
}
```

Example: Delete Log Files

The following example shows you how to delete log files from an existing job:

Example: Delete Log Files

Request:

```
DELETE /receive/metrics/jobs/483e2a46-1f40-4e8a-9ee2-532f66dc549b/contents/logs
```

Response:

```
{
  "message": "Contents of directory deleted successfully.",
  "status": 200
}
```

Example: Delete Job

The following example shows you how to delete a job:

Example: Delete Job

Request:
```
DELETE /receive/metrics/jobs/883e2a46-1f40-4e8a-9ee2-532f66dc549b
```

Response:
```
{
  "message": "Job deleted successfully.",
  "status": 200
}
```

Example: Update Job Parameters

The following example shows you how to update parameters for an existing job:

Example: Update Job Parameters

Request:
```
PUT /receive/metrics/jobs/acb37a49-cd7c-4bf1-a94b-884fdce11ed7/parameters
```

JSON Request Body:
```
[
        {"name": "License",
         "value": "true"
        },
        {"name": "Priority",
         "value": "low"
        }
]
```

Response:
```
{
  "jobId": "acb37a49-cd7c-4bf1-a94b-884fdce11ed7",
  "name": "SuperDuperJob",
  "parameters": [{
        "name": "License",
        "value": "true"
      },
      {
        "name": "Priority",
        "value": "low"
      }
  ],
  "input_files": [],
  "lines_to_tail_in_logs": -1,
  "logs": [],
```

```
  "job_status": {
    "code": 0,
    "message": ""
  }
}
```

Systems

Description

This resource is used to return information about systems loaded through data collection in Densify. These systems do not have to belong to a control environment or a cloud environment, but must be of a supported entity type. Returned resource element values match those that are displayed in the Densify Analysis Console from the Data Collection > Discovered System Explorer (DSE) > System Summary or Attributes pages.

Resource

```
/systems
```

Supported Operations

Table: Systems Supported Operations

Operation	HTTP Method	Input	Output	Description
Get Collection	GET /systems	Query String Parameter Options: 🔖 Collection Details 🔖 Paging 🔖 Element Filters 🔖 Attributes On/Off	Collection of Response elements	Used to return configuration and attribute information of systems (of supported type) that are being tracked under recent data collection activity. Example: Getting a Collection of Sorted GCP Systems, Displaying One Page

Operation	HTTP Method	Input	Output	Description
		◾ Mode Attributes On/Off Mode ◾ Attribute Filters - Multiple Values Support ◾ Sorting		
Get Individual Systems	GET /systems/<id>	Path Parameter: ◾ id (system ID)	Response elements	Used to return configuration and attribute information for a system specified by id. The system returned is being tracked under recent data collection activity and must be one of the supported system type. Systems Systems
Get System's Instance Type Details	GET /systems/<id>/analysis-details	Path Parameter: ◾ id (system ID) Request Body Parameters: ◾ name ◾ value ◾ value	Response JSON structure for each supplied system ID.	Used to provide instance type details of the specified system as well as details of the optimal instance for this workload. Additional targets are listed based on the optional spend and effort tolerance parameters.
Get Report PDF	GET /systems/<id>/analysis-report	Path Parameter: ◾ id (system ID)		Used to obtain the Impact Analysis and Recommendation Report (also known as the Application Owner report). See *Viewing the Impact Analysis and Recommendation Report* (Help Topic ID 380450) for details on the content of the report. Use `Accept : application/octet-stream` in the request header to download the PDF file. Example: Downloading an Impact Analysis and Recommendation Report
Modify System Attributes	PUT /systems/<id>/attributes	Path Parameter: ◾ id (system ID) Request Body	Response elements with attributes modified	Used to modify the values of a system's attributes. If attribute name corresponds to an existing set attribute in the system, then the attribute's

Operation	HTTP Method	Input	Output	Description
		Parameters: array of attributes ▪ name ▪ value		value will be updated to `value`. If attribute `name` is not set in the system, then the `[name,value]` tuple will be added to the system's attribute array. For multi-value attributes (i.e. "Multiple Values" property is enabled for the attribute), a PUT request for that attribute will append the new `[name,value]` pair to the attribute array, if that attribute `[name,value]` pair does not exist. To overwrite a multi-value attribute value, you will first need to delete the existing attribute `value`. Example: Modifying a System's Attributes
Delete System Attributes	`DELETE` `/systems/<id>/attributes`	Path Parameter: ▪ id (system ID) Request Body Parameters: array of attributes ▪ id ▪ value or ▪ id (attribute ID)	Response elements with attributes deleted	Used to delete attributes from an individual system given the attribute ID-value `[id,value]` pair or attribute `[id]`. If the attribute `[id,value]` pair is provided, the attribute will be deleted only if the attribute `[id]` and corresponding `[value]` matched the one found in the system. For multi-value attributes (i.e. "Multiple Values" property is enabled for the attribute), a DELETE request with only attribute `[id]` provided will delete all the values with that attribute ID; a DELETE request with `[id,value]` pair provided will delete only the attribute ID entry with that specific value. Example: Deleting a System's Attributes

Parameters

Path Parameters

Table: Systems Path Parameters

Parameter Name	Type	Description
id	string	The Densify unique referenced ID of the system.

Request Body Parameters

Table: Systems Request Body Parameters (Array of `[id, name, value]` attributes)

Parameter Name	Type	Description
id	string	The unique ID for the attribute in Densify. System attributes are properties set by a data collection from a vendor platform or by Densify for control analytics. The system's `attributes` array contains attribute elements, stored as an `[id, name, value]` triple in the system's `attributes` array.
name	string	The `name` of the attribute corresponding to the `id` and `value` attribute tuple.
value	string	The `value` of the attribute corresponding to the `id` and `name` attribute tuple.

Query String Parameters

The following query string parameter options are available to the `/systems` resource:

- Sorting
- Attributes On/Off Mode
- Attribute Display Categories
- Element Filters
- Attribute Filters - Multiple Values Support

Sorting

You can order the returned collection by name or by memory size using the `sort_by` option. See Sort By for an overview of the Sort By option.

Supported Sort By elements for the /systems resource are described below:

- **size**

  The `size` element is a representation of the following properties to sort by, in priority order:

- `memory`
- `total_physical_cpus`
- `cores_per_cpu`

This implies that if two systems have the same total `memory`, the `total_physical_cpus` count is considered next in the sort logic; and if the physical CPU count is the same, the next property to consider for sorting is the `cores_per_cpu` count. By default, the returned `sort_by=size` collection is ordered in ascending (`asc`) order. Specify `desc` for descending order. For example:

```
GET /systems/?sort_by=size,desc
```

Systems with any "_Unknown_" values are sorted at the end of the collection independent of the specified sort order.

- **name**

  The returned collection is sorted by the system's `name` element in ascending alphabetical order. Specify `desc` for descending alphabetical order.

  If you specify the [Paging](#) option, the returned collection will be sorted by `name` in ascending order automatically. For example, the following two requests will return the same result (i.e. page two of the collection in ascending name order):

  ```
  GET /systems/?page=1
  ```

  ```
  GET /systems/?sort_by=name&page=1
  ```

Attributes On/Off Mode

By default, all the attributes with set values for each system in the collection are returned (which is the `attributes_mode=On` option). You can suppress the attributes of the returned collection by using the `attributes_mode=Off` option. For instance, if you want to return a collection of AWS systems without displaying their attributes, specify the following request:

```
GET /systems/?platform=aws&attributes_mode=Off
```

When you filter by attribute in your request, the `attributes_mode=Off` is disabled and all the set attributes are returned regardless if you provide the `attribute_mode=Off` option. For instance, if you provide the following request, the set attributes for the returned collection are displayed:

```
GET /systems/?attribute_name_value=Department,Finance&attributes_mode=Off
```

Attribute Display Categories

By default, all the attributes with set values for each system in the collection are returned. You can choose to return only specific attribute categories of the requested collection by using the `setdisplay_category` option. For instance, if you want to return a collection of AWS systems, but only display Business and Transformation attributes with set values, specify the following request:

```
GET /systems/?platform=aws&setdisplay_category=Business,Transformation
```

Element Filters

The /systems endpoint supports the following elements as filters for returning a subset of the collection:

- [name](#)
- [name_like](#)
- [resource_id](#)
- [attributes](#) **
- [type](#)
- [platform](#) *
- [infrastructure_group](#) *
- [control_environment](#) *
- [platform_category](#) *

These filters are also denoted by "F" in the Filter column of the Systems [Response](#) schema table. Refer to [Filters](#) for a complete description of the Filters feature.

Element filters denoted with ***** support multiple values input. You can provide a list of values, separated by a comma between values, to logically OR a list of possible values for an element. For example, the following request will return all systems from AWS, Azure, or GCP:

```
GET /systems/?platform=aws,azure,gcp
```

Refer to [Filters](#) for a description of the Multiple Values filtering.

System attributes filtering extends the standard multiple values filtering support (denoted by ******) with multiple attribute name, ID, and value combinations. See [Attribute Filters - Multiple Values Support](#) for further details.

> **Note:** When filtering with values which include spaces or "+", use URL encoding . See [Filters](#) for details on special character support.

Attribute Filters - Multiple Values Support

Systems attributes supports the following filtering features:

- A single value for `attribute_name_like` element filter—A collection of systems is returned with attribute names containing the sub-string provided. You can use the '%' character to match zero or more characters. For example, the request below returns systems with attribute values set for attribute names containing the sub-string "east":

```
GET /systems/?attribute_name_like=east
```

- Multiple attribute IDs for element filter `attribute_id`—The list of provided attribute IDs are matched to the system's set attributes and results are logically OR'ed to produce the returned collection. For example, the request below returns systems with set values for attribute IDs *attr_5*, *attr_6*, or *attr_7*:

```
GET /systems/?attribute_id=attr_5,attr_6,attr_7
```

- Multiple attribute names for element filter `attribute_name`—The list of provided attribute names are matched to the system's set attributes and results are logically OR'ed to produce the returned collection. For example, the request below returns systems with set values for attribute names *Department* or *Resource Group*:

```
GET /systems/?attribute_name=Department,Resource+Group
```

■ Multiple attribute ID-value criteria for the element filter `attribute_id_value`—Existing systems which match all of the provided attribute ID-value pairs (separated by semicolon) are returned. The format of this attribute ID-value criteria filter request is as follows:

```
GET /systems/?attribute_id_value=<attr1_id,attr1_value;attr2_id,attr2_
value;attr3_id,attr3_value,...>
```

The results of matching attribute ID-value pairs are logically AND'ed to produce the returned collection. For example, the systems returned for the request below have

- attr_Application = "General" AND

- CLD.BUSINESS.UNIT = "Support" AND

- attr_OperationalEnvironment = "Production"

```
GET /systems/?attribute_id_value=attr_
Application,General;CLD.BUSINESS.UNIT,Support;attr_
OperationalEnvironment,Production
```

■ Multiple attribute name-value criteria for the element filter `attribute_name_value`—Existing systems which match all of the provided attribute name-value pairs (separated by semicolon) are returned. The format of this attribute name-value criteria filter request is below:

```
GET /systems/?attribute_name_value=<attr1_name,attr1_value;attr2_name,attr2_
value;attr3_name,attr3_value,...>
```

The results of matching attribute name-value pairs are logically AND'ed to produce the returned collection. For example, the systems returned for the request below have "debian-8-jessie" as their License model *AND* "Disk-AB21" as their disk name:

```
GET /systems/?attribute_name_value=License+model,debian-8-
jessie;Disk+Name,Disk-AB21
```

> **Note:** *When you use attributes for filtering a collection, the attributes_mode=Off feature is disabled. See* Attributes On/Off Mode *for details.*

Response

The following is a complete list of possible response elements that are returned for the `/systems` resource. If the response element does not apply to the system returned, then the element is not displayed in the results.

Table: Systems Response Schema

Element	Type	Mod	Filter/Sort	Description
id, name, href	strings		F by ■ `id` ■ `name` ■ `name_like` S by `name`	See ID, Name and Self Reference (id, name, href). To filter systems with names that contain a given input string, use `?name_like="<substring>"` in your collection request. The '%' character can be used to match zero or more characters. Example: Getting a Collection of Azure Systems with Name "*test*"
resource_id	strings		F	The unique system ID assigned by the cloud

Element	Type	Mod	Filter/Sort	Description
				provider.
				To search for a system with a particular cloud provider ID, use `?resource_id=<cloud_id>` in your request.
				[Example: Getting Public Cloud Attributes for an Individual System](#)
attributes	[id, name, value]	M	F by ■ `attribute_name_like` ■ `attribute_id` ■ `attribute_name` ■ `attribute_id_value` ■ `attribute_name_value`	System attributes are properties set by an audit from a vendor platform or by Densify for control analytics.
				On a `GET` request, only those attributes that have values are returned. To hide the set attributes, use the `attributes_mode=Off` option. See [Attributes On/Off Mode](#) for details. To only return set attributes in specific categories, use the `setdisplay_category` option. See [Attribute Display Categories](#) for details.
				To filter based on attributes, you must use one of the following element options – in each case, system data will be returned only if the attributes selected have values assigned to them: ■ `attribute_name_like` ■ `attribute_id` with a provided list of attribute IDs ■ `attribute_name` with a provided list of attribute names ■ `attribute_id_value` with a provided list of attribute ID-value pairs ■ `attribute_name_value` with a provided list of attribute name-value pairs
				Refer to [Attribute Filters - Multiple Values Support](#) for a description of how to use these attribute filters.
				Note that when you use attributes to filter the collection, the `attributes_mode=Off` feature is disabled.
				[Systems](#)
				[Example: Getting Public Cloud Attributes for an Individual System](#)
children	string			The number of other systems that are considered "children" of the current system.
				For a host system (i.e. `"type": "host"`), this is the number of VMs the host system has (e.g. `"children": "10"`). For AWS ASGs (i.e. `"type" : "asg"`), this is the number of active EC2 instances that belong to the group. For other type of systems where this element does not apply, `children` is not returned.
				This element is returned for host and ASG systems only when the `details=true` option is

Element	Type	Mod	Filter/Sort	Description
				in the request URI. See Collection Details.
control_ environment	[id, name, platform_ category, href, icon]		F by ▪ `control_ environment` ▪ `platform_ category`	If the system is associated with a Control Environment, the environment details are returned: ▪ `id` ▪ `name` ▪ `platform_category` ▪ `href` ▪ `icon` To filter based on Control Environment, you must use one of the following elements: ▪ `control_environment`—with a control environment name specified ▪ `platform_category`—with a platform category specified (i.e. "External Cloud" or "Internal Virtual") Example: Getting a Filtered Collection of Systems.
cores_per_cpu	string		Tertiary sort key for `sort_by=size`	Cores per CPU for computing systems.
cpu_ benchmarks	[name, score_ type, value]			The default CPU benchmark for applicable systems. ▪ `name`—Label with possible values: • `"CINT2000"` • `"CINT2000 Rate"` • `"CINT2006 Rate"` • `"RPE2"` ▪ `score_type`—Score type with possible values: • `"cint2000"` • `"cint2000rate"` • `"cint2006rate"` • `"rpe2"` ▪ `value`
cpu_model	string			The CPU architecture model of the compute system.
cpu_speed	string			The normalized CPU speed (MHz) of the compute system.
entity_role_ name	string			The entity role name that Densify assigns to the system (e.g. "VMWARE_VM") based on their platform and system role.
entity_type	string			The entity type that Densify assigns to the system (e.g. "VMware ESX Guest") based on their platform and system type.
hostId	string			An ID of the system provided from the data collection audit for the purpose of resolving duplication conflicts.
I/O_ benchmarks	[name, score_ type, value]			The list of all available I/O benchmarks. ▪ `name`—Label with possible values: • `"Maximum Disk Throughput (bytes/s)"` • `"Maximum Network Throughput (bytes/s)"`

Element	Type	Mod	Filter/Sort	Description
				▪ `score_type`—Score type with possible values: • `"disk"` • `"net"` ▪ `value`—A value of -1 means there is no value specified.
infrastructure_ group	[id, name, href]		F by `infrastructure_ group`	If the system is associated with an Infrastructure Group (cluster), the group details are returned: ▪ `id` ▪ `name` ▪ `href` When filtering based on Infrastructure Group, you must use the element `infrastructure_group` with the group name specified. Example: Getting a Filtered Collection of Systems.
ip_address	string			Primary IP address of the system.
mac_address	string			MAC address of the system provided by the data collection audit.
manufacturer	string			For host type systems, the manufacturer of the host is returned (e.g. "Dell", "IBM"). For cloud systems, the vendor platform is returned (e.g. "GCP") or "CONTAINERS" is returned for container type systems. This element is returned only when the `details=true` option is in the request URI. See Collection Details.
memory	string		S by `size`	The normalized total memory (MB) for the system.
os	string			Operating System name of the host or VM instance. This element is returned only when the `details=true` option is in the request URI. See Collection Details
os_patch_level	string			Operating System patch level of the host or VM instance.
os_version	string			Operating System version of the host or VM instance.
parent	string			Logical parent of the current system. For a host system (i.e. `"type": "HOST"`), this does not apply (i.e. `"parent": "N/A"`). For a VM, this is the name of the parent host system (e.g. `"parent": "esx-host-221"`).
platform	string		F	The platform of the system. Use this element to filter systems from the various supported platforms: ▪ `VMWARE`—For systems in VMware ▪ `HMC`—For systems in IBM Power ▪ `AWS`—For systems in AWS ▪ `GCP`—For systems in GCP ▪ `AZURE`—For systems in Azure ▪ `CONTAINERS`—For containers, both

Element	Type	Mod	Filter/Sort	Description
				Kubernetes and ECS
				For example, use `platform=azure` to return a collection of Azure cloud systems. See Example: Getting a Collection of Azure Systems with Name "*test*".
platform_model	string			For host systems, the host system model is returned. For cloud instance systems, the instance type is returned.
serial_number	string			The serial number of the system from the vendor.
size	hidden element		S by `size`	This element is used to sort a collection by size. See Sorting for details.
total_logical_cpus	string			The total number of logical CPUs for the system.
total_physical_cpus	string		Secondary sort key for `sort_by=size`	The total number of physical CPUs for the system.
threads_per_core	string			The threads per core for the system.
type	string		F	The type of system. The supported types of systems for the `/systems` resource include: • `host`—For host systems in VMware, IBM PowerVM (via data collection for HMC 8 method), and AWS platforms. • `guest`—For guest systems in a VMware platform. • `vm`—For instances in VMware, IBM PowerVM (via data collection for HMC 8 method), GCP (Compute Engine), and AWS (EC2, ECS) platforms. • `arm_vm`—For instances in the Azure platform. • `classic_vm`—For classic instances in the Azure platform. • `rds`—For RDS instances in the AWS platform. • `asg`—For Auto Scaling groups in the AWS platform. • `ecs_svc`—For ECS services in the AWS platform. • Containers realized through data collection are supported by the `/systems` endpoint, but does not have the type element populated. You can filter Container systems via the `platform=containers` element filter.

Note: *IBM Power systems returned by the /systems endpoint are the ones created in Densify from the HMC 8 data collection method. Refer to Data Collection for IBM PowerVM Systems for HMC 8 (Help Topic ID 220170) for details.*

Examples

Example: Getting a Collection of Azure Systems with Name "*test*"

The following example shows you how to retrieve a collection of systems from the Azure cloud platform with a name containing "test".

Example: Getting a Collection of Azure Systems with Name "*test*"

Request:

```
GET /systems/?platform=azure&name_like=test
```

Response:

```
[
  {
    "id": "5da2be9c-3915-46df-a127-387e837a0697",
    "name": "esx-test-274",
    "href": "/systems/5da2be9c-3915-46df-a127-387e837a0697",
    "resource_id": "esx-test-274",
    "type": "classic_vm",
    "platform_model": "standard_a0",
    "platform": "AZURE",
    "total_physical_cpus": "1",
    "cores_per_cpu": "1",
    "memory": "768",
    "infrastructure_group": {
      "id": "c55e12ae-d568-427b-997d-1d5c54065a2d",
      "name": "eastus-test-vm",
      "href": "/infrastructure-groups/c55e12ae-d568-427b-997d-1d5c54065a2d"
    },
    "control_environment": {
      "id": "da70ab94-cea1-4a8a-83d7-c26a675ce650",
      "name": "cc377154-9605-4cb0-8b41-1b39e1c4ac0f",
      "platform_category": "External Cloud",
      "href": "/control-environments/da70ab94-cea1-4a8a-83d7-c26a675ce650",
      "icon": "/control-environments/da70ab94-cea1-4a8a-83d7-c26a675ce650/icon"
    },
    "attributes": [
      {
        "id": "attr_azure_resource_group",
        "name": "Resource Group",
        "value": "test-vm"
      },
      {
        "id": "attr_azure_tenant_id",
        "name": "Tenant ID",
        "value": "6c9190a7-bca6-4fcd-b35e-36378aadc695"
      },
      // ... *SNIP* ... additional attributes not displayed ... *SNIP* ...
    ]
  },
  {
```

```
    "id": "0e580bbb-9c21-49ff-b3df-729e80e26558",
    "name": "TestVMTestSub",
    "href": "/systems/0e580bbb-9c21-49ff-b3df-729e80e26558",
    "resource_id": "TestVMTestSub",
    "type": "arm_vm",
    "platform_model": "basic_a0",
    "platform": "AZURE",
    "total_physical_cpus": "1",
    "cores_per_cpu": "1",
    "memory": "768",
    "infrastructure_group": {
      "id": "30030e26-f5f3-431c-9d21-79e499ec763b",
      "name": "eastus+testresourcegroup",
      "href": "/infrastructure-groups/30030e26-f5f3-431c-9d21-79e499ec763b"
    },
    "control_environment": {
      // ... *SNIP* ... control_environment elements ... *SNIP* ...
    },
    "attributes": [
      {
        "id": "attr_2",
        "name": "Department",
        "value": "IT"
      },
      {
        "id": "attr_azure_resource_group",
        "name": "Resource Group",
        "value": "testresourcegroup"
      },
      // ... *SNIP* ... additional attributes not displayed ... *SNIP* ...
    ]
  },
    // ... *SNIP* ... additional Azure systems not displayed ... *SNIP* ...
]
```

Example: Getting a Collection of Sorted GCP Systems, Displaying One Page

This example shows you how to retrieve a collection of GCP systems, sorted by memory size in ascending order and only displaying the second page (note that `page=0` is the first page).

Example: Getting a Collection of Sorted GCP Systems and Displaying One Page

Request:

```
GET /systems/?platform=gcp&sort_by=size&page=1&page_size=1
```

Example: Getting a Filtered Collection of Systems

This example shows you how to retrieve a collection of systems filtered by platform and cluster.

Example: Getting a Filtered Collection of Systems

Request:

```
GET /systems/?platform=aws&infrastucture_group=us-east-1b&platform_
category=External+Cloud
```

Response:

```
[
  {
    "id": "007c3e76-9d76-45ec-b8b6-16fcc493121e",
    "name": "007c3e76-9d76-45ec-b8b6-16fcc493121e",
    "href": "/systems/007c3e76-9d76-45ec-b8b6-16fcc493121e",
    "resource_id": "i-036ddcaa03393444b",
    "type": "vm",
    "platform_model": "c4.2xlarge",
    "platform": "AWS",
    "total_physical_cpus": "8",
    "cores_per_cpu": "1",
    "memory": "16512",
    "infrastructure_group": {
      "id": "7393175f-cb71-4a1c-9f17-55a94a98f752",
      "name": "us-east-1b",
      "href": "/infrastructure-groups/7393175f-cb71-4a1c-9f17-55a94a98f752"
    },
    "control_environment": {
      "id": "d96431b4-99f3-43c0-98d6-609a4b80f0dd",
      "name": "us-east-manual",
      "platform_category": "External Cloud",
      "href": "/control-environments/d96431b4-99f3-43c0-98d6-609a4b80f0dd",
      "icon": "/control-environments/d96431b4-99f3-43c0-98d6-609a4b80f0dd/icon"
    },
    "attributes": [
      {
        "id": "attr_BasePerformanceCINT2006Rate",
        "name": "Base Performance CINT2006Rate",
        "value": "0.021513293"
      },
      // ... *SNIP* ... additional attributes not displayed ... *SNIP* ...
    ]
  },
    // ... *SNIP* ... additional filtered AWS systems not displayed ...
*SNIP* ...
]
```

Example: Downloading an Impact Analysis and Recommendation Report

The following example shows you how to download a PDF version of the Impact Analysis and Recommendation Report for your public cloud (AWS, Azure, GCP) or private cloud (VMware, IBM PowerVM) instances. The report is available after the Densify analysis generates the right-sizing recommendations and rdb-populate has run to completion.

Note: *HTTPS needs to be enabled to download the Impact Analysis and Recommendation Report PDF.*

Example: Download an Impact Analysis and Recommendation Report

Request:

```
GET /systems/7836335a-1942-4115-a65d-a298be1d390c/analysis-report
```

Headers:

```
Accept: application/octet-stream
Authorization: Bearer <apiToken>
```

You need to save the file (Save Response) and then open it in a PDF reader.

Figure:Postman Header Configuration

You must disable the default **Accept** key value "*/" for **Headers**.

Example: Modifying a System's Attributes

The following example shows you how to modify system attributes.

Example: Modifying a System's Attributes

Request:

```
PUT /systems/0060ed45-6f83-4d74-9ff6-7a359e206428/attributes
[
  {"name": "Observed Uptime", "value": "0.33"},
  {"name": "Cost", "value": "17"}
]
```

Response:

```
[
  // ... *SNIP* of other elements ...
    "attributes": [
    // ... *SNIP* of other attributes ...
```

```
        {
          "id": "attr_ObservedUptime",
          "name": "Observed Uptime",
          "value": "0.33"
        },
        {
          "id": "attr_Cost",
          "name": "Cost",
          "value": "17"
        },
        // ... *SNIP* of other attributes ...
      ]
   // ... *SNIP* of other elements ...
```

Example: Deleting a System's Attributes

The following example shows you how to delete a system's attr_3 attribute if it has value "0.3", and remove set values of attr_10 and attr_11.

Example: Deleting a System's Attributes

Request:

```
DELETE /systems/0060ed45-6f83-4d74-9ff6-7a359e206428/attributes
[
  {"id": "attr_3", "value": "0.3"},
  {"id": "attr_10"},
  {"id": "attr_11"}
]
```

Example: Getting Public Cloud Attributes for an Individual System

The following example shows you how request information for a system based on the cloud provider's system ID (resource_id) and only show Public Cloud category attributes with set values.

Example: Getting Public Cloud Attributes for an Individual System

Request:

```
GET /systems/?resource_id=i-00bc69701199bd58&setdisplay_category=Public+Cloud
```

Systems: Analysis Details

Description

The `/systems/<id>/analysis-details/` resource is used to return instance type details of the specified system as well as details of the optimal target instance type. Additional targets are listed based on the optional spend and effort tolerance parameters.

See for details of attribute management using the `/systems` endpoint.

Resource

```
/systems/<id>/analysis-details?target=<target value>
```

Supported Operations

Table: Supported Operation

Operation	HTTP Method	Input	Output	Description
Get a system's instance type details	GET `/systems/<id>/analysis-details`	Path Parameters: 🔖 id (system ID) 🔖 target 🔖 spend tolerance 🔖 effort tolerance	Response for each supplied system ID.	Provides instance type details of the specified system as well as details of the recommended, optimal instance for this workload. Additional targets are listed based on the mandatory target parameter and the optional spend and effort tolerance parameters.

Parameters

Path Parameters

Table: Path Parameter

Parameter Name	Type	Description
id	string	The unique entity ID of the instance for which you want to review recommendation details.
target	string	Target is mandatory and you can optionally specify one or both of spend and effort tolerance. Specify details of the target instance type. Details are returned in the last section of the response. Possible values are: - "all_instances"—The optimal instance type is returned as well as the targets array containing all target instance types. Depending on the cloud provider, this can be 700+ instance types. - "compatible_instances"—The returned array contains only, instance types that have compatibility of "OK". - "incompatible_instances"—The returned array contains instance types that all have compatibility NOT equal to "OK"; - "optimal_instance"—Returns the single optimal instance. The empty targets array is not returned. - Specific instance type name—Specify an instance type name, if different from the optimal instance type. i.e. m6i.large. The returned array contains details of the single, specified instance type. You do not need to specify the manufacturer's exact instance type name. Densify will map the entry to the provider's cloud catalog.
spend tolerance	string	Specify a decimal, limit value that when exceeded makes the instance incompatible. i.e. a value of "1.2" means that target instance types that cost 20% more than the current instance type cost are considered compatible. Specifying a spend tolerance affects the compatibility value of instance types, that may have been considered compatible, based only on resource allocation. i.e. if the instance is compatible but outside spend tolerance, the compatibility value changes from "OK" to "Outside Spend Tolerance". Specifying the spend tolerance does not reduce the number of instance types returned in the targets array. The spend tolerance value impacts the compatibility value of the instance types listed in the in the targets array.
effort tolerance	string	Specify the effort level, above which an instance type is considered incompatible. i.e. a value of "moderate" means that target instance types with effort level of low or very low will be considered compatible. Specifying an effort tolerance affects the compatibility value of

Parameter Name	Type	Description
		instance types that may have been considered compatible, based only on resource allocation. i.e. if the instance type is compatible but exceeds the effort tolerance, the "OK" becomes "Outside Effort Tolerance". Possible values are: • "very_low"–Only instances with very low effort or no effort ("none") are considered compatible." • "low"–Only instances with low, very low or no effort required are considered compatible. • "moderate"–Only instances with moderate, low, very low effort" or no effort required are considered compatible. • "high"–Only instances with high, medium, moderate, low or very low effort are considered compatible. All instances other than those that are impossible are considered compatible. As indicated above, specifying an effort tolerance does not reduce the number of instance types returned in the targets array.

The analysis uses the following priority to exclude target instance types from the list of compatible options.

1. Insufficient Resources (Highest priority)

2. Technically Incompatible

3. Outside Spend Tolerance

4. Outside Effort Tolerance

5. OK

For example, if you have a target instance type that exceeds both spend tolerance and the effort tolerance, Densify returns "Outside Spend Tolerance" for the compatibility value since it has a higher priority.

Also, as indicated above, specifying an effort and /or spend tolerance affects the compatibility value of instance types that may have been considered compatible, based only on resource allocation. i.e. if the instance is compatible but exceeds the effort or spend tolerance, the "OK" becomes Outside Effort | Spend Tolerance. See the second example below.

Response

The response consists of a JSON structure with 3 sections for each specified entity ID.

• Current instance type details;

• Optimal instance type details;

• Nested array with details of the additional targets based on query inputs

Table: Response–Current Instance Type Details

Element	Type	Filter/Sort	Description
entityID	string		The unique instance identifier.

Element	Type	Filter/Sort	Description
			When a new analysis is requested from the `/analyze` resource, the entity ID will not be available until after data collection completes and the analysis entity is created.
displayName	string		The cloud provider account name (e.g. AWS account name).
resourceID	string		The cloud provider-assigned identifier for this instance.
azureId	string		The unique identifier for the analysis entity in Densify. Provided for Azure instances only.
resourceGroup	string		The resource group, to which this instance belongs. Provided for Azure instances only.
currentInstanceType	string		The current instance type.
blendedScore	string		The blended score that is used to determine compatibility.
compatibility	string		The compatibility value.

Table: Response–Optimal Instance Type Details

Element	Type	Filter/Sort	Description
instanceType	string		The optimal instance type as determined by the Densify analysis. An instance type value is not returned for terminate recommendations.
blendedScore	string		The blended score that is used to determine compatibility. The blended score is set to -1 for terminate recommendations.
compatibility	string		The value that determines whether or not the instance type is compatible to move the workload from the current instance type. Possible values are: • Insufficient Resources–Blended Score = 0; • Technically Incompatible–Blended Score < 50; • OK–Blended Score ≥ 50 • Outside Spend Tolerance–Blended Score ≥ 50 but cost exceeds the defined spend tolerance. • Outside Effort Tolerance–Blended Score ≥ 50 but effort exceeds the defined effort tolerance. A compatibility value is not returned for terminate recommendations.
recommendationType	string		The recommendation generated by the Densify analysis.

Table: Response–Additional Instance Details

Element	Type	Output Section	Description
instanceType	string		Other compatible instance type as determined by the Densify analysis.

Element	Type	Output Section	Description
cpuModel	string	All	The CPU architecture for this instance type. For terminate this value will be set to N/A, in the optimal section.
numCpus	integer	All	The number of CPUs for this instance type. For terminate this value will be set to -1, in the optimal section.
memory	decimal/float	All	The total_memory, in GB for this instance type. For terminate this value will be set to -1, in the optimal section.
generation	integer	All	The instance's generation. For terminate this value will be set to -1, in the optimal section.
catalogCost	decimal/float	All	The cost of the instance type without considering the uptime. For terminate this value will be set to -1, in the optimal section.
blendedScore	string		The blended score that is used to determine compatibility.
compatibility	string		The compatibility value.
predictedUptime	decimal/float	Current only	The predicted uptime (%) for the system is based on the percentage of hours CPU utilization data is present in the workload range specified in the policy settings. Predicted uptime % for new systems started mid-way within the workload range is calculated from the time/date that the system was started, as opposed to the beginning of the interval resulting, in more accurate prediction for the future.
percentOptimalCost	decimal/float	Current and Targets	The cost of the current instance type divided by the optimal cost or the cost of the target instance type divided by the optimal cost
effortEstimate	string	Optimal and Targets	The effort required to make the change.

Table: Response–Status Details

Element	Type	Filter/Sort	Description
message	string		The informational message returned with the status response.
status	number		The HTTP response code of the request. Possible status values include: ▪ 200–OK; ▪ 200–Entity Not Analyzed–The specified entity type is not supported for the analysis used to generate the catalog map. i.e. AWS ASGs and GCP instance types are not currently supported. ▪ 200–Instance Type Analysis is Disabled–Publishing of the required data has been disabled. Contact Support@Densify.com to enable this feature. ▪ 400–Invalid parameter; ▪ 401–Unauthorized; ▪ 404–Entity not found;

Element	Type	Filter/Sort	Description
			500–Internal server error.

Examples

Example: Return Compatible Targets within Spend Tolerance

The following example shows you how to obtain a list of only compatible targets within the spend tolerance. The cost must be within 30% of the current instance type:

Example: Listing Compatible Targets within Spend Tolerance

Request:

```
/systems/076e70cf-cff2-3cc6-8a4c-54200241de7c/analysis-
details?target=compatible_instances&spendTolerance=1.3
```

Response:

```
[
    {
        "current": {
            "entityId": "076e70cf-cff2-3cc6-8a4c-54200241de7c",
            "displayName": "testsystem",
            "resourceId": "9a0546d2-f4e1-46f0-bc09-a7422a6b72927",
            "azureId": "/subscriptions/cc377154-9605-4cb0-8b41-1b39e1c/re-
source-
groups/testsystem/providers/microsoft.compute/virtualmachines/testsystem",
            "resourceGroup": "testsystem",
            "instanceType": "Standard_DS1_v2",
            "blendedScore": 90,
            "compatibility": "Outside Spend Tolerance"
            "cpuModel": "Intel Xeon E5-2673 v3",
            "numCpus": 1,
            "memory": 3.5,
            "generation": 2,
            "catalogCost": 53.29,
            "predictedUptime": 84.51,
            "percentOptimalCost": 7.0210805
        },
        "optimal": {
            "instanceType": "Standard_B1s",
            "blendedScore": 99,
            "compatibility": "OK"
            "recommendationType": "Downsize - Optimal Family"
            "cpuModel": "Intel Haswell E5-2673 v3",
            "numCpus": 1,
            "memory": 1.0,
```

```
            "generation": 4,
            "catalogCost": 7.59,
            "effortEstimate": "Low",
    },
        "targets": [
            {
                "instanceType": "Standard_B1s",
                "cpuModel": "Intel Haswell E5-2673 v3",
                "numCpus": 1,
                "memory": 1.0,
                "generation": 4,
                "catalogCost": 7.59,
                "blendedScore": 99,
                "compatibility": "OK"
                "percentOptimalCost": 1.0
                "effortEstimate": "Low",
            },

            {...},
            {...},
            {...},
        ]
    }
```

Example: Return Compatible Targets within Spend and Effort Tolerance

The following example shows you how to obtain a list of all compatible targets within the spend tolerance. The cost must be within 50% of the current instance type. The effort must be equal to or lower than the value specified in the request.

Example: Listing All Instance and Setting Both Spend and Effort Tolerances

Request:

```
systems/123456-1234-12315245-1231/analysis-details?target=all_
instances&spendTolerance=1.5&effortTolerance=low
```

Response:

```
[
    {
        "current": {
            "entityId": "123451231231-123-123-123",
            "displayName": "testsystem",
            "resourceId": "456789-123-5657",
            "instanceType": "c5.2xlarge",
            "blendedScore": 95,
            "compatibility": "OK"
            "cpuModel": "Intel Xeon",
            "numCpus": 8,
```

```
        "memory": 16.0625,
        "generation": 7,
        "catalogCost": 11.43,
        "blendedScore": 98,
        "compatibility": "Outside Spend Tolerance"
        "percentOptimalCost": 1.0
        "effortEstimate": "Low",
    },
    "optimal": {
        "instanceType": "r5.xlarge",
        "blendedScore": 99,
        "catalogCost": 12.00,
        "compatibility": "OK"
        "recommendationType": "Downsize - Optimal Family"
        "effortEstimate": "Very Low"
    },
    "targets": [
        {
            "instanceType": "m5.8xlarge",
            "blendedScore": 87,
            "catalogCost": 80.00,
            "compatibility": "Outside Spend Tolerance"
            "effortEstimate": "Low"
        },
        {
            "instanceType": "r5.xlarge",
            "blendedScore": 99,
            "catalogCost": 12.00,
            "compatibility": "OK"
            "effortEstimate": "Very Low"
        },
// The following instance type is within the effort tolerance, but outside the
spend tolerance.
        {
            "instanceType": "r5.8xlarge",
            "blendedScore": 88,
            "catalogCost": 96.00,
            "compatibility": "Outside Spend Tolerance"
            "effortEstimate": "Very Low"
        },
//The following instance type is outside both spend tolerance and effort tol-
erance.
        {
            "instanceType": "c5.2xlarge",
            "blendedScore": 95,
            "catalogCost": 35.00,
            "compatibility": "Outside Spend Tolerance"
            "effortEstimate": "High"
        },
//The following instance type is within the spend tolerance but outside the
effort tolerance.
        {
            "instanceType": "m6g.2xlarge",
            "blendedScore": 77,
            "catalogCost": 14.00,
            "compatibility": "Outside Effort Tolerance"
```

API Reference Guide for Public Cloud

```
            "effortEstimate": "Moderate"
        },
        {...},
        {...},
        {...},
    ]
}
```

Subscriptions

Description

The `/subscriptions/<platformType>` resource is used to create and manage subscription-based notifications for Densify cloud and container recommendations.

To learn more watch the video, Subscription API Overview

This resource allows you to set up and schedule the delivery of personalized Densify recommendations to a third-party application for targeted distribution.

Personalization of the recommendation data set (returned subscription notification output) is facilitated by the following mechanisms:

- Return only the systems you are interested in by specifying system attribute conditions with the tagReferences parameter.

- Return only specific recommendations you are interested in by specifying recommendation property conditions with the propertyReferences parameter.

- Suppress certain systems or recommendations in the returned output, for a period of time, with the suppressionReferences parameter.

- Finally, the fields and attributes displayed in the returned output can be customized with the returnStructure parameter. You can specify which recommendation field to display and whether to display the field name or the field alias as the element key.

Subscription notifications are delivered to a third party application, such as Microsoft Teams™ or Slack™, using the webhook definition. If you do not specify a webhook in your Subscriptions definition, the notifications are not triggered because there is no delivery location. See *Subscriptions: Status* (Help Topic ID 340740) for details on Subscriptions results and webhook status.

You can dynamically retrieve the current personalized Subscriptions recommendations by providing a subscription ID in the GET subscription request. See *Subscriptions: Cloud Results* (Help Topic ID 340750) for details on retrieving Subscriptions results on-demand.

You can also schedule the frequency of the notification distribution using the schedule definition parameter. Subscriptions without a schedule definition will typically have notifications triggered nightly after recommendation analysis and reporting database updates.

Resource

```
/subscriptions/cloud
/subscriptions/containers
/subscriptions
```

Note: *If you use this resource without the `<platformType>` specified (i.e. without `cloud` or `containers` specified), the behavior is exactly the same as specifying the `cloud`-specific resource. This behavior enables backward compatibility with scripts using the Densify API prior to release 12.1.6, where the platform-specific indicator was not available.*

Supported Operations

Table: Subscriptions Supported Operations

HTTP Method	Input	Output	Description
GET `/subscriptions/ <platformType>`	Path Parameter: ■ platformType Query String Parameter: ■ type ■ owner ■ subscriptionRef	Collection of ■ subscriptionRef ■ subscriptionName ■ description ■ owner ■ outputType ■ active ■ webhook ■ propertyReferences ■ tagReferences ■ suppressionReferences ■ schedule ■ returnStructure ■ webhookStatus ■ lastTriggered	Returns a list of existing platform-specific subscriptions in Densify. ■ The type query string parameter is used to return global or private subscriptions. If type is not specified, all global and only private subscriptions belonging to you are returned. ■ If type is not specified and you are an **administrative user**[1], then all global and all private platform-specific subscriptions (belonging to all users) are returned. ■ Administrative users can use the owner

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
			query string parameter to return all the private subscriptions belonging to a specific user. ■ Use the subscriptionRef query string parameter to return details of a single subscription. Example: Getting a Collection of Subscriptions Example: Getting a Collection of Subscriptions for a Specific User
POST /subscriptions/ <*platformType*>	Path Parameter: ■ platformType Collection of Request Body Parameters: ■ subscriptionName ■ owner ■ description ■ outputType ■ active ■ webhook ■ propertyReferences ■ tagReferences ■ suppressionReferences ■ returnStructure ■ schedule	Collection of ■ subscriptionRef ■ subscriptionName	Creates and defines a collection of platform-specific subscriptions. ■ If you are not an **administrative users**[1], the owner parameter is automatically set to your Densify username. ■ The Subscriptions creation (bulk-add) operation is committed as a whole unit. One Subscriptions creation error will roll back the entire operation. For any Subscriptions creation failure, an appropriate error message will be returned in the response body. See status for possible error status. Example: Creating a New Subscription

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

API Reference Guide for Public Cloud

HTTP Method	Input	Output	Description
PUT /subscriptions/ *<platformType>*/ *<subscriptionId>*	Path Parameters: ■ platformType ■ subscriptionRef Request Body Parameters: ■ subscriptionName ■ owner ■ description ■ active ■ webhook ■ propertyReferences ■ tagReferences ■ suppressionReferences ■ returnStructure ■ schedule	■ subscriptionRef ■ subscriptionName	Deletes and replaces all the parameters in an existing platform-specific subscription definition. You must specify all parameters (from the Request Body Parameters section) required for the existing subscription. For any parameter replacement failure, an appropriate error message will be returned in the response body. See status for possible error status. Example: Modifying a Subscription *Note:* *If you are not an administrative user, you can only modify your own private subscription.*
DELETE /subscriptions/ *<platformType>*	Path Parameter: ■ platformType Collection of Request Body Parameters: ■ subscriptionRef	■ message ■ status ■ HTTP status "204 No Content" is returned for successful deletions. ■ HTTP status "404 Not Found" is returned if the subscription does not exist or you have no privilege to access the subscription.	Deletes a collection of your private platform-specific subscriptions. If you are an administrative user, use this method to delete any private or global platform-specific Subscriptions collections. Each subscription delete operation is independent from the other subscription delete operations in the same request. An error with one subscription delete action does not affect the delete actions of the other subscriptions in the same request body parameter .
DELETE	Path Parameters:	■ message ■ status	Deletes one platform-specific subscription

HTTP Method	Input	Output	Description
/subscriptions/ <platformType> /<subscriptionRef>	platformType subscriptionRef	HTTP status "204 No Content" is returned for successful deletions. HTTP status "404 Not Found" is returned if the subscription does not exist or you have no privilege to access the subscription.	definition from Densify. If you are an administrative user, use this method to delete any private or global subscription. Otherwise, if you are a non-administrative user, use this method to delete one of your private subscriptions. Example: Deleting a Subscription

Parameters

Path Parameters

Table: Subscriptions Path Parameters

Parameter Name	Type	Required (Y/N)	Description
platformType	string	Y	[cloud|containers] Specify the technology platform for the subscription resource.
subscriptionRef	string	Y	Specify the unique subscription identifier.

Query String Parameters

Table: Subscriptions Query String Parameters

Parameter Name	Type	Required (Y/N)	Description
type	string		Specify the type of subscription to return: • all–(default) Return all subscriptions: global and private user-specific. If you are not an administrative user, only private subscriptions owned by you and global subscriptions are returned. This is the default behavior if "type" is not specified in the request. • global–Return all global subscriptions. • owner–Return user-specific subscriptions. If you are not an administrative user, only private subscriptions owned by you are returned. If you are an administrative user, all global and private subscriptions are returned. Typically, this option is used in conjunction with the owner query string

Parameter Name	Type	Required (Y/N)	Description
			parameter.
			A subscription is considered *global* if the owner parameter is not populated. Global subscriptions can be used by all Densify API users.
			A subscription is considered *private* if the owner parameter contains a Densify username. Private subscriptions can only be used by their owners or administrative users.
owner	string		If you are an **administrative user**[1], you can specify a Densify username in conjunction with the `type=owner` query string parameter to return all of the specified user's private subscriptions.
			If you are not an administrative user, you can only request for your own private subscriptions. If you use the `?type=owner&owner=<anotherusername>` query string option with a username other than your own, the returned response is a 400 Bad Request - "Current login user cannot query for owner" error.
subscriptionRef	string		Specify the identifier of the subscription details to return.

Request Body Parameters

Table: Subscriptions Request Body Parameters

Parameter Name	Type	Required (C-create/M-modify/D-delete)	Description
subscriptionRef	string	D	Specify the unique subscription identifier to delete.
subscriptionName	string	C M	Specify a name for this subscription.
			For *global* subscriptions, the subscriptionName must be unique per platform. For *private* subscriptions, the subscriptionName must be unique per owner and across all global subscriptions for a particular platform. For example, owner A and owner B can both have a private cloud subscriptionName named "SubA", as long as "SubA" is not also a global cloud subscription.
owner	string	M[2]	When the `owner` parameter is not set, the subscription is considered *global*. Global subscriptions can be used by all Densify API users. Only **administrative users**[3] can

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.
[2]The `owner` parameter is mandatory for modifying private subscriptions.
[3]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Parameter Name	Type	Required (C-create/M-modify/ D-delete)	Description
			create global subscriptions. When the `owner` parameter is set, the subscription is considered *private*. Private subscriptions can only be used by their owners or administrative users. If you are an administrative user, you have the ability to assign any Densify user as the owner of the subscription in a POST request. In a PUT request, administrative users can promote the subscription from private to global by setting `owner: ""`. If you are not an administrative user, you can only set the `owner` parameter to your username. In a POST request, the `owner` parameter is automatically set to your username.
description	string		Specify a description for subscription.
outputType	string		`application/json` is the only supported type.
active	string: `[true|false]`		Specify if the subscription is active: ■ true—The subscription is active and notifications are triggered at the scheduled times to the webhook defined. ■ false (default)—The subscription is dormant: no subscription notification is triggered, even if webhook and schedule parameters are defined. Setting the subscription to dormant allows you to temporarily disable the subscription notification without modifying the subscription configuration. Dormant subscriptions can also be used as an example or template for future subscription creation. Results from dormant subscriptions can be requested on-demand. See *Subscriptions: Cloud Results* (Help Topic ID 340750).
propertyReferences	array of property conditions: ■ propertyID ■ operator ■ values	C M[1]	Specify an array of property conditions to filter the customized data set. The following items need to be specified for each property condition: ■ propertyID—specify a global property reference or a private property reference owned by you (see propertyID | tagID | suppressionID); ■ operator—specify the comparison operator (see operator); ■ values—specify the array of comparison values (see values). See Filter and Suppression Conditions for details on how to define property conditions.

[1]At least one filter array (i.e. propertyReferences, tagReferences, or SuppressionReferences) must be defined for each subscription.

Parameter Name	Type	Required (C-create/M-modify/ D-delete)	Description
			The platform of the properties defined in property conditions must match the subscription's platform. For instance, you must only use container properties for a container subscription.
			Refer to *Subscriptions: Properties > Default Properties* (Help Topic ID 340710) for the list of default properties to use in your array of property conditions.
			Note: *You must have at least one core property in your property condition set for the purpose of filtering subscription returned results. Refer to Core and Ancillary Properties in Filter Conditions for details.*
tagReferences	array of: • tagID • operator • values	C M[1]	Specify an array of attribute tag conditions to filter the customized data set. For example, if you want to return only recommendations for systems in the *ABC* account, in the *Marketing* business unit, and running the *Demo* application, then you would define `tagReferences` conditions where account="ABC", business_unit="Marketing", and app="Demo". The platform of the tags defined in tag conditions must match the subscription's platform. For instance, you must only use cloud tags for a cloud subscription. Each tag condition contains the following items: • tagID–specify a global or private tag reference owned by you (see propertyID | tagID | suppressionID); • operator–see operator; • values–see values. Refer to Filter and Suppression Conditions for details on how to define tag conditions.
suppressionReferences	array of suppression conditions: • suppressionID • operator • values • revokeBy (optional)	C M[2]	Specify an array of suppression conditions to determine systems to be excluded from the notification data set. For example, if you want to remove all "m3" systems from the results data set for this month, then you would create a suppression list with currentType="m3*" that will expire at the end of the month. The platform of the suppressions defined in suppression conditions must match the subscription's platform. For example, you must only use cloud suppressions for a cloud subscription.

[1]At least one filter array (i.e. propertyReferences, tagReferences, or SuppressionReferences) must be defined for each subscription.
[2]At least one filter array (i.e. propertyReferences, tagReferences, or SuppressionReferences) must be defined for each subscription.

Parameter Name	Type	Required (C-create/M-modify/ D-delete)	Description
			Each suppression condition contains the following items: 🔹 suppressionID–specify the global or private suppression reference owned by you (see propertyID | tagID | suppressionID); 🔹 operator–see operator; 🔹 values–see values; 🔹 revokeBy (optional)–the time when the suppression condition expires (in Unix-time format); if revokeBy is not specified, then the suppression condition does not expire. Refer to Filter and Suppression Conditions for details on how to define conditions.
webhook	🔹 uri 🔹 authType 🔹 authValue		Specify the webhook definition to an external application, where your personalized recommendations will be sent. See *Analysis: Webhook: Request Body Parameters* (Help Topic ID 340490) for details of each parameter in the webhook definition. If you do not specify a webhook definition, then notifications for this subscription are not triggered.
schedule	🔹 dayOfMonth 🔹 dayOfWeek		Specify the frequency of the subscription notifications. If no schedule is specified, then the subscription notification is triggered after daily Densify analyses are completed and reporting database tables are updated. Typically, these processes occur at night. The schedule parameter is defined by one or both of the following options: 🔹 dayOfMonth–trigger the notification on the specified day of the month (e.g. `"dayOfMonth" : [10,20]`); 🔹 dayOfWeek–trigger the notification on the specified day of the week (e.g. `"dayOfWeek" : [2,3]`, where 7=Sunday, 1=Monday, 2=Tuesday, etc.). If both scheduling options are defined, the scheduling options are logically AND'ed to compute the notification schedule. For example, if dayOfMonth = [1, 10, 20] and dayofWeek = [2, 4], then notifications will trigger on: 🔹 the 1st, 10th, and 20th of the calendar month, and only if 🔹 the day is a Tuesday or a Thursday.
returnStructure	🔹 properties: • propertyID • useAlias 🔹 tags: • tagID • useAlias		Specify property and attribute tag references to be included in the output of the subscription notification. The `properties` array contains the following items for each property: 🔹 propertyID–the property reference from the subscriptions property catalog; 🔹 useAlias–[true|false] indicates if the property alias is

Parameter Name	Type	Required (C-create/M-modify/ D-delete)	Description
			used as the element label in the subscription output.
			The `tags` array contains the following items per tag: • tagID–the tagreference from the subscriptions attribute tags catalog; • usesAlias–[true|false] specifies if the tag alias is used as the element label in the subscription output.
			If `returnStructure` is defined, then only the properties and tags in `returnStructure` are returned in the output of the subscription notification.
			The platform of the tags or properties defined in the `returnStructure` must correspond with the platform of the subscription. For example, you must only use cloud tag or properties for a cloud subscription.
			See `returnStructure` inside the POST request in Example: Creating a New Subscription.
			If `returnStructure` is not defined, then the full list of possible recommendation properties are returned. The full list of recommendation properties correspond to the technology-specific Analysis recommendation response schema. For example, see *Analysis: AWS Recommendations: Response* (Help Topic ID 340470) for a full list of AWS recommendation properties.
			For `auditInfo` and `dataQuality` property structures, you must specify the final properties to return by using the dot-walking syntax for each substructure. For example, if you want all properties of `dataCollection` substructure within `auditInfo` to be returned, you need to specify all the final properties in `auditInfo.dataCollection`: • `auditInfo.dataCollection.dateFirstAudited` • `auditInfo.dataCollection.dateLastAudited` • `auditInfo.dataCollection.auditCount`
			Note: *Resource attribute string values containing double quotes will be converted to single quotes in the subscription results output.*

Filter and Suppression Conditions

Each filter or suppression condition is defined as a set of three parameters:

```
{
    "propertyID | tagID | suppressionID": "<GUID>",
    "operator": "<supported logical operator>",
    "values": [<array of values>]
}
```

The propertyID | tagID | suppressionID are the unique identifiers to the property, tag or suppression in your Subscriptions catalogs. You can only reference identifiers that are global or privately owned by you. The system property or tag values are compared to the values array provided, based on the logical operator.

For example, if you want to filter cloud systems where "predictedUptime" is between 50-100%:

1. First, find the `propertyRef` for the "predictedUptime" property in your Cloud Subscriptions Property catalog:

```
{
    "propertyRef": "f2a38773-db60-478a-9982-1a2d1ba7d380",
    "propertyName": "predictedUptime",
    "aliasName": "propertyPredictedUptime",
    "owner": ""
}
```

2. Use the `propertyRef` identifier (in the previous step) for the `propertyReference` condition in your subscription:

```
"propertyReferences": [
    {
    "propertyID": "f2a38773-db60-478a-9982-1a2d1ba7d380",
    "operator": "[]",
    "values": [50,100]
    }
]
```

A subscription can have multiple filter conditions to customize the resulting data set. Each condition referencing an individual property, tag, or operator is logically *AND*'ed together during evaluation. Conditions referencing the same property or tag with the same logical operator are *OR*'ed together during evaluation. Systems matching the suppression conditions are removed from the resulting data set.

> *Note:* You cannot compare entire multi-structure recommendation elements, such as `auditInfo` and `dataQuality`. You can only use their final substructure properties in the subscription filter condition, such as `auditInfo.dataCollection.dateFirstAudited` or `dataQuality.completedDays`.

Table: Logical Evaluation of Multiple Conditions

Logical Evaluation	Property/Tag/Suppression	Operator
AND	different property/tag/suppression	different or same operator

Logical Evaluation	Property/Tag/Suppression	Operator
OR	▪ same property/tag/suppression	▪ same operator

Example: Multiple Filter Conditions

For example, if you have the following conditions:

▪ propertyReferences =

- (propertyA > value1)
- (propertyB = value2)
- (propertyB >= value3)

▪ tagReferences =

- (tagC = value4)
- (tagD = value5)
- (tagC = value6)
- (tagC = value7)

▪ suppressionReferences =

- (suppressionE–>propertyE = value8)
- (suppressionF–>tagC = value9)

then the resulting subscription data set would be an intersect of the following groups:

▪ Find all the systems where (propertyA > value1) AND (propertyB = value2) AND (propertyB >= value3)

AND

▪ Find all the systems where ((tagC = value4) OR (tagC = value6) OR (tagC = value7)) AND (tagD = value5)

AND

▪ Remove all systems where (propertyE = value8) AND (tagC = value9)

> **Note:** *Suppression conditions are evaluated after property and tag conditions.*

Table: Subscriptions Filter and Suppression Condition Parameters

Parameter Name	Type	Required (Y/N)	Description
propertyID | tagID | suppressionID	string	Y	Specify the reference ID of the property, tag, or suppression from the corresponding catalog: ▪ See *Subscriptions: Properties* (Help Topic ID 340710) to retrieve propertyIDs available in the platform-specific Subscriptions Properties catalog. ▪ See *Subscriptions: Tags* (Help Topic ID 340720) to retrieve tagIDs available in the platform-specific Subscriptions Tags catalog. ▪ See *Subscriptions: Suppressions* (Help Topic ID

Parameter Name	Type	Required (Y/N)	Description
			340730) to retrieve suppressionIDs available in the platform-specific Subscriptions Suppressions catalog. You can only use global reference IDs or reference IDs that are owned by you.
operator	one of • = • < • > • <= • >= • like • [] • () • (] • [)	Y	Specify a logical operator for the property or tag comparison. The following operators are supported: • "=" –equal to; • "<" –lesser than; • ">" –greater than; • "<=" –less than or equal to; • ">=" –greater than or equal to; • "like" –matches to the provided substring; the "*" wildcard character can be used to anchor substring searches; • "[]" –matches to values contained in the provided inclusive range, e.g. "values" : [1,3] matches to {1, 2, 3}; • "()" –matches to values in the provided exclusive range, e.g. "values": [3-8] matches to {4, 5, 6, 7}; • "(]" –matches to values in the provided exclusive beginning and inclusive ending range, e.g. "values": [2,5] matches to {3,4,5}; • "[)" –matches to values in the provided inclusive beginning and exclusive ending range, e.g. "values": [1, 3] matches to {1,2};
values	an array or range of • string • integer • float	Y	Specify an array of values for the logical comparison. For numeric ranges, specify the beginning and the end range values, e.g. "values": [1,10] implies from 1 to 10. For multiple values, the filter condition considers the list as a series of *OR* clauses. For example, if you are comparing serviceType = ["EC2", "RDS", "Spot"], then the condition is evaluated as [serviceType = "EC2" OR serviceType = "RDS" OR serviceType = "Spot"]. For the "like" operator, use the "*" wildcard character to match zero, one or multiple characters to one end of the substring, e.g. "values": ["t2*"] matches to "t2", "t2.5", or "t2.micro", but not to "rds-t2.large". Using the "like" operator without "*" wildcard will match the substring to any part of the property or tag value, e.g. "values": ["size"] matches to "Resize", "Upsize-Family", or "size-right". Refer to the various cloud recommendation resource elements for valid values: Analysis: AWS Recommendations: Response on page 86, Analysis: Azure Recommendations: Response on page 125, and Analysis: GCP Recommendations: Response on page 171. **Note:** *To filter "Just Right" recommendations or "Not Analyzed" systems, you must use the exact*

Parameter Name	Type	Required (Y/N)	Description
			recommendationType property string, `"Not Analyzed"` *or* `"Just Right"`, *when you use the "=" or "like" operator.*

Core and Ancillary Properties in Filter Conditions

The concept of core and ancillary properties is taken into consideration only for filtering purposes. Ancillary properties cannot stand alone in a Subscriptions property condition set; they must be used in conjunction with a core property for the purpose of filtering subscription returned results.

In any Subscriptions property condition set, you must have at least one core property condition. If you want to only filter subscription data based on an ancillary property, you can add an *always-true* condition with a core property.

Refer to Default Cloud Properties on page 315 for the list of core and ancillary properties.

For example:

If you want to create a subscription to only review systems with the same recommendation for over 30 days, you would create a property filter condition with `recommSeenCount` > 30. Since `recommSeenCount` is an ancillary property, you would also need to add another *always-true* condition with a core property, such as `savingsEstimate` > -10000. This will couple an ancillary property with a core property in your condition set. Therefore, your resulting subscription property condition set for this example would be:

`propertyReferences` =

- (`recommSeenCount` > 30)

- (`savingsEstimate` > -10000)

After finding the `propertyID` for `recommSeenCount` and `savingsEstimate` in your Subscriptions Properties catalog, you can form the `propertyReferences` filter:

```
"propertyReferences": [
    {
    "propertyID": "1dd76248-a503-40b1-9303-e990bbad7818",
    "operator": ">",
    "values": [30]
    },
    {
    "propertyID": "cebcd841-89d8-4007-a4c6-1f0b06723db4",
    "operator": ">",
    "values": [-10000]
    }
]
```

Response

The following is a complete list of possible response elements returned for the
`/subscriptions/<platformType>` resource.

Table: Subscriptions Response Schema

Element	Type	Filter/Sort	Description
Subscription Identification			
subscriptionRef	string	F	The unique referenced ID of the Densify subscription.
subscriptionName	string		The subscription name.
description	string		The subscription description.
owner	string	F	The designated owner of the subscription. The subscription is considered *global* if this element is empty and *private* otherwise.
Subscription Configuration			
outputType	string		`application/json` is the only supported output type.
active	string: `[true|false]`		Indicates if the subscription is active or dormant: ■ true (default)—The subscription is active and notification will trigger at the scheduled time to the webhook defined. ■ false—The subscription is dormant: no subscription notification will trigger, even if webhook and schedule parameters are defined.
webhook	■ uri ■ authType ■ authValue		The webhook definition to an external application, where your personalized recommendations will be sent. See *Analysis: Webhook* in the *Request Body Parameters* section (Help Topic ID 340490) for details of each parameter in the webhook definition. If a webhook definition is not specified, then no notifications are triggered for this subscription.
propertyReferences	array of ■ propertyID ■ operator ■ values		An array of property conditions to be evaluated on the set of system recommendations before including them in the returned data set. For details of the `propertyID` listed in the condition, see *Subscriptions: Properties* (Help Topic ID 340710) to retrieve property definitions available in the platform-specific Subscriptions Properties catalog. See Filter and Suppression Conditions for details on how property conditions are defined.
tagReferences	array of		An array of tag conditions to be evaluated on

Element	Type	Filter/Sort	Description
	 • tagID • operator • values		system attributes before including the system in the returned subscription data set. For details of the `tagID` listed in the condition, see *Subscriptions: Tags* (Help Topic ID 340720) to retrieve tag definitions available in the platform-specific Subscriptions Tags catalog. See Filter and Suppression Conditions for details on how tag conditions are defined.
suppressionReferences	array of • suppressionID • operator • values • revokeBy (optional)		An array of suppression conditions to determine system recommendations excluded from the returned subscription data set. Revoked suppression conditions (i.e. when `revokeBy` datetime has passed) are not taken into consideration during the results data set generation. For details of the `suppressionID` listed in the condition, see *Subscriptions: Suppressions* (Help Topic ID 340730) to retrieve suppression definitions available in the platform-specific Subscriptions Suppressions catalog. See Filter and Suppression Conditions for details on how suppression conditions are defined.
schedule	• dayOfMonth • dayOfWeek		The scheduled frequency of the subscription notification. If no schedule is specified, then the subscription notification is triggered after daily Densify analyses are completed and reporting database tables are updated. Typically, these processes occur at night. The schedule parameter is defined by one or both of the following options: • dayOfMonth–trigger the notification on the specified day of the month (e.g. `"dayOfMonth" : [10,20]`); • dayOfWeek–trigger the notification on the specified day of the week (e.g. `"dayOfWeek" : [2,3]`, where 7=Sunday, 1=Monday, 2=Tuesday, etc.). If both scheduling options are defined, the scheduling options are logical AND'ed to compute the notification schedule.
returnStructure	array of property references and an array of attribute tag references: • properties • propertyID		Arrays of property and attribute tag references to be included in the output of the subscription notification. The following items are listed for each property in the properties array: • propertyID–the property reference from

Element	Type	Filter/Sort	Description

• useAlias
■ tags
   • tagID
   • useAlias
			`/subscriptions/<platformType>`
			`/properties;`
			■ useAlias–[true|false] indicates if the property alias is used as the title in the recommendation output.
			
			The following items are listed for each attribute tag in the tags array:
			■ tagID–the tag reference from `/subscriptions/<platformType>`
			`/tags;`
			■ useAlias–[true|false] indicates if the tag alias is used as the title in the recommendation output.
			
			If `returnStructure` is defined, then only the properties and tags in `returnStructure` are returned in the output of the subscription notification.
			
			If `returnStructure` is not defined, then the full list of possible recommendation properties are returned. The full list of recommendation properties correspond to the technology-specific Analysis recommendation response schema. For example, see *Analysis: AWS Recommendations: Response* (Help Topic ID 340470) for a full list of AWS recommendation properties.
			
			Note: *You must have at least one core property in your property condition set for the purpose of filtering subscription returned results. Refer to Core and Ancillary Properties in Filter Conditions for details.*

webhookStatus	string		The status, date, and time of the last subscription results pushed to the webhook location: ■ Success–subscription results sent to the webhook successfully; ■ Failure–transmission of subscription results to the webhook failed.
lastTriggered	string		The status, date, and time of the last subscription results request: ■ On-Demand Success–the last request was on-demand and it was successful; ■ On-Demand Failure–the last request was on-demand and it failed to produce results; ■ Scheduled Success–the last request was a successful scheduled subscription event posted to a webhook; ■ Scheduled-Failure–the last request was a failed webhook post of the scheduled subscription event.

Element	Type	Filter/Sort	Description
Returned Error Message			
message	string		The message for the status response is returned.
status	number		The HTTP response code of the request. Possible status values include: ▪ 200–success with request (usually with content in response body); ▪ 204–success with request, no content returned; ▪ 400–bad request (invalid parameters, logical errors); ▪ 401–authentication failed; ▪ 404–resource not found (or no privileges); ▪ 415–unsupported media type; ▪ 500–internal server error.

Examples

Example: Getting a Collection of Subscriptions

The following example shows you how to retrieve your collection of cloud subscriptions. Assuming that your username is "saas", the returned collection is a set of global subscriptions and subscriptions that belong to the "saas" user. You are not able to see private subscriptions belonging to other Densify users.

> **Note:** If you are an **administrative user**[1] , then the returned collection will be all global cloud subscriptions and private cloud subscriptions for all Densify users.

Example: Getting a Collection of Cloud Subscriptions

Request:

```
GET /subscriptions/cloud
```

Response:

```
[
  {
    "subscriptionRef": "2db7753a-26a3-4cba-be84-8e4e3a12daa1",
```

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

```
      "subscriptionName": "MyOwnSubscription",
      "description": "My first saas private subscription.",
      "owner": "saas",
      "outputType": "application/json",
      "active": true,
      "propertyReferences": [
          {
              "values": ["55"],
              "propertyID": "e3ad3bcd-e66d-40aa-8c55-af1f033fdb13",
              "operator": ">"
          },
          {
              "values": ["asg", "ec2"],
              "propertyID": "d17c1058-e346-4e12-a8c3-d4a440d34cfe",
              "operator": "="
          }
      ],
      "returnStructure": {},
      "webhookStatus": "",
      "lastTriggered": "On-Demand Success. Wed Dec 18 16:26:52 EST 2019"
  },
  {
      "subscriptionRef": "4503e5ff-7a1e-4f4c-a106-0f31ca38dc22",
      "subscriptionName": "1GlobalSubscription",
      "description": "Shared global subscription - 1.",
      "owner": "",
      "outputType": "application/json",
      "active": true,
      "webhook": {
              "uri": "http://localhost:8080/sampletest",
              "authType": "basic",
              "authValue": "test:test1"
          },
      "propertyReferences": [
          {
              "values": ["asg"],
              "propertyID": "d17c1058-e346-4e12-a8c3-d4a440d34cfe",
              "operator": "="
          },
          {
              "values": ["100"],
              "propertyID": "e3ad3bcd-e66d-40aa-8c55-af1f033fdb13",
              "operator": ">"
          }
      ],
      "tagReferences": [
          {
              "values": [
                 "229",
                 "524"
                 ],
              "tagID": "0eece6ab-9794-4b12-9868-ec2fa8e9d721",
              "operator": "like"
          }
      ],
      "suppressionReferences": [
          {
```

```
            "values": [
               "Terminate"
               ],
            "revokeBy": 1576645200000,
            "suppressionID": "abc57b21-d85f-4e80-8da4-377551cbf089",
            "operator": "="
      }
    ],
    "schedule": {
            "dayofWeek": [
               1,
               3,
               5
            ]
    },
    "returnStructure": {
       "properties": [
            {
               "useAlias": true,
               "propertyID": "96214999-5104-41b5-88e9-1681d1900e42"
            },
            {
               "useAlias": true,
               "propertyID": "ebcfa581-7c12-4e56-8eb5-3c2225941cbf"
            },
            {
               "propertyID": "e3ad3bcd-e66d-40aa-8c55-af1f033fdb13"
            },
            {
               "propertyID": "1a2c49b6-a7aa-4953-800b-f4fded2339a2"
            }
       ],
       "tags": [
            {
               "useAlias": true,
               "tagID": "5f2ff6aa-864b-4063-948e-388ad42e8f13"
            }
       ]
    },
    "webhookStatus": "Failure. java.net.ConnectException: Connection refused
(Connection refused) Thu Jan 02 16:41:53 EST 2020",
    "lastTriggered": "On-Demand Success. Thu Jan 02 16:40:33 EST 2020"
  }
]
```

Example: Getting a Collection of Subscriptions for a Specific User

If you are an **administrative user**[1], you can use the `owner` query string parameter to get a collection of subscriptions for a particular user. The following example shows you how to retrieve a collection of cloud subscriptions for the "saas" user.

Example: Getting a Collection of Cloud Subscriptions for "saas" User

Request:

```
GET /subscriptions/cloud?type=owner&owner=saas
```

Response:

```
[
  {
    "subscriptionRef": "2db7753a-26a3-4cba-be84-8e4e3a12daa1",
    "subscriptionName": "MyOwnSubscription",
    "description": "My first saas private subscription.",
    "owner": "saas",
    "outputType": "application/json",
    "active": true,
    "propertyReferences": [
      {
        "values": ["55"],
        "propertyID": "e3ad3bcd-e66d-40aa-8c55-af1f033fdb13",
        "operator": ">"
      },
      {
        "values": ["asg", "ec2"],
        "propertyID": "d17c1058-e346-4e12-a8c3-d4a440d34cfe",
        "operator": "="
      }
    ],
    "returnStructure": {},
    "webhookStatus": "",
    "lastTriggered": "On-Demand Success. Wed Dec 18 16:26:52 EST 2019"
  }
]
```

Example: Creating a New Subscription

This example shows you how to create and define a new cloud subscription. In this sample subscription, we filter for systems where predicted uptime is between 50-100% and suppress systems

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

that have a "Terminate" recommendation.

Example: Creating a New Cloud Subscription

Request:

```
POST /subscriptions/cloud
```

Parameters:

```
[
    {
    "subscriptionName": "Sample Subscription",
    "description": "A subscription for testing",
    "active": true,
    "webhook": {
        "uri": "https://192.168.100.100:443/test/webhook",
        "authType": "basic",
        "authValue": "test:test"
    },
    "propertyReferences": [
        {
        "propertyID": "f2a38773-db60-478a-9982-1a2d1ba7d380",
        "operator": "[]",
        "values": [50,100]
        }
    ],
    "suppressionReferences": [
        {
        "suppressionID": "a6827ae4-fa2b-405e-a564-d70f2dad45c2",
        "operator": "=",
        "values": ["Terminate"]
        }
    ],
    "returnStructure": {
        "properties": [
            {
            "propertyID": "cebcd841-89d8-4007-a4c6-1f0b06723db4"
            },
            {
            "propertyID": "183b1f50-a20b-4d29-a488-8ba53bbf7c40"
            },
            {
            "useAlias": true,
            "propertyID": "b4e7260f-1aae-4150-b75d-1b1234075500"
            },
            {
            "useAlias": true,
            "propertyID": "e2ae92c5-91c7-4ff5-a29f-99aa92d65178"
            }
        ],
        "tags": [
            {
            "tagID": "8a556754-f7d2-4098-a3f5-c6777e2be697"
            }
```

```
            ]
        }
        }
    ]
```

Response:

```
[
    {
        "subscriptionRef": "731cd17f-c2c6-4aef-8bdb-84ac05bf8dff",
        "SubscriptionName": "Sample Subscription"
    }
]
```

Example: Modifying a Subscription

This example shows you modify an existing cloud subscription and assumes that you are the "saas" user. You have to specify all the required parameters (including the `owner` parameter) for the subscription since previous parameters are deleted or reset to default in a PUT request. In the example below, the subscription name, description and returned elements are updated from the original subscription (in the previous <u>Example: Creating a New Subscription</u>).

Example: Modifying a Cloud Subscription

Request:

```
PUT /subscriptions/cloud/731cd17f-c2c6-4aef-8bdb-84ac05bf8dff
```

Parameters:

```
{
    "subscriptionName": "Sample Subscription Modified",
    "description": "A modified subscription for testing",
    "owner": "saas",
    "active": true,
    "webhook": {
        "uri": "https://192.168.100.100:443/test/webhook",
        "authType": "basic",
        "authValue": "test:test"
        },
    "propertyReferences": [
        {
        "propertyID": "f2a38773-db60-478a-9982-1a2d1ba7d380",
        "operator": "[]",
        "values": [50,100]
        }
    ],
```

```
    "suppressionReferences": [
        {
        "suppressionID": "a6827ae4-fa2b-405e-a564-d70f2dad45c2",
        "operator": "=",
        "values": ["Terminate"]
        }
    ],
    "returnStructure": {
        "properties": [
            {
            "propertyID": "08d84679-1816-4cd7-b766-e4ad441b9a6c"
            },
            {
            "propertyID": "cebcd841-89d8-4007-a4c6-1f0b06723db4"
            },
            {
            "propertyID": "183b1f50-a20b-4d29-a488-8ba53bbf7c40"
            },
            {
            "useAlias": true,
            "propertyID": "b4e7260f-1aae-4150-b75d-1b1234075500"
            },
            {
            "useAlias": true,
            "propertyID": "e2ae92c5-91c7-4ff5-a29f-99aa92d65178"
            },
            {
            "propertyID": "e2dc1f96-7149-4767-894b-23e65485e314"
            },
            {
            "useAlias": true,
            "propertyID": "1dd76248-a503-40b1-9303-e990bbad7818"
            }
        ],
        "tags": [
            {
            "tagID": "8a556754-f7d2-4098-a3f5-c6777e2be697"
            }
        ]
    }
}
```

Response:

```
{
    "subscriptionRef": "731cd17f-c2c6-4aef-8bdb-84ac05bf8dff",
    "SubscriptionName": "Sample Subscription Modified"
}
```

Example: Deleting a Subscription

This example shows you how to delete a subscription. An HTTP "204 No Content" response is returned for a successful deletion.

Example: Deleting a Cloud Subscription

Request:

```
DELETE /subscriptions/cloud/9f58839a-dcad-4581-b883-051f729e3e1c
```

Subscriptions: Results

Description

The `/subscriptions/<platformType>/<subscriptionRef>` resource is used to dynamically retrieve your latest personalized Densify subscription recommendations. This resource allows you to review your platform-specific subscription results on-demand. The same information is delivered to a third-party application for targeted distribution if the third-party application is defined in the subscription's webhook.

To see how subscriptions are defined, refer to *Subscriptions* (Help Topic ID 340690).

Resource

```
/subscriptions/cloud/<subscriptionRef>
/subscriptions/containers/<subscriptionRef>
/subscriptions/<subscriptionRef>
```

> **Note:** *If you use this resource without the `<platformType>` specified (i.e. without `cloud` or `containers` specified), the behavior is exactly the same as specifying the `cloud`-specific resource. This behavior enables backward compatibility with scripts using the Densify API prior to release 12.1.6, where the platform-specific indicator was not available.*

Supported Operations

Table: Subscription Results Supported Operations

HTTP Method	Input	Output	Description

GET /subscriptions/ <platformType> /< subscriptionRef >	Path Parameter:
	■ platformType
	■ subscriptionRef
	
	Query String Parameter:
	■ divider
	■ limit
		subscription:
		■ subscription
		  • name
		  • description
		  • created
		  • createdBy
		  • updated
		  • updatedBy
		  • lastRefreshed
		  • owner
		■ count
		■ results
		• collection of personalized recommendations in the format defined by *returnStructure*
			Returns the current results of the platform-specific subscription notification, identified by *<subscriptionRef>*.
			
			The returned output is the same information that is delivered to the webhook application if the subscription is scheduled to run at the current time. Note that the following parameters are ignored for on-demand subscription results request:
			■ active—results are always returned for on-demand requests, even if "active": "false";
			■ webhook—results are always returned for on-demand requests, even if webhook is undefined;
			■ schedule—results are always returned for on-demand requests, regardless of the notification schedule defined.
			
			An appropriate error message and HTTP code are returned for failed requests. See Returned Status
			
			See Example: Getting On-Demand Subscription Cloud Results

Parameters

Path Parameters

Table: Subscription Results Path Parameters

Parameter Name	Type	Required	Description
platformType	string	Y	[cloud|containers] Specify the technology platform for the subscription results resource.
subscriptionRef	string	Y	Specify the unique subscription identifier.

Query String Parameters

Table: Subscription Results Query String Parameters

Parameter Name	Type	Required (Y/N)	Description

divider	string
	
	`[true|false]`
			An option to display a divider that separates properties and tags for each system in the output:
			
			🔹 true (default)–show the divider:
			`"divider": "--------------------`
			`---",`
			🔹 false–do not show the divider.
			
			Note: *The divider is not displayed in the output for scheduled subscription results sent to webhooks.*

limit	number range [1:30000]		An option to set the limit of returned results. By default, the limit is set to 3000. This implies that the number of results returned is limited to 3000 systems. If a subscription's result count is more than 3000 system recommendations, an error is returned with code 400. If this is the case, you can adjust your on-demand subscription request to a higher limit. For example: `?limit=8000`. The range of the limit value is from 1 to 30000. *Note:* *This limit does not apply to subscription recommendations sent to webhooks. It only applies to on-demand subscription queries, and hence, will negatively impact web server performance if the limit is set too high.* See Example: On-Demand Subscription Results Count Exceeding Limit.

Response

The following is a complete list of possible response elements returned for the `/subscriptions/<platformType>/<subscriptionRef>` resource.

Table: Subscription Results Response Schema

Element	Type	Filter/Sort	Description
\multicolumn Subscription Information			
subscription	🔹 name 🔹 description 🔹 created 🔹 createdBy 🔹 updated		Subscription header information. See: 🔹 name 🔹 description 🔹 created 🔹 createdBy

Element	Type	Filter/Sort	Description
	🔖 updatedBy 🔖 lastRefreshed 🔖 owner		🔖 updated 🔖 updatedBy 🔖 lastRefreshed 🔖 owner
name	string		The subscription name.
description	string		The subscription description.
created	datetime		The date and time when the subscription entity was created.
createdBy	string		The authenticated Densify user who created the subscription.
updated	datetime		The date and time when the subscription was last modified.
updatedBy	string		The authenticated Densify user who modified the subscription.
lastRefreshed	datetime		The date, and time of the last recommendation analysis. This is typically scheduled to run nightly after data collection.
owner	string		The designated owner of the subscription. This field contains the Densify username of the owner. The subscription is considered *global* if this element is empty and *private* otherwise.
Subscription Results			
count	number		The number of system recommendations in the results collection.
results	array of returnStructure		An array of system results, with properties and tags defined in the subscription's `returnStructure`. Refer to *Subscriptions: returnStructure* (Help Topic ID 340690) for details on how the returnStructure can be defined. Below are the possible elements returned for each system recommendation: 🔖 properties–the list of properties defined in the returnStructure (if no returnStructure is defined, all recommendation properties are returned); 🔖 divider–the divider element between properties and tags (if defined): `"divider": "------------------------",` 🔖 tags–the list of attribute tags defined in the returnStructure. Refer to the various cloud or container recommendation resource topics for a description of each result element: Analysis: AWS Recommendations: Response on page 86, Analysis: Azure Recommendations: Response on page 125, Analysis: GCP Recommendations: Response on page 171, and Analysis: Kubernetes Container Recommendations: Response on page 188.
Returned Status			
message	string		The message for the status response is returned. See Example: On-Demand Subscription Results Count Exceeding Limit for an example of HTTP 400 Bad

Element	Type	Filter/Sort	Description
			Request error message.
status	number		The HTTP response code of the request. Possible status values include: ■ 200–success with request (usually with content in response body); ■ 204–success with request, no content returned; ■ 400–bad request (invalid parameters, logical errors); ■ 401–authentication failed; ■ 404–resource not found (or no privileges); ■ 415–unsupported media type; ■ 500–internal server error.

Examples

Example: Getting On-Demand Subscription Cloud Results

The following example shows you how to retrieve your subscription cloud results, on-demand.

Example: Getting Cloud Subscription Results On-Demand

Request:

```
GET /subscriptions/cloud/fa3be33c-fbb7-4912-88c2-a83ee58e444e
```

Response:

```
{
    "subscription": {
        "name": "Sample Subscription",
        "description": "A subscription for testing",
        "created": "Mon Jan 19 13:52:31 EST 2020",
        "createdBy": "saas",
        "updated": "Mon Jan 20 14:32:38 EST 2020",
        "updatedBy": "SaaSadmin",
        "lastRefreshed": "Mon Jan 20 01:32:59 EST 2020",
        "owner": "saas"
        },
    "count": 452,
    "results": [
        {
          "currentType": "standard_d2",
          "name": "st01-prepro-edge-307",
          "recommendationType": "Modernize - Optimal Family",
          "savingsEstimate": "43.850475",
```

```
            "serviceType": "Virtual Machine",
            "divider": "-----------------------",
            "Availability Zone": "eastus+group"
        },
        {
            "currentType": "r3.2xlarge",
            "name": "ex-prepro-stoc-384",
            "recommendationType": "Downsize",
            "savingsEstimate": "75.36128",
            "serviceType": "EC2",
            "divider": "-----------------------",
            "Resource Tags":
              [
               "Owner : Bob Snow",
               "Product Code : PR000263",
               "Environment : Pre-prod",
               "Inventory Code : prepro-stoc",
               "Name : ex-prepro-stoc-384"
              ],
            "Availability Zone": "us-east-1d"
        },
        {
         ...
         ...
        }
    ]
}
```

Example: On-Demand Subscription Results Count Exceeding Limit

If your on-demand Subscriptions request contains a large number of results, exceeding the results limit, you will encounter an HTTP 400 Bad Request error message with the count limit.

Example: On-Demand Subscription Results Exceed Limit

Request:

```
GET /subscriptions/cloud/489331bd-5db9-408d-b7a0-2f3f8b1f66e9
```

Response:

```
{
    "message": "On-Demand Failure. The return count of 3891 has exceeded
object return limit of 3000. Update your call with a new limit value. Wed Jul
29 09:05:15 EDT 2020",
    "status": 400
}
```

This error is returned only for on-demand Subscriptions results requests with results count exceeding the default count limit. Subscription results sent to webhooks do not have a count limit. You can override this limit by requesting the on-demand results again with an increased limit count in the URL query string. Note that increasing the results limit may affect response time of the Densify web server.

Request with increased limit count:

```
GET /subscriptions/cloud/489331bd-5db9-408d-b7a0-2f3f8b1f66e9?limit=4000
```

Subscriptions: Properties

Description

The `/subscriptions/<platformType>/properties` resource is used to store a list of recommendation properties available to the Subscriptions resource for the purpose of filtering recommendations from the resulting subscription data set.

This resource acts as a catalog of recommendation properties to be referenced by the propertyReferences parameter in a subscription. Properties not defined in the `/subscriptions/<platformType>/properties` resource, cannot be referenced by a subscription property filter condition. See *Subscriptions* (Help Topic ID 340690) for details on defining a subscription and using property filter conditions.

There is a catalog for each supported `<platformType>`, which can only be referenced by the corresponding `<platformType>` subscription. For example, a container subscription (i.e. `/subscriptions/containers`) can only reference properties from the Container Subscriptions Properties catalog (i.e. `/subscriptions/containers/properties`).

Densify provides a default set of properties for each `<platformType>` catalog. You can extend these default sets by adding additional properties, or you can customize the sets by modify or deleting existing properties.

For the full set of available properties to add to the Cloud Subscriptions Properties catalog, refer to the cloud recommendation response schema:

- See *Analysis: AWS Recommendations Response* (Help Topic ID 340470) for a list of all the recommendation elements available for the AWS instances.

- See *Analysis: Azure Recommendations Response* (Help Topic ID 340510) for a list of all the recommendation elements available for the Azure instances.

- See *Analysis: GCP Recommendations Response* (Help Topic ID 340500) for a list of all the recommendation elements available for the GCP instances.

See Default Cloud Properties on page 315 for the list of default cloud properties.

> **Note:** Some recommendation elements are not common to all technologies. It is good practice to indicate the technology for an element that is technology-specific in the *aliasName* (e.g. aliasName = "AWS minGroupRecommended"). This practice is helpful when you use the Cloud Subscriptions Properties catalog to form property conditions.

For the full set of available properties to add to the Container Subscriptions Properties catalog, refer to the container recommendation response schema:

- See *Analysis: Kubernetes Container Recommendations: Response* (Help Topic ID 340570) for a list of all the recommendation elements available for Kubernetes containers.

See Default Container Properties on page 320 for the list of default cloud properties.

Properties in the platform-specific Subscriptions Properties catalogs can be declared as: *global* or *private* (i.e. user-specific). Global properties can be used by any API enabled user, while private properties can only be used by their owners. An exception to this rule is that **administrative users**[1] have access to all properties - global or private user-specific.

Resource

```
/subscriptions/cloud/properties
/subscriptions/containers/properties
/subscriptions/properties
```

> **Note:** If you use this resource without the `<platformType>` specified (i.e. without `cloud` or `containers` specified), the behavior is exactly the same as specifying the `cloud`-specific resource. This behavior enables backward compatibility with scripts using the Densify API prior to release 12.1.6, where the platform-specific indicator was not available.

Supported Operations

Table: Subscriptions Properties Supported Operations

HTTP Method	Input	Output	Description
`GET` `/subscriptions/` `<platformType>/properties`	Path Parameter: - platformType Query String Parameter Options: - type - owner	Collection of - propertyRef - propertyName - aliasName - owner	Returns a list of existing properties in the platform-specific Subscriptions Properties catalog. - Use the type query string parameter to return only global or only private properties. If type is not specified, all global and only private properties

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
			belonging to you are returned. ■ **Administrative users**[1] can use the owner query string parameter to return all the private properties belonging to a specific user. See Example: Getting a Collection of Private Cloud Subscriptions Properties on page 323.
GET /subscriptions/ *\<platformType\>* /properties/\<propertyRef\>	Path Parameters: ■ platformType ■ propertyRef	■ propertyRef ■ propertyName ■ aliasName ■ owner	Returns a Subscriptions property with unique identifier \<propertyRef\> from a platform-specific Subscriptions Properties catalog. See Example: Getting a Specific Container Subscriptions Property on page 323.
POST /subscriptions/ *\<platformType\>*/properties	Path Parameter: ■ platformType Request Body Parameters: Collection of ■ propertyName ■ aliasName ■ owner	Collection of ■ propertyRef ■ propertyName	Adds new recommendation properties into a platform-specific Subscriptions Properties catalog. ■ **Administrative users**[2] can add global or private properties. ■ Non-administrative users can only add private properties. ■ propertyName is validated for uniqueness within a catalog (i.e. you cannot define two properties with the same propertyName). ■ aliasName is validated for uniqueness, depending on the private or global scope of the property (i.e. two users can use the same aliasName for their private property). ■ The properties bulk-add operation is committed as a whole unit. One property add error will roll back the

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.
[2]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
			entire bulk add operation. ▪ Platform-specific properties must be added into the same platform-specific Subscriptions Properties catalog. For example, you can only add container properties into the Container Subscriptions Properties catalog. See Example: Adding New Subscriptions Properties on page 324.
PUT /subscriptions/ *\<platformType>*/properties	Path Parameter: ▪ platformType Request Body Parameters: Collection of ▪ propertyRef ▪ propertyName ▪ aliasName ▪ owner	Collection of ▪ propertyRef ▪ propertyName	Replaces parameters for existing properties in a platform-specific Subscriptions Properties catalog. You must specify all parameters required for the property you want to update. ▪ The \<propertyRef> parameter is used to identify the property to update; hence it cannot be modified. ▪ Only an **administrative user**[1] can change the owner parameter to promote a private property to a global property (i.e. set owner=""). ▪ If you are not an administrative user, you can only set the owner parameter to your username. ▪ propertyName is validated for uniqueness within a catalog. ▪ aliasName is validated for uniqueness, depending on the private or global scope of the property (i.e. two users can use the same aliasName for heir private property). ▪ The bulk-edit operation is committed as a whole unit; an error resulting from one

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
			of the edits will roll back the entire bulk edit operation. See Example: Modifying Subscriptions Properties on page 324 **Note:** *If you are not an administrative user, you can only modify your own private properties.*

`PUT`
`/subscriptions/`
`<platformType>`
`/properties/<propertyRef>`
	Path Parameters:
	• platformType
	• propertyRef
	
	Request Body Parameters:
	• propertyName
	• aliasName
	• owner
		• propertyRef
		• propertyName
			Replaces parameters in an existing property, identified by `<propertyRef>` in a platform-specific Subscriptions Properties catalog.
			
			You must specify all parameters required for the existing property, as all previous parameters are deleted.
			
			• Only an **administrative user**[1] can change the `owner` parameter to promote a private property to a global property (i.e. set `owner=""`).
			• If you are not an administrative user, you can only set the `owner` parameter to your username.
			• `propertyName` is validated for uniqueness within a catalog.
			• `aliasName` is validated for uniqueness, depending on the private or global scope of the property (i.e. two users can use the same `aliasName` for their private property).
			
			See Example: Modifying a Subscriptions Property on page 325.
			
			Note: *If you are not an administrative user,*

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
			you can only modify your own private property.
DELETE /subscriptions/ `<platformType>`/properties	Path Parameter: 　platformType Request Body Parameter: Collection of 　propertyRef	HTTP status of "204 No Content" if all delete operations are successful If delete errors occur, then the following is returned for each property delete request: • propertyRef • message • status	Remove Subscriptions properties from a platform-specific Subscriptions Properties catalog. 　**Administrative users**[1] can delete any global or private properties. 　Non-administrative users can only delete their own private properties from the catalog. 　Properties referenced in filtering conditions by subscriptions cannot be deleted. 　Each property delete operation is independent from the other delete operations in the same request. An error with one property delete action does not affect the delete actions of the other properties in the same request parameter body. See Example: Deleting Subscriptions Properties.
DELETE /subscriptions/ `<platformType>` /properties/`<propertyRef>`	Path Parameter: 　propertyRef	HTTP status of "204 No Content" if delete operation is successful HTTP status of "404 Not Found" if property is not found If the property is referenced by a subscription, then the following elements are returned: • propertyRef	Removes a Subscriptions property with `<propertyRef>` identifier from a platform-specific Subscriptions Properties catalog. 　**Administrative users**[2] can delete any global or any private properties. 　Non-administrative users can only delete their own private properties from the catalog. 　Properties referenced by subscriptions cannot be deleted.

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.
[2]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
		• [message](#) • [status](#)	

Parameters

Path Parameters

Table: Subscriptions Properties Path Parameters

Parameter Name	Type	Required (Y/N)	Description
platformType	string	Y	[cloud|containers] Specify the technology platform for the Subscriptions property resource.
propertyRef	string	Y	Specify the unique identifier for a Subscriptions property.

Query String Parameters

Table: Subscriptions Properties Query String Parameters

Parameter Name	Type	Required (Y/N)	Description
type	string		Specify the type of Subscriptions properties to return: ■ all–Return all properties: global and private user-specific. If you are not an **administrative user**[1], only private properties owned by you and global properties are returned. This is the default behavior if "type" is not specified in the request. ■ global–Return all global Subscriptions properties. ■ owner–Return user-specific Subscriptions properties. If you are not an administrative user, only private properties owned by you are returned. If you are an administrative user, all global and private properties are returned. A Subscriptions property is considered *global* if the [owner](#) parameter is not populated. Global Subscriptions properties can be used by all Densify API users. A Subscriptions property is considered *private* if the owner parameter contains a Densify username. Private Subscriptions properties can only be used by their owners or administrative users.
owner	string		If you are an **administrative user**[2], you can specify a Densify

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.
[2]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Parameter Name	Type	Required (Y/N)	Description
			username in conjunction with the `type=owner` query string parameter to return all of the specified user's private Subscriptions properties. If you are not an administrative user, you can request for your own private properties. If you use the `?type=owner&owner=<anotherusername>` query string option with a username other than your own, the returned response is a 400 Bad Request - "Current login user cannot query for owner" error.

Request Body Parameters

Table: Subscriptions Properties Request Body Parameters

Parameter Name	Type	Required (C-create/M-modify/D-delete)	Description
propertyRef	string	M D	Specify the unique identifier for a Subscriptions property.
propertyName	string	C M	Specify the recommendation element for the Subscriptions property. The list of available recommendation elements can be found in the Response schema section of the *Analysis: technology-specific Recommendations* page. For example, refer to the Response schema section of the Analysis: AWS Recommendations on page 83 for a full list of AWS recommendation elements. The `propertyName` must be unique within a platform-specific Subscriptions Properties catalog.
aliasName	string		Specify an alias name for the Subscriptions property. For *global* properties, the aliasName must be unique per platform-specific catalog. For *private* properties, the aliasName must be unique per owner and across all global Subscriptions properties per platform-specific catalog. For example, owner A and owner B can both have a private property alias named "OptimizedSize", as long as "OptimizedSize" is also not a global property alias within the same catalog.
owner	string	M[1]	When the `owner` parameter is not set, the Subscriptions property is considered *global*. Global Subscriptions properties can be used by all API users. Only administrative users[2] can create global properties. When the `owner` parameter is set, the property is considered *private*. Private Subscriptions properties can only be used by their owners or administrative users.

[1]The `owner` parameter is mandatory for private Subscriptions properties.

[2]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Parameter Name	Type	Required (C-create/M-modify/D-delete)	Description
			If you are an administrative user, you have the ability to assign any Densify user as the owner of the Subscriptions property in a POST request. In a PUT request, administrative users can promote the property from private to global by setting `owner: ""`. If you are not an administrative user, you can only set the `owner` parameter to your Densify username. In a POST request, the `owner` parameter is automatically populated with your username.

Response

The following is a complete list of possible response elements returned for the `/subscriptions/properties` resource.

Table: Subscriptions Properties Response Schema

Element	Type	Filter/Sort	Description
propertyRef	string		The unique referenced ID of the Densify Subscriptions property.
propertyName	string		The Subscriptions property name.
aliasName	string		The Subscriptions property alias name.
owner	string	F	The designated owner of this Subscriptions property. ▪ A Subscriptions property is considered *global* if this parameter is not populated. ▪ A Subscriptions property is considered *private* if the owner parameter contains a Densify username.
message	string		The message for the status response.
status	number		The HTTP response code of the request. Possible status values include: ▪ 200–success with request (usually with content in response body); ▪ 204–success with request, no content returned; ▪ 400–bad request (invalid parameters, logical errors); ▪ 401–authentication failed; ▪ 404–resource not found (or no privileges); ▪ 415–unsupported media type; ▪ 500–internal server error.

Ancillary Properties

The concept of core and ancillary properties is taken into consideration only for filtering purposes. Ancillary properties cannot stand alone in a Subscriptions property condition set; they must be used in conjunction with a core property for the purpose of filtering subscription returned results.

In any Subscriptions property condition set, you must have at least one core property condition. If you want to only filter subscription data based on an ancillary property, you can add an *always-true* condition with a core property.

Refer to Default Cloud Properties on page 315 for the list of core and ancillary properties.

For example, if you want to see recommendations for systems that have been collected for over 10 days, create a property filter condition with `auditInfo.dataCollection.auditCount` > 10. Since `auditInfo.dataCollection.auditCount` is an ancillary property, you need to add another condition with a core property that would evaluate to true for the systems you are interested in, such as `currentCost` > 0 (the current cost of the instance is greater than 0). In this example, the Subscription's property condition set would be:

`propertyReferences` =

- (`auditInfo.dataCollection.auditCount` > 10)

- (`currentCost` > 0)

Refer to Filter and Suppression Conditions for further details on how to define a Subscription's property conditions.

Default Cloud Properties

Default Cloud Subscriptions Properties List

The default Subscriptions Properties catalog contains both core and ancillary properties which can be used by all API users (i.e. default properties are all global). Properties are listed in alphabetic order in the following table.

> **Note:** Ancillary properties must be used in conjunction with at least one core property for the purpose of filtering Subscriptions results. Refer to Ancillary Properties for details.

propertyName	aliasName	Core / Ancillary	Description Reference
accountIdRef	accountIdRef	core	Analysis: AWS Recommendations: accountIdRef on page 86
accountName	accountName	core	The account name, as specified by connectionName, when creating the cloud connection.
approvalType	approvalType	core	Analysis: AWS Recommendations:

propertyName	aliasName	Core / Ancillary	Description Reference
			approvalType on page 89
avgInstanceCountCurrent	avgInstanceCountCurrent	core	Analysis: AWS Recommendations: Analysis: AWS Recommendations on page 83
avgInstanceCountRecommended	avgInstanceCountRecommended	core	Analysis: AWS Recommendations: Analysis: AWS Recommendations on page 83
auditInfo.dataCollection.auditCount	dc_auditCount	ancillary	Analysis: AWS Recommendations: auditInfo on page 95 *Note:* *Ancillary properties must be used in conjunction with at least one core property for the purpose of filtering subscription results. Refer to* Ancillary Properties *for details.*
auditInfo.dataCollection.dateFirstAudited	dc_firstAudited	ancillary	Analysis: AWS Recommendations: auditInfo on page 95
auditInfo.dataCollection.dateLastAudited	dc_lastAudited	ancillary	Analysis: AWS Recommendations: auditInfo on page 95
auditInfo.workloadDataLast30.firstDate	last30_firstDate	ancillary	Analysis: AWS Recommendations: auditInfo on page 95
auditInfo.workloadDataLast30.lastDate	last30_lastDate	ancillary	Analysis: AWS Recommendations: auditInfo on page 95
auditInfo.workloadDataLast30.seenDays	last30_seenDays	ancillary	Analysis: AWS Recommendations: auditInfo on page 95
auditInfo.workloadDataLast30.totalDays	last30_totalDays	ancillary	Analysis: AWS Recommendations: auditInfo on page 95
currentCost	currentCost	core	Analysis: AWS Recommendations: currentCost on page 91
currentDesiredCapacity	currentDesiredCapacity	core	Analysis: AWS Recommendations: currentDesiredCapacity on page 93
currentHourlyRate	currentHourlyRate	core	Analysis: AWS Recommendations: currentHourlyRate on page 91
currentRiCoverage	currentRiCoverage	core	Analysis: AWS

propertyName	aliasName	Core / Ancillary	Description Reference
			Recommendations: currentRiCoverage on page 91
currentType	currentType	core	Analysis: AWS Recommendations: currentType on page 86
dataQuality.completeDays	dq_completeDays	ancillary	Analysis: AWS Recommendations: dataQuality on page 96 *Note:* *Ancillary properties must be used in conjunction with at least one core property for the purpose of filtering subscription results. Refer to Ancillary Properties for details.*
dataQuality.firstSeen	dq_firstSeen	ancillary	Analysis: AWS Recommendations: dataQuality on page 96
dataQuality.lastSeen	dq_lastSeen	ancillary	Analysis: AWS Recommendations: dataQuality on page 96
dataQuality.partialDays	dq_partialDays	ancillary	Analysis: AWS Recommendations: dataQuality on page 96
dataQuality.workloadName	dq_workloadName	ancillary	Analysis: AWS Recommendations: dataQuality on page 96
deferRecommendation	deferRecommendation	core	Analysis: AWS Recommendations: deferRecommendation on page 92
densifyPolicy	densifyPolicy	core	Analysis: AWS Recommendations: densifyPolicy on page 89
effortEstimate	effortEstimate	core	Analysis: AWS Recommendations: effortEstimate on page 90
entityId	entityId	core	Analysis: AWS Recommendations: entityId on page 86
implementationMethod	implementationMethod	core	Analysis: AWS Recommendations: implementationMethod on page 88
maxGroupCurrent	maxGroupCurrent	core	Analysis: AWS Recommendations: minGroupCurrent on page 92

propertyName	aliasName	Core / Ancillary	Description Reference
maxGroupRecommended	maxGroupRecommended	core	Analysis: AWS Recommendations: maxGroupRecommended on page 93
minGroupCurrent	minGroupCurrent	core	Analysis: AWS Recommendations: minGroupCurrent on page 92
minGroupRecommended	minGroupRecommended	core	Analysis: AWS Recommendations: minGroupRecommended on page 92
name	name	core	Analysis: AWS Recommendations: name on page 88 Analysis: Azure Recommendations: name on page 127 Analysis: GCP Recommendations: name on page 172
osName	osName	core	OS name as defined by data collected from each instance.
powerState	powerState	core	Analysis: AWS Recommendations: powerState on page 90
predictedUptime	predictedUptime	core	Analysis: AWS Recommendations: predictedUptime on page 88
recommendationType	recommendationType	core	Analysis: AWS Recommendations: recommendationType on page 86 Analysis: Azure Recommendations: recommendationType on page 125 Analysis: GCP Recommendations: recommendationType on page 171 **Note:** *To filter "Just Right" recommendations or "Not Analyzed" systems, you must use the exact recommendationType property string,* `"Not Analyzed"`

propertyName	aliasName	Core / Ancillary	Description Reference
			or `"Just Right"`, *when you use the "=" or "like" operator.*
recommendedCost	recommendedCost	core	Analysis: AWS Recommendations: recommendedCost on page 91
recommendedHostEntityId	recommendedHostEntityId	core	Analysis: AWS Recommendations: recommendedHostEntityId on page 91
recommendedHourlyRate	recommendedHourlyRate	core	Analysis: AWS Recommendations: recommendedHourlyRate on page 91
recommendedType	recommendedType	core	Analysis: AWS Recommendations: recommendedType on page 88
recommFirstSeen	recommFirstSeen	ancillary	Analysis: AWS Recommendations: recommFirstSeen on page 95. **Note:** *Ancillary properties must be used in conjunction with at least one core property for the purpose of filtering subscription results. Refer to Ancillary Properties for details.*
recommLastSeen	recommLastSeen	ancillary	Analysis: AWS Recommendations: recommLastSeen on page 95
recommSeenCount	recommSeenCount	ancillary	Analysis: AWS Recommendations: recommSeenCount on page 95
region	region	core	Analysis: AWS Recommendations: region on page 86
resourceId	resourceId	core	Analysis: AWS Recommendations: resourceId on page 86
rptHref	rptHref	core	Analysis: AWS Recommendations: rptHref on page 88
savingsEstimate	savingsEstimate	core	Analysis: AWS Recommendations: savingsEstimate on page 89

propertyName	aliasName	Core / Ancillary	Description Reference
serviceType	serviceType	core	Analysis: AWS Recommendations: serviceType on page 91 Analysis: Azure Recommendations: serviceType on page 129 Analysis: GCP Recommendations: serviceType on page 174
totalHoursRunning	totalHoursRunning	core	Analysis: AWS Recommendations: totalHoursRunning on page 88

Default Container Properties

Default Container Subscriptions Properties List

The default Container Subscriptions Properties catalog contains both core and ancillary properties which can be used by all API users (i.e. default properties are all global).

> **Note:** *Ancillary properties must be used in conjunction with at least one core property for the purpose of filtering subscription results. Refer to Ancillary Properties for details.*

propertName	aliasName	Core / Ancillary	Description Reference
auditInfo.dataCollection.auditCount	dc_auditCount	ancillary	Analysis: Kubernetes Container Recommendations: auditInfo on page 188
auditInfo.dataCollection.dateFirstAudited	dc_firstAudited	ancillary	Analysis: Kubernetes Container Recommendations: auditInfo on page 188
auditInfo.dataCollection.dateLastAudited	dc_lastAudited	ancillary	Analysis: Kubernetes Container Recommendations: auditInfo on page 188
auditInfo.workloadDataLast30.firstDate	last30_firstDate	ancillary	Analysis: Kubernetes Container Recommendations: auditInfo on page 188
auditInfo.workloadDataLast30.lastDate	last30_lastDate	ancillary	Analysis: Kubernetes Container Recommendations: auditInfo on page 188
auditInfo.workloadDataLast30.seenDays	last30_seenDays	ancillary	Analysis: Kubernetes Container Recommendations: auditInfo on page 188
auditInfo.workloadDataLast30.totalDays	last30_totalDays	ancillary	Analysis: Kubernetes Container Recommendations: auditInfo on page 188

propertName	aliasName	Core / Ancillary	Description Reference
cluster	cluster	core	Analysis: Kubernetes Container Recommendations: cluster on page 188
container	container	core	Analysis: Kubernetes Container Recommendations: container on page 188
currentCount	currentCount	core	Analysis: Kubernetes Container Recommendations: currentCount on page 189
currentCpuLimit	currentCpuLimit	core	Analysis: Kubernetes Container Recommendations: currentCpuLimit on page 191
currentCpuRequest	currentCpuRequest	core	Analysis: Kubernetes Container Recommendations: currentCpuRequest on page 192
currentMemLimit	currentMemLimit	core	Analysis: Kubernetes Container Recommendations: currentMemLimit on page 190
currentMemRequest	currentMemRequest	core	Analysis: Kubernetes Container Recommendations: recommendedMemRequest on page 192
controllerType	controllerType	core	Analysis: Kubernetes Container Recommendations: controllerType on page 191
dataQuality.completeDays	dq_completeDays	ancillary	Analysis: Kubernetes Container Recommendations: dataQuality on page 191
dataQuality.firstSeen	dq_firstSeen	ancillary	Analysis: Kubernetes Container Recommendations: dataQuality on page 191
dataQuality.lastSeen	dq_lastSeen	ancillary	Analysis: Kubernetes Container Recommendations: dataQuality on page 191
dataQuality.partialDays	dq_partialDays	ancillary	Analysis: Kubernetes Container Recommendations: dataQuality on page 191
dataQuality.workloadName	dq_workloadName	ancillary	Analysis: Kubernetes Container Recommendations: dataQuality on page 191
displayName	displayName	core	Analysis: Kubernetes Container Recommendations: displayName on page 188
entityId	entityId	core	Analysis: Kubernetes Container Recommendations: entityId on page 191
hostName	hostName	core	Analysis: Kubernetes Container Recommendations: hostName on page 188

propertName	aliasName	Core / Ancillary	Description Reference
namespace	namespace	core	Analysis: Kubernetes Container Recommendations: namespace on page 192
podService	podService	core	Analysis: Kubernetes Container Recommendations: podService on page 188
predictedUptime	predictedUptime	core	Analysis: Kubernetes Container Recommendations: predictedUptime on page 188
recommendationType	recommendationType	core	Analysis: Kubernetes Container Recommendations: recommendationType on page 190 **Note:** *To filter "Just Right" recommendations or "Not Analyzed" systems, you must use the exact recommendationType property string,* `"Not Analyzed"` *or* `"Just Right"`*, when you use the "=" or "like" operator.*
recommendedCpuLimit	recommendedCpuLimit	core	Analysis: Kubernetes Container Recommendations: recommendedCpuLimit on page 189
recommendedCpuRequest	recommendedCpuRequest	core	Analysis: Kubernetes Container Recommendations: recommendedCpuRequest on page 191
recommendedMemRequest	recommendedMemRequest	core	Analysis: Kubernetes Container Recommendations: recommendedMemRequest on page 192
recommendedMemLimit	recommendedMemLimit	core	Analysis: Kubernetes Container Recommendations: recommendedMemLimit on page 190
recommFirstSeen	recommFirstSeen	ancillary	Analysis: Kubernetes Container Recommendations: recommFirstSeen on page 191
recommLastSeen	recommLastSeen	ancillary	Analysis: Kubernetes Container Recommendations: recommLastSeen on page 188
recommSeenCount	recommSeenCount	ancillary	Analysis: Kubernetes Container Recommendations: recommSeenCount on page 189

Examples

Example: Getting a Collection of Private Cloud Subscriptions Properties

The following example shows you how to retrieve a collection of your private cloud Subscriptions properties. This example assumes that your username is "saas".

Example: Getting a Collection of Private Subscriptions Properties

Request:

```
GET /subscriptions/cloud/properties/?type=owner
```

Response:

```
{
    "propertyRef": "fe602c6f-77c2-4105-9ec4-aecce77cc104",
    "propertyName": "deferUntil",
    "aliasName": "Defer Purchase RI Until",
    "owner": "saas"
}
```

Example: Getting a Specific Container Subscriptions Property

The following example shows you how to retrieve a specific container Subscriptions property with a known property reference ID. This property must be of type "*global*" or owned by you before a successful response is returned.

Example: Getting a Specific Container Subscriptions Property

Request:

```
GET /subscriptions/containers/properties/3f6485a0-5106-4d25-a7c5-13f4270462ef
```

Response:

```
{
    "propertyRef": "3f6485a0-5106-4d25-a7c5-13f4270462ef",
    "propertyName": "controllerType",
```

```
        "aliasName": "controllerType",
        "owner": ""
    }
```

Example: Adding New Subscriptions Properties

This example shows you how to add new properties to the Cloud Subscriptions Properties catalog. Notice that the `owner` parameter is not set. If you are a non-administrative Densify user authenticating the POST request, the `owner` parameter is automatically set to your username. By having the `owner` parameter set, the property is considered *private* and can only be used by you (or any administrative user). If you are a Densify **administrative user**[1] and you do not set the `owner` parameter in the POST request, then the property is considered *global*.

If there is an error in the POST request resulting from any one of the property additions, then all the property additions in the request body are rolled back and not committed.

Example: Adding Properties to the Cloud Subscriptions Properties Catalog

Request:

```
POST /subscriptions/cloud/properties
```

Parameters:

```
[
  {
    "propertyName" :    "deferUntil",
    "aliasName"    :    "Defer Purchase RI Until"
  },
  {
    "propertyName": "deferRecommendation",
    "aliasName": "Defer Recommendation"
  }
]
```

Example: Modifying Subscriptions Properties

The example below shows you how to modify two Subscriptions properties in the `/subscriptions/cloud/properties` resource catalog.

Observe the following behavior in the PUT request example:

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

API Reference Guide for Public Cloud

- propertyRef and propertyName parameters are mandatory for identifying the property to modify.

- For both modification entries, the aliasName and owner parameters are updated.

- You must be an administrative user to call the PUT request to update the owner parameter to an empty string. This promotes the property from *private* type to *global* type.

Example: Modifying Properties from the Cloud Subscriptions Properties Catalog

Request:

```
PUT /subscriptions/cloud/properties
```

Parameters:

```
[
  {
    "propertyRef": "f8da8042-5186-45aa-8e22-bf5b589f95a6",
    "propertyName": "deferUntil",
    "aliasName": "DU",
    "owner":      ""
  },
  {
    "propertyRef": "fe602c6f-77c2-4105-9ec4-aecce77cc104",
    "propertyName": "deferRecommendation",
    "aliasName": "DR",
    "owner":      ""
  }
]
```

Example: Modifying a Subscriptions Property

This example shows you how to modify a single Subscriptions property using the /subscriptions/cloud/properties/<propertyRef> resource. This PUT request updates aliasName parameter for the specified property. The propertyName and owner parameters are mandatory in a PUT request.

Example: Modifying a Property from the Cloud Subscriptions Properties Catalog

Request:

```
PUT /subscriptions/cloud/properties/f8da8042-5186-45aa-8e22-bf5b589f95a6
```

Parameters:

```
{
    "propertyName": "deferUntil",
    "aliasName": "Defer RI purchase Until Date",
    "owner":  "saas_user"
```

```
    }
```

Example: Deleting Subscriptions Properties

This example shows you how to delete a collection of properties from the
`/subscriptions/cloud/properties` resource catalog.

Example: Deleting a Collection of Properties from the Cloud Subscriptions Properties Catalog

Request:

```
DELETE /subscriptions/cloud/properties
```

Parameters:

```
[
    {
      "propertyRef": "f8da8042-5186-45aa-8e22-bf5b589f95a6"
    },
    {
      "propertyRef": "fe602c6f-77c2-4105-9ec4-aecce77cc104"
    }
]
```

Response:

```
[
    {
      "propertyRef": "f8da8042-5186-45aa-8e22-bf5b589f95a6",
      "status": "404",
      "message": "Not found."
    },
    {
      "propertyRef": "fe602c6f-77c2-4105-9ec4-aecce77cc104",
      "message": "Delete successfully."
    }
]
```

Subscriptions: Status

Description

The `/subscriptions/<platformType>/<subscriptionId>/status` resource is used to return the status of the results posted to a webhook for a specific subscription in Densify.

Refer to *Subscriptions* (Help Topic ID 340690) for details on subscription results and defining a webhook.

Resource

```
/subscriptions/cloud/<subscriptionId>/status
/subscriptions/containers/<subscriptionId>/status
/subscriptions/<subscriptionId>/status
```

Note: *If you use this resource without the* `<platformType>` *specified (i.e. without* `cloud` *or* `containers` *specified), the behavior is exactly the same as specifying the* `cloud`*-specific resource. This behavior enables backward compatibility with scripts using the Densify API prior to release 12.1.6, where the platform-specific indicator was not available.*

Supported Operations

Table: Subscription Status Supported Operations

HTTP Method	Input	Output	Description
GET /subscriptions/	Path Parameter: ▪ platformType	▪ lastTriggered ▪ webHookStatus	Returns the last subscription results request and webhook status of the

HTTP Method	Input	Output	Description
<platformType> /< subscriptionRef >/status	▪ subscriptionRef		platform-specific subscription identified by `<subscriptionRef>`. See Example: Getting the Status of a Subscription on page 329.

Parameters

Path Parameters

Table: Subscription Status Path Parameters

Parameter Name	Type	Required (Y/N)	Description
platformType	string	Y	[cloud|containers] Specify the technology platform for the subscription resource.
subscriptionRef	string	Y	Specify the unique subscription identifier.

Response

The following is a list of possible response elements returned for the /subscriptions/*<platformType>*/*<subscriptionRef>*/status **resource.**

Table: Subscription Status Response Schema

Element	Type	Filter/Sort	Description
lastTriggered	string		The status, date, and time of the last request for subscription results: ▪ On-Demand Success–the last request was on-demand and it was successful; ▪ On-Demand Failure–the last request was on-demand and it failed to produce results; ▪ Scheduled Success–the last request was a successful scheduled subscription event posted to a webhook; ▪ Scheduled-Failure–the last request was a failed webhook post of the scheduled subscription event.
webHookStatus	string		The status, date, and time of the last subscription results pushed to the webhook location: ▪ Success–subscription results were sent to the webhook successfully; ▪ Failure–transmission of subscription results to the webhook failed.
Returned Status			
message	string		The message for the status response is returned.

Element	Type	Filter/Sort	Description
status	number		The HTTP response code of the request. Possible status values include: ▪ 200–success with request (usually with content in response body); ▪ 204–success with request, no content returned; ▪ 400–bad request (invalid parameters, logical errors); ▪ 401–authentication failed; ▪ 404–resource not found (or no privileges); ▪ 415–unsupported media type; ▪ 500–internal server error.

Examples

Example: Getting the Status of a Subscription

The following example shows you how to retrieve the status of a specific cloud subscription. The status message for the subscription will be returned if the subscription is global or owned by you.

Example: Getting a Specific Cloud Subscription Status

Request:

```
GET /subscriptions/cloud/4503e5ff-7a1e-4f4c-a106-0f31ca38dc22/status
```

Response:

```
{
    "lastTriggered": "On-Demand Success.  Thu Jan 02 16:41:52 EST 2020",
    "webhookStatus": "Failure. java.net.ConnectException: Connection refused
(Connection refused). Thu Jan 02 16:41:53 EST 2020"
}
```

Subscriptions: Suppressions

Description

The `/subscriptions/<platformType>/suppressions` resource is used to store a list of attribute tags and recommendation properties available to the Subscriptions resource for the purpose of suppressing specific sets of system recommendations from the resulting subscription data set.

This resource acts as a catalog of the suppression tags or properties to be referenced by the suppressionReferences parameter in the Subscriptions resource . Suppression entries (tags or properties) not defined in a `/subscriptions/<platformType>/suppressions` resource cannot be referenced; you must define the suppression entries before using them in a suppression condition. See *Subscriptions* (Help Topic ID 340690) for details on defining a subscription.

There is a catalog for each supported `<platformType>`, which can only be referenced by the corresponding `<platformType>` subscription. For example, a container subscription (i.e. `/subscriptions/containers`) can only reference suppressions from the Container Subscriptions Suppressions catalog (i.e. `/subscriptions/containers/suppressions`).

Suppression entries can be declared as *global* or *private* (i.e. user-specific). Global suppression entries can be used by any API enabled user, while private entries can only be used by their owners. Note that **administrative users**[1] have access to all suppression entries - global or private user-specific.

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Resource

```
/subscriptions/cloud/suppressions
/subscriptions/containers/suppressions
/subscriptions/suppressions
```

Note: *If you use this resource without the `<platformType>` specified (i.e. without `cloud` or `containers` specified), the behavior is exactly the same as specifying the `cloud`-specific resource. This behavior enables backward compatibility with scripts using the Densify API prior to release 12.1.6, where the platform-specific indicator was not available.*

Supported Operations

Table: Subscriptions Suppressions Supported Operations

HTTP Method	Input	Output	Description

GET
`/subscriptions/`
`<platformType>/suppressions`
	Path Parameter:
	▪ platformType
	
	Query String Parameter Options:
	▪ type
	▪ owner
		Collection of
		▪ suppressionRef
		▪ suppressionName
		▪ attributeName |
		  propertyName
		▪ key
		▪ technology
		▪ aliasName
		▪ owner
			Returns a list of existing suppression tags or properties from the platform-specific Subscriptions Suppressions catalog.
			▪ The type query string parameter is used to return global or private suppressions. If type is not specified, all global and only private suppressions belonging to you are returned.
			▪ **Administrative users**[1] can use the owner query string parameter to return all the private suppressions belonging to a specific user.

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
			See Example: Getting a List of Available Cloud Subscriptions Suppressions.
GET /subscriptions/ *\<platformType\>*/suppressions/ \<suppressionRef\>	Path Parameters: ◾ platformType ◾ suppressionRef	◾ suppressionRef ◾ suppressionName ◾ attributeName | propertyName ◾ key ◾ technology ◾ aliasName ◾ owner	Returns a Subscriptions suppression with unique identifier \<suppressionRef\> from a platform-specific Subscriptions Suppressions catalog. See Example: Getting a Specific Container Subscriptions Suppression.
POST /subscriptions/ *\<platformType\>*/suppressions	Path Parameter: ◾ platformType Collection of Request Body Parameters: ◾ suppressionName ◾ attributeName | propertyName ◾ key ◾ technology ◾ aliasName ◾ owner	Collection of ◾ suppressionRef ◾ attributeName | propertyName See Example: Adding New Subscription Suppressions.	Adds new suppression entries into a platform-specific Subscriptions Suppressions catalog. A suppression entry can be either a tag or a property suppression. You can specify an attributeName or a propertyName, but not both. ◾ Administrative users[1] can add global or private suppressions. ◾ Non-administrative users can add only private suppressions. ◾ suppressionName, propertyName, attributeName, and aliasName are validated for uniqueness, depending on their global or private-scoped

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
			suppression. ■ The suppressions bulk-add operation is committed as a whole unit. One suppression add error will roll back the entire bulk-add operation. ■ For property suppressions, the property platform must correspond with the suppression platform. For example, you can only add container properties into the Container Subscriptions Suppressions catalog.
PUT /subscriptions/ *<platformType>*/suppressions	Path Parameter: ■ platformType Request Body Parameters: Collection of ■ suppressionRef ■ attributeName | propertyName ■ suppressionName ■ key ■ technology ■ aliasName ■ owner	Collection of ■ suppressionRef ■ attributeName | propertyName	Deletes and replaces parameters for existing suppression entries in a platform-specific Subscriptions Suppressions catalog. You must specify all parameters (except `suppressionRef`) required for the suppression you want to update. ■ Only an **administrative user**[1] can modify the `owner` parameter to promote a private suppression to a global suppression (i.e. set `owner=""`). ■ If you are not an administrative user, you can only

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
			set the `owner` parameter to your username. ■ suppressionName, propertyName, attributeName, and aliasName are validated for uniqueness, depending on their global or private-scoped suppression. ■ propertyName is platform-validated (i.e. property and suppression platforms must be the same). ■ The multiple suppression delete-and-replace operation is committed as a whole unit; an error resulting from one of the suppression entries will roll back the entire operation. See Example: Modifying Subscription Suppressions on page 343.
`PUT /subscriptions/ suppressions/<suppressionRe f>`	Path Parameters: ■ platformType ■ suppressionRef Request Body Parameters: ■ attributeName | propertyName ■ suppressionName ■ key ■ technology ■ aliasName ■ owner	■ suppressionRef ■ attributeName | propertyName	Deletes and replaces parameters for an existing suppression entry identified by `<suppressionRef>` in a platform-specific Subscriptions Suppressions catalog. You must specify all parameters required (in the Request Body Parameters section) for the suppression you want to update, as all previous

HTTP Method	Input	Output	Description
			parameters are deleted. ■ Only an **administrative user**[1] can modify the `owner` parameter to promote a private suppression to a global suppression (i.e. set `owner=""`). ■ If you are not an administrative user, you can only set the `owner` parameter to your username. ■ suppressionName, propertyName, attributeName, and aliasName are validated for uniqueness, depending on their global or private-scoped suppression. ■ propertyName is platform-validated (i.e. property and suppression platforms must be the same). See Example: Modifying a Subscription Suppression on page 344.

`DELETE`
`/subscriptions/`
`<platformType>/suppressions`
	Path Parameter:
	■ platformType
	
	Collection of
	■ suppressionRef
		■ HTTP status of "204 No Content" if all delete operations are successful
		■ If delete errors occur, the following is
			Deletes suppressions from a platform-specific Subscriptions Suppressions catalog.
			
			■ **Administrative users**[2] can delete any global or

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.
[2]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
		returned for each delete suppression request: • suppressionRef • status • message See Example: Deleting Subscriptions Suppressions on page 344.	private suppressions. ▪ Non-administrative users can only delete their own private suppression from the catalog. ▪ Suppressions referenced by subscriptions cannot be deleted. ▪ Each suppression delete operation is independent from the other delete operations in the same request (an error with one suppression delete action does not affect the delete actions of the other suppressions in the same request).
DELETE /subscriptions/ *\<platformType\>*/ suppressions/ \<suppressionRef\>	Path Parameters: ▪ platformType ▪ suppressionRef	▪ HTTP status of "204 No Content" if delete operation is successful; ▪ HTTP status of "404 Not Found" if suppression is not found; ▪ If the suppression is referenced by a subscription, or if there are other errors, then the following is returned: • suppressionRef • suppressionName • message	Deletes a suppression with \<suppressionRef\> identifier from a platform-specific Subscriptions Suppressions catalog. ▪ Administrative users[1] can delete any global or private suppression. ▪ Non-administrative users can only delete their own private suppression from the catalog. ▪ A suppression referenced by

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
			subscriptions cannot be deleted.

Parameters

Path Parameters

Table: Subscriptions Suppressions Path Parameters

Parameter Name	Type	Required (Y/N)	Description
platformType	string	Y	[cloud|containers] Specify the technology platform for the Subscriptions suppression resource.
suppressionRef	string	Y	Specify the unique identifier for a Subscriptions suppression entry.

Query String Parameters

Table: Subscriptions Suppressions Query String Parameters

Parameter Name	Type	Required (Y/N)	Description
type	string		Specify the type of Subscriptions suppression to return: • all–Return all suppression entries: global and private user-specific. If you are not an **administrative user**[1], only private suppressions owned by you and global suppressions are returned. This is the default behavior if `type` is not specified in the request. • global–Return all global Subscriptions suppression entries. • owner–Return user-specific Subscriptions suppression entries. If you are not an administrative user, only private suppressions owned by you are returned. If you are an administrative user, all global and private suppressions are returned. A Subscriptions suppression is considered *global* if the owner parameter is not populated. Global Subscriptions suppressions can be used by all Densify API users. A Subscriptions suppression is considered *private* if the owner parameter contains a Densify username. Private Subscriptions suppressions can only be used by their owners or administrative users.

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Parameter Name	Type	Required (Y/N)	Description
owner	string		If you are an **administrative user**[1], you can specify a Densify username in conjunction with the `type=owner` query string parameter to return all of the specified user's private Subscriptions suppressions. If you are not an administrative user, you can request for your own private suppressions. If you use the `?type=owner&owner=<anotherusername>` query string option with a username other than your own, the returned response is a 400 Bad Request -"Current login user cannot query for owner" error.

Request Body Parameters

Table: Subscriptions Suppressions Request Body Parameters

Parameter Name	Type	Required (C-create/M-modify/D-delete)	Description
suppressionRef	string	M D	Specify the unique identifier for a Subscriptions suppression entry.
suppressionName	string	C M	Specify a unique name for a Subscriptions suppression entry. For *global* Subscriptions suppressions, the suppressionName must be unique within a platform-specific Subscriptions Suppressions catalog. For *private* Subscriptions suppressions, the suppressionName must be unique per owner and across all global subscription suppressions per platform-specific catalog. For example, owner A and owner B can both have a private suppression named "SuperA", as long as "SuperA" is not also a global suppression within the same platform-specific catalog.
aliasName	string		Specify an alias name for the Subscriptions attribute tag. For *global* suppressions, the aliasName must be unique within a platform-specific Subscriptions Suppressions catalog. For *private* suppressions, the aliasName must be unique per owner and across all global suppressions per platform-specific catalog. For example, owner A and owner B can both have a private suppression alias named "Joe", as long as "Joe" is also not a global suppression alias within the same platform-specific catalog.
attributeName	string	C M	Specify the attribute name for the Subscriptions suppression entry. The attribute name must exist in the Densify standard set of system attributes. Contact Support@Densify.com for a list of available system attributes.

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Parameter Name	Type	Required (C-create/M-modify/D-delete)	Description
			Use the "`Resource Tags`" attribute name along with [key](#) and [technology](#) parameters for cloud technology-specific resource attributes in the "`key:value`" form. See [Example: Adding New Subscription Suppressions](#) for "Resource Tags" usage.
			Use the "`Container Labels`" attribute name along with [key](#) and [technology](#) parameters for container-specific attributes in "`key:value`" form.
			Note that the `attributeName` must be unique in a platform-specific Subscriptions Suppressions catalog (i.e. you cannot have two suppressions with the same `attributeName` in a catalog).
key	string	CM[1]	Specify the key string required for the technology platform resource attribute.
			If the Subscriptions suppression is a reference to a resource attribute (i.e. `"attributeName": "Resource Tags"` or `"attributeName": "Container Labels"`), you need to specify both the [key](#) and [technology](#) platform for the specific resource attribute.
technology	string	CM[2]	Specify the technology platform for the resource attribute. Currently, the following technology platforms are supported:
			● AWS ● CONTAINER
			If the Subscriptions suppression is a reference to a resource attribute (i.e. `"attributeName": "Resource Tags"` or `"attributeName": "Container Labels"`), you need to specify both the [key](#) and [technology](#) platform for the specific resource attribute.
propertyName	string	C M	Specify the recommendation element for the Subscriptions suppression.
			The list of available recommendation elements can be found in the *Analysis: technology-specific Recommendations* page. For example, refer to the [Response](#) schema section of the [Analysis: AWS Recommendations on page 83](#) for a full list of AWS recommendation elements.
			The `propertyName` must be unique within a platform-specific Subscriptions Suppressions catalog.
owner	string	M[3]	When the `owner` parameter is not set, the Subscriptions suppression is considered *global*. Global Subscriptions suppressions can be used by all API users. Only **administrative**

---

[1]For resource attributes, you need to specify both `key` and `technology` parameters.
[2]For resource attributes, you need to specify both `key` and `technology` parameters.
[3]The `owner` parameter is mandatory for private Subscriptions suppressions.

Parameter Name	Type	Required (C-create/M-modify/D-delete)	Description
			users[1] can create global suppressions. When the `owner` parameter is set, the suppression is considered *private.* Private Subscriptions suppressions can only be used by their owners or administrative users. If you are an administrative user, you have the ability to assign any Densify user as the owner of the subscription suppression in a POST request. In a PUT request, administrative users can promote the suppression from private to global by setting `owner: ""`. If you are not an administrative user, you can only set the `owner` parameter to your Densify username. In a POST request, the `owner` parameter is automatically populated with your username.

Response

The following is a complete list of possible response elements that are returned for the `/subscriptions/suppressions` resource.

Table: Subscriptions Suppressions Response Schema

Element	Type	Filter/Sort	Description
suppressionRef	string		The unique referenced ID of the Densify Subscriptions suppression entry.
suppressionName	string		The Subscriptions suppression name.
aliasName	string		The Subscriptions suppression alias name.
attributeName	string		The attribute name for the Subscriptions suppression entry.
technology	string		The Subscriptions suppression's technology platform for the resource attribute.
key	string		The resource attribute key string for the suppression's associated technology platform.
propertyName	string		The Subscriptions suppression property name.
owner	string	F	The designated user/owner of this Subscriptions suppression. A Subscriptions suppression is considered *global* if the owner parameter is not populated. Global Subscriptions suppressions can be used by all Densify API users. A Subscriptions suppression is considered *private* if the owner parameter contains a Densify username. Private Subscriptions

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Element	Type	Filter/Sort	Description
			suppressions can only be used by their owners or administrative users[1].
message	string		The message for the error or status response is returned.
status	number		The HTTP response code of the request. Possible status values include: ■ 200–success with request (usually with content in response body); ■ 204–success with request, no content returned; ■ 400–bad request (invalid parameters, logical errors); ■ 401–authentication failed; ■ 404–resource not found (or no privileges); ■ 415–unsupported media type; ■ 500–internal server error.

Examples

Example: Getting a List of Available Cloud Subscriptions Suppressions

The following example shows you how to retrieve a list of suppressions available to you from the Cloud Subscriptions Suppressions catalog.

Example: Getting a List of Available Cloud Subscription Suppressions

Request:

```
GET /subscriptions/cloud/suppressions
```

Response:

```
[
    {
        "suppressionRef": "2ff4501e-df32-4f57-8a77-e539192fa043",
        "suppressionName": "Entity ID",
        "propertyName": "entityId",
        "aliasName": "supprEntityID",
        "owner": ""
    },
    {
        "suppressionRef": "38f6e37c-86f4-44a4-9741-dc9d179cbbc1",
```

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

```
        "suppressionName": "Effort Estimate",
        "propertyName": "effortEstimate",
        "aliasName": "supprEffortEstimate",
        "owner": ""
    }
    ...
]
```

Example: Getting a Specific Container Subscriptions Suppression

The following example shows you how to retrieve a specific container Subscriptions suppression with a known reference ID. This suppression must be of type "*global*" or owned by you before a successful response is returned.

Example: Getting a Specific Container Subscriptions Suppression Entry

Request:

```
GET /subscriptions/containers/suppressions/8b58927e-8f1a-4105-b8f2-5f2b0f-
d0238d
```

Response:

```
{
    "suppressionRef": "fdc363b2-523c-4bb5-bdbf-a4f4ef994487",
    "suppressionName": "RecommendedMemRequest",
    "propertyName": "recommendedMemRequest",
    "aliasName": "supprRecMemRequest",
    "owner": ""
}
```

Example: Adding New Subscription Suppressions

This example shows you how to add new suppression entries to the Cloud Subscriptions Suppressions catalog. Notice that the `owner` parameter is not set. If you are a non-administrative Densify user authenticating the POST request, the `owner` parameter is automatically set to your username. By having the `owner` parameter set, the suppression is considered *private* and can only be used by you (or any administrative user). If you are a Densify **administrative user**[1] and you do not set the `owner` parameter in the POST request, then the suppression is considered *global*.

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

If there is an error in the POST request resulting from any one of the suppression additions, then all the suppression additions in the request body are rolled back and not committed.

Example: Adding Suppressions to the Cloud Subscriptions Suppressions Catalog

Request:

```
POST /subscriptions/cloud/suppressions
```

Parameters:

```
[
    {
        "suppressionName": "Health_Check",
        "attributeName": "Health Check",
        "aliasName": "Suppr_Health_Check"
    },
    {
        "suppressionName": "AWS RTag Env Suppr",
        "attributeName": "Resource Tags",
        "key": "Environment :",
        "technology": "AWS",
        "aliasName": "Suppr_Health_Check"
    },
    {
        "suppressionName": "Recommended_Cost",
        "propertyName": "recommendedCost",
        "aliasName": "Suppr Recommended Cost"
    }
]
```

Example: Modifying Subscription Suppressions

This example shows you how to modify two subscription suppression entries in the Cloud Subscriptions Suppressions catalog. You need to specify all the request body parameters for PUT request. With the exception of the `suppressionRef` and `owner` parameters, all other parameters can be updated If there is an error in the PUT request from any one of the update entries, then all the updates are rolled back and not applied. Note that only the administrative user is eligible to make this PUT request.

Example: Modifying Suppressions from the Cloud Subscriptions Suppressions Catalog

Request:

```
PUT /subscriptions/cloud/suppressions
```

Parameters:

API Reference Guide for Public Cloud

343

```
[
  {
    "suppressionRef": "61acf182-1773-4988-b4d9-a76c866b5c68",
    "suppressionName": "Business Applications",
    "attributeName": "Business Applications",
    "aliasName": "BusinessApplicationsSuppression",
    "owner":      ""
  },
  {
    "suppressionRef": "c6da9e05-92be-4ec6-9892-7a2ed68d57f0",
    "suppressionName": "Instance Name",
    "propertyName": "name",
    "aliasName": "SuppressThisName",
    "owner":      ""
  }
]
```

Example: Modifying a Subscription Suppression

This example shows you how to modify a single cloud Subscriptions suppression using the `/subscriptions/cloud/suppressions/<suppressionRef>` resource. You need to specify all the request body parameters for a PUT request, even if you only want to modify the `suppressionName` or `aliasName` parameters. In this example, you must either be the "saas" user or the administrative user to be authorized to make this PUT request.

Example: Modifying a Suppression from the Cloud Subscriptions Suppressions Catalog

Request:

```
PUT /subscriptions/cloud/suppressions/d6472966-52bd-4231-a0ab-ea9cae2f5016
```

Parameters:

```
{
    "suppressionName": "OS Arch Suppr",
    "attributeName": "OS Architecture",
    "aliasName": "Suppress OS Architecture",
    "owner":    "saas"
}
```

Example: Deleting Subscriptions Suppressions

This example shows you how to delete a collection of suppressions from the `/subscriptions/cloud/suppressions` resource catalog. Keep in mind that you can only delete your

own private suppressions from the catalog. Only **administrative users**[1] can delete any global and any private suppressions.

Example: Deleting a Collection of Suppressions from the Cloud Subscriptions Suppressions Catalog

Request:

```
DELETE /subscriptions/cloud/suppressions
```

Parameters:

```
[
    {
        "suppressionRef": "209baebd-fc67-4902-a7a2-d9386af26a4e"
    },
    {
        "suppressionRef": "9cc94889-b15d-4050-aa8f-d9f605271e58"
    }
]
```

Response:

```
[
    {
        "suppressionRef": "9cc94889-b15d-4050-aa8f-d9f605271e58",
        "status": "404",
        "message": "Not found."
    },
    {
        "suppressionRef": "209baebd-fc67-4902-a7a2-d9386af26a4e",
        "message": "Delete successfully."
    }
]
```

Default Suppressions

Default Cloud Subscriptions Suppression List

The Cloud Subscriptions Suppressions catalog contains the following default entries:

```
[
    {
        "suppressionRef": "03922061-96eb-450b-a30f-a397a19c9a6f",
```

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

```
            "suppressionName": "Effort Estimate",
            "propertyName": "effortEstimate",
            "aliasName": "supprEffortEstimate",
            "owner": ""
        },
        {
            "suppressionRef": "29150379-9252-498d-a59d-de47d183eee8",
            "suppressionName": "Approval Type",
            "propertyName": "approvalType",
            "aliasName": "supprApprovalType",
            "owner": ""
        },
        {
            "suppressionRef": "61acf182-1773-4988-b4d9-a76c866b5c68",
            "suppressionName": "Business Applications",
            "attributeName": "Business Applications",
            "aliasName": "supprBusinessApplications",
            "owner": ""
        },
        {
            "suppressionRef": "93dba027-8391-4303-b413-7c4cb41ec5e9",
            "suppressionName": "Suppress Region",
            "propertyName": "region",
            "aliasName": "supprRegion",
            "owner": ""
        },
        {
            "suppressionRef": "a6827ae4-fa2b-405e-a564-d70f2dad45c2",
            "suppressionName": "recommendationType",
            "propertyName": "recommendationType",
            "aliasName": "supprRecommendationType",
            "owner": ""
        }
    ]
```

Default Container Subscriptions Suppression List

The Container Subscriptions Suppressions catalog contains the following default entries:

```
[
    {
        "suppressionRef": "fdc363b2-523c-4bb5-bdbf-a4f4ef994487",
        "suppressionName": "RecommendedMemRequest",
        "propertyName": "recommendedMemRequest",
        "aliasName": "supprRecMemRequest",
        "owner": ""
    }
]
```

API Reference Guide for Public Cloud

Subscriptions: Tags

Description

The `/subscriptions/<platformType>/tags` resource is used to store a list of system attributes available to the [Subscriptions](#) resource for the purpose of filtering systems from the resulting subscription data set.

This resource acts as a catalog of system attributes to be referenced by the [tagReferences](#) parameter in a subscription. System attributes that are not defined in a `/subscriptions/<platformType>/tags` resource, cannot be referenced by a Subscriptions tag filter condition. See *Subscriptions* (Help Topic ID 340690) for details on defining a subscription and using tag filter conditions.

The SQL to filter results employs a "STRING" comparison and not a numeric comparison on tags so you may observe inconsistent results. Tags can only be lexicographically compared.

There is a catalog for each supported `<platformType>`, which can only be referenced by the corresponding `<platformType>` subscription. For example, a container subscription (i.e. `/subscriptions/containers`) can only reference tags from the Container Subscriptions Tags catalog (i.e. `/subscriptions/containers/tags`). Densify provides an initial, default set of system attributes for each `<platformType>` catalog. You can extend these default sets by adding additional system attributes, or you can customize the sets by modifying or deleting existing attributes. Only Densify standard attributes or technology-specific attributes (e.g. AWS, Containers) can be added to the platform-specific Subscriptions Tags catalogs. Contact Support@Densify.com for a complete list of available Densify standard attributes. See [Default Cloud Attribute Tags on page 354](#) or [Default Container Attribute Tags](#).

Attributes in the platform-specific Subscriptions Tags catalog can be declared as *global* or *private* (i.e. user-specific). Global attribute tags can be used by any API enabled user, whereas private attributes

can only be used by their owners. Note that **administrative users**[1] have access to both global and private attributes for all users.

Resource

```
/subscriptions/cloud/tags
/subscriptions/containers/tags
/subscriptions/tags
```

> **Note:** If you use this resource without the `<platformType>` specified (i.e. without `cloud` or `containers` specified), the behavior is exactly the same as specifying the `cloud`-specific resource. This behavior enables backward compatibility with scripts using the Densify API prior to release 12.1.6, where the platform-specific indicator was not available.

Supported Operations

Table: Subscriptions Tags Supported Operations

HTTP Method	Input	Output	Description
GET /subscriptions/ `<platformType>` /tags	Path Parameter: ▪ platformType Query String Parameter Options: ▪ type ▪ owner	Collection of ▪ tagRef ▪ tagName ▪ aliasName ▪ attributeName ▪ technology ▪ key ▪ owner	Returns a list of existing attributes from the platform-specific Subscriptions Tags catalog. ▪ The type query string parameter is used to return global or private attribute tags. If type is not specified, all global and only private attribute tags belonging to you are returned. ▪ Administrative users[2] can use the owner query string parameter to return all the private properties belonging to a specific user. See Example: Getting a Collection of Private Cloud Subscriptions Tags on page 356. See Example: Getting a Collection of Container Subscriptions Tags on page 356.
GET /subscriptions/ `<platformType>` /tags/`<tagRef>`	Path Parameters: ▪ platformType ▪ tagRef	▪ tagRef ▪ tagName ▪ aliasName ▪ attributeName ▪ technology	Returns an attribute tag with unique identifier `<tagRef>` from a platform-specific Subscriptions Tags catalog. See Example: Getting a Specific Cloud Subscription Attribute Tag on page 357.

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.
[2]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
		• key • owner	
POST /subscriptions/ *<platformType>* /tags	Path Parameter: • platformType Request Body Parameters: Collection of • tagName • attributeName • aliasName • key • technology • owner	Collection of • tagRef • tagName	Adds new attributes into a platform-specific Subscriptions Tags catalog. • **Administrative users**[1] can add global or private attribute tags. • Non-administrative users can only add private tags. • attributeName is validated for uniqueness within the catalog (i.e. you cannot define two tags with the same `attributeName`). • tagName and aliasName are validated for uniqueness, depending on the private or global scope of the tag. • The attribute tags bulk-add operation is committed as a whole unit; an error resulting from adding one of the tag entries will roll back the entire operation. See .
PUT /subscriptions/ *<platformType>* /tags	Path Parameter: • platformType Request Body Parameters: Collection of • tagRef • tagName • attributeName • aliasName • key • technology • owner	Collection of • tagRef • tagName • message	Replaces parameters from existing attribute tags in a platform-specific Subscriptions Tags catalog. You must specify all parameters required for the tag you want to update, as all previous parameters are deleted (except `<tagRef>` since it is used to identify the attribute tag to update). • Only an **administrative user**[2] can modify the `owner` parameter to promote a private tag to a global tag (i.e. set `owner=""`). • If you are not an administrative user, you can only set the `owner` parameter to your username. • attributeName, tagName, and aliasName are validated for uniqueness, depending on the private or global scope of the tag. • This bulk-edit operation is committed as a whole unit; an error resulting from one of the edits will roll back the entire bulk edit operation. See .

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.
[2]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

HTTP Method	Input	Output	Description
PUT /subscriptions/ <platformType> /tags/<tagRef>	Path Parameters: • platformType • tagRef Request Body Parameters: • tagName • attributeName • aliasName • key • technology • owner	• tagRef • tagName • message	Replaces parameters for an existing tag, identified by <tagRef>, in a platform-specific Subscriptions Tags catalog. • Only an administrative user[1] can modify the owner parameter to change a private tag to a global tag (i.e. set owner=""). • If you are not an administrative user, you can only set the owner parameter to your username. • attributeName, tagName, and aliasName are validated for uniqueness, depending on the private or global scope of the tag. See Example: Modifying a Technology Subscription Attribute Tag on page 360.
DELETE /subscriptions/ <platformType> /tags	Path Parameter: • platformType Collection of Request Body Parameter: • tagRef	• HTTP status of "204 No Content" if all delete operations are successful • If delete errors occur, then the following is returned for each tag delete request: • tagRef • status • message	Deletes attribute tags from a platform-specific Subscriptions Tags catalog. • Administrative users[2] can delete any global and any private attribute tags. • Non-administrative users can only delete their own private tags. • Tags referenced in filtering conditions by subscriptions cannot be deleted. • Each tag delete in the bulk-delete operation is an independent action: an error with one tag delete action does not affect the delete actions of the other tags in the request body parameter. See Example: Deleting Subscription Attribute Tags on page 360
DELETE /subscriptions/ <platformType> /tags/<tagRef>	Path Parameters: • platformType • tagRef	• HTTP status of "204 No Content" if delete operation is successful • HTTP status of "404 Not Found" if tag is not found • If the tag is referenced by a subscription or if there are other errors, then the following is	Deletes attribute tag with <tagRef> identifier from a platform-specific Subscriptions Tags catalog. • Administrative users[3] can delete any global and any private attribute tags. • Non-administrative users can only delete their own private tags. • Tags referenced in filtering conditions by subscriptions cannot be deleted. See Example: Deleting a Single Subscription Attribute Tag

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.
[2]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.
[3]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

API Reference Guide for Public Cloud

HTTP Method	Input	Output	Description
		returned: • tagRef • message • status	

Parameters

Path Parameters

Table: Subscriptions Tags Path Parameters

Parameter Name	Type	Required (Y/N)	Description
platformType	string	Y	[cloud|containers] Specify the technology platform for the Subscriptions attribute tag resource.
tagRef	string	Y	Specify the unique identifier for a Subscriptions attribute tag entry.

Query String Parameters

Table: Subscriptions Tags Query String Parameters

Parameter Name	Type	Required (Y/N)	Description
type	string		Specify the type of Subscriptions attribute tag to return: • all–Return all attribute tags: global and private user-specific. If you are not an administrative user, only private attribute tags owned by you and global attribute tags are returned. This is the default behavior if `type` is not specified in the request. • global–Return all global Subscriptions attribute tags. • owner–Return user-specific Subscriptions attribute tags. If you are not an administrative user, only private attribute tags owned by you are returned. If you are an administrative user, all global and private attribute tags are returned. A Subscriptions attribute tag is considered *global* if the owner parameter is not populated. Global Subscriptions tags can be used by all Densify API users. A Subscriptions tag is considered *private* if the owner parameter contains a Densify username. Private Subscriptions tags can only be used by their owners or administrative users.

Parameter Name	Type	Required (Y/N)	Description
owner	string		If you are an **administrative user**[1], you can specify a Densify username in conjunction with the `type=owner` query string parameter to return all of the specified user's private Subscriptions attribute tags.
			If you are not an administrative user, you can request for only your own private attribute tags. If you use the `?type=owner&owner=<anotherusername>` query string option with a username other than your own, the returned response is a 400 Bad Request - "Current login user cannot query for owner" error.

Request Body Parameters

Table: Subscriptions Tags Request Body Parameters

Parameter Name	Type	Required (C-create/M-modify/D-delete)	Description
tagRef	string	M D	Specify the unique identifier for a Subscriptions attribute tag entry.
tagName	string	C M	Specify a unique name for a Subscriptions attribute tag entry.
			For *global* Subscriptions attribute tags, the tagName must be unique within a platform-specific Subscriptions Tags catalog. For *private* Subscriptions attribute tags, the tagNames must be unique per owner and across all global Subscriptions attribute tags per platform-specific catalog. For example, owner A and owner B can both have a private attribute tag named "Bob", as long as "Bob" is not also a global attribute tag within the same platform-specific catalog.
attributeName	string	C M	Specify the attribute name for the Subscriptions attribute tag.
			The attribute name must exist in the Densify standard set of system attributes. Contact Support@Densify.com for a list of available system attributes.
			Use the "`Resource Tags`" attribute name along with key and technology parameters for cloud technology-specific resource attributes in the "`key:value`" form. See Example: Adding New Subscription Attribute Tags for "Resource Tags" usage.
			Use the "`Container Labels`" attribute name along with key and technology parameters for container-specific attributes in "`key:value`" form. See Default Container Attribute Tags for an example of "Container Labels" usage.
aliasName	string		Specify an alias name for the Subscriptions attribute tag.
			For *global* attribute tags, the aliasName must be unique system-

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Parameter Name	Type	Required (C-create/M-modify/D-delete)	Description
			wide. For *private* attribute tags, the aliasName must be unique per owner and across all global Subscriptions attribute tags. For example, owner A and owner B can both have a private attribute tag alias named "Jane", as long as "Jane" is also not a global attribute tag alias.
key	string	CM[1]	Specify the key string required for the technology platform resource attribute. If the Subscriptions tag is a reference to a resource attribute (i.e. `"attributeName": "Resource Tags"` or `"attributeName": "Container Labels"`), you need to specify both the key and technology elements for the specific resource attribute.
technology	string	CM[2]	Specify the technology platform for the resource attribute. Currently, the following technology platforms are supported: • AWS • CONTAINER If the Subscriptions tag is a reference to a resource attribute (i.e. `"attributeName": "Resource Tags"` or `"attributeName": "Container Labels"`), you need to specify both the key and technology platform for the specific resource attribute.
owner	string	M[3]	When the `owner` parameter is not set, the Subscriptions tag is considered *global*. Global Subscriptions tags can be used by all API users, but can only be created by **administrative users**[4]. When the `owner` parameter is set, the tag is considered *private*. Private Subscriptions tags can only be used by their owners or administrative users. If you are an administrative user, you have the ability to assign any Densify user as the owner of the Subscriptions tag in a POST request. In a PUT request, administrative users can promote the tag from private to global by setting `owner: ""`. If you are not an administrative user, you can only set the `owner` parameter to your Densify username. In a POST request, the `owner` parameter is automatically populated with your username.

[1]For resource attributes, you need to specify both `key` and `technology` parameters.

[2]For resource attributes, you need to specify both `key` and `technology` parameters.

[3]The owner parameter is mandatory for private Subscriptions tags.

[4]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Response

The following is a complete list of possible response elements that are returned for the `/subscriptions/<platformType>/tags` resource.

Table: Subscriptions Tags Response Schema

Element	Type	Filter/Sort	Description
tagRef	string		The unique referenced ID of the Densify Subscriptions attribute tag.
tagName	string		The Subscriptions attribute tag name.
attributeName	string		The attribute name for the Subscriptions attribute tag entry.
aliasName	string		The Subscriptions attribute tag alias name.
technology	string		The Subscriptions attribute tag technology platform.
key	string		The resource attribute key string for the associated technology platform.
owner	string	F	The designated owner of this Subscriptions attribute tag. A Subscriptions attribute tag is considered *global* if this parameter is not populated. A Subscriptions attribute tag is considered *private* if the owner parameter contains a Densify username.
message	string		The message for the status response.
status	number		The HTTP response code of the request. Possible status values include: • 200–success with request (usually with content in response body); • 204–success with request, no content returned; • 400–bad request (invalid parameters, logical errors); • 401–authentication failed; • 404–resource not found (or no privileges); • 415–unsupported media type; • 500–internal server error.

Default Cloud Attribute Tags

Default Cloud Attribute Tags List

The Cloud Subscriptions Tags catalog contains the following default attributes:

tagName	aliasName	attributeName	technology	key
Availability Zone	availabilityZone	Availability Zone		
AWS Environment	awsEnvironment	Resource Tags	AWS	Env :
AWS Inventory Code	inventoryCode	Resource Tags	AWS	Inventory Code :
AWS Name	awsName	Resource Tags	AWS	Name :
AWS OS	awsOS	Resource Tags	AWS	OS :

tagName	aliasName	attributeName	technology	key
AWS Owner	awsOwner	Resource Tags	AWS	Owner :
Business Applications	businessApplications	Business Applications		
CPU Util (%)	cpuUtil%	CPU Util (%)		
Disk IO (Bytes)	diskIOBytes	Disk IO (Bytes)		
Disk IO (Ops)	diskIOOps	Disk IO (Ops)		
Department	department	Department		
License Group	licenseGroup	License Group		
License Model	licenseModel	License Model		
Mem Util (%)	memUtil%	Mem Util (%)		
Network IO (Bytes)	networkIOBytes	Network IO (Bytes)		
Observed Uptime	observedUptime	Observed Uptime		
OS Architecture	architecture	OS Architecture		
Predicted Uptime	predictedUptime	Predicted Uptime		
Product Code	productCode	Resource Tags	AWS	Product Code :
Resource Tags	resourceTags	Resource Tags		
RDS Multi-AZ Deployment	RDSmultiAZ	RDS Multi-AZ Deployment		
Virtual Cluster	virtualCluster	Virtual Cluster		
Virtual Datacenter	virtualDatacenter	Virtual Datacenter		
Virtual Domain	awsAccount	Virtual Domain		
VPC ID	vpcID	VPC ID		

Default Container Attribute Tags

Default Container Attribute Tags List

The Container Subscriptions Tags catalog contains the following default attributes:

tagName	aliasName	attributeName	technology	key
ContainerPodName	containerPodName	Container Labels	CONTAINER	pod_name :
My Container Labels	containerLabels	Container Labels		
Resource Tags	resourceTags	Resource Tags		
Virtual Datacenter	virtualDatacenter	Virtual Datacenter		
Virtual Domain	virtualDomain	Virtual Domain		

Examples

Example: Getting a Collection of Private Cloud Subscriptions Tags

The following example shows you how to retrieve a collection of your private Subscriptions cloud attributes. The example assumes that your username is "saas".

Example: Getting a Collection of Private Cloud Subscriptions Tags

Request:

```
GET /subscriptions/cloud/tags?type=owner
```

Response:

```
[
    {
    "tagRef": "548b6567-e523-45a4-b779-3d2f24f8d64d",
    "tagName": "AWS Tag 1",
    "aliasName": "AWS RTag Location",
    "attributeName": "Resource Tags",
    "technology": "AWS",
    "key": "aws:location",
    "owner": "saas"
    },
    {
    "tagRef": "b1dc5d9e-3646-4527-b5db-73d31bfb66bf",
    "tagName": "AWS Tag 2",
    "aliasName": "AWS RTag Application",
    "attributeName": "Resource Tags",
    "technology": "AWS",
    "key": "aws:application",
    "owner": "saas"
    }
]
```

Example: Getting a Collection of Container Subscriptions Tags

The following example shows you how to retrieve a collection of Subscriptions container attributes available to you.

Example: Getting a Collection of Container Subscriptions Tags

Request:

```
GET /subscriptions/containers/tags
```

Response:

```
[
    {
        "tagRef": "17d17ac1-52c0-4e4a-a34d-0dd787c6b5d2",
        "tagName": "Virtual Datacenter",
        "aliasName": "virtualDatacenter",
        "attributeName": "Virtual Datacenter",
        "owner": ""
    },
    {
        "tagRef": "5cfb1b8d-b462-40f3-9783-624b3c2127d5",
        "tagName": "Resource Tags",
        "aliasName": "resourceTags",
        "attributeName": "Resource Tags",
        "owner": ""
    },
    {
        "tagRef": "acc99dbc-d21c-4c7d-a160-c62e8295dae1",
        "tagName": "My Container Labels",
        "aliasName": "containerLabels",
        "attributeName": "Container Labels",
        "owner": ""
    },
    {
        "tagRef": "cf424bf2-79ac-4031-aa68-e941327609c2",
        "tagName": "Virtual Domain",
        "aliasName": "virtualDomain",
        "attributeName": "Virtual Domain",
        "owner": ""
    },
    {
        "tagRef": "e61b3263-b715-4bc8-8310-a2bb7145f9e1",
        "tagName": "ContainerPodName",
        "aliasName": "containerPodName",
        "attributeName": "Container Labels",
        "technology": "CONTAINER",
        "key": "pod_name : ",
        "owner": ""
    }
]
```

Example: Getting a Specific Cloud Subscription Attribute Tag

The following example shows you how to retrieve a specific cloud Subscriptions attribute tag with a known tag ID. This attribute tag must be of type "*global*" or owned by you before a successful response is returned.

Example: Getting a Specific Cloud Subscription Attribute Tag

Request:

```
GET /subscriptions/cloud/tags/0e9751c4-2c5c-45a1-ab92-e34c2fa395c8
```

Response:

```
[
    {
    "tagRef": "0e9751c4-2c5c-45a1-ab92-e34c2fa395c8",
    "tagName": "Virtual Domain",
    "aliasName": "AWS Account",
    "attributeName": "Virtual Domain",
    "owner": ""
    }
]
```

Example: Adding New Subscription Attribute Tags

This example shows you how to add two new technology-specific attributes to the Cloud Subscriptions Tags catalog. Notice that the `owner` parameter is not set. If you are a non-administrative Densify user authenticating the POST request, the `owner` parameter is automatically set to your username. By having the `owner` parameter set, the attribute entry is considered *private* and can only be used by you (or any administrative user). If you are a Densify **administrative user**[1] and you do not set the `owner` parameter in the POST request, then `owner` is not set, making the attribute entry *global*.

If there is an error in the POST request from any one of the attribute additions, then all the new attribute additions are rolled back.

Example: Adding Technology-Specific Attributes to the Cloud Subscriptions Tags Catalog

Request:

```
POST /subscriptions/cloud/tags
```

Parameters:

```
[
    {
    "tagName":      "AWS RTag Env",
    "attributeName": "Resource Tags",
    "aliasName":     "AWS Environment",
    "key":           "Environment :",
```

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

API Reference Guide for Public Cloud

```
    "technology":     "AWS"
    },
    {
    "tagName":      "AWS RTag LOB",
    "attributeName": "Resource Tags",
    "aliasName":      "AWS Inventory Code",
    "key":            "LOB :",
    "technology":     "AWS"
    }
]
```

Example: Modifying Subscription Attribute Tags

This example shows you how to modify two Subscriptions attribute tags in the Cloud Subscriptions Tags catalog.

Observe the following behavior in the PUT request example:

- Even if you want to modify one parameter (e.g. `aliasName`) you still have to specify all parameters for the existing tag. A PUT request is essentially a DELETE and REPLACE operation.

- This example assumes that you are an **administrative user**[1] to update a global tag (i.e. the first tag has `"owner": ""` , implying that it is a *global* tag).

- `tagName` and `aliasName` are validated for uniqueness, depending on the private or global scope of the tags.

- If there is an error in the PUT request from any one of the update entries, then all the updates are rolled back and not applied.

Example: Modifying an Attribute Tag from the Cloud Subscriptions Tags Catalog

Request:

```
PUT /subscriptions/cloud/tags
```

Parameters:

```
[
    {
    "tagRef": "0e9751c4-2c5c-45a1-ab92-e34c2fa395c8",
    "tagName": "Virtual Domain",
    "aliasName": "My Virtual Domain",
    "attributeName": "Virtual Domain",
    "owner":     ""
    },
```

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

```
          {
          "tagRef": "7e6c4c8a-82d7-44ea-b254-56c12a8af449",
          "tagName": "Notes",
          "aliasName": "SaaS Notes",
          "attributeName": "Notes",
          "owner": "saas"
          }
      ]
```

Example: Modifying a Technology Subscription Attribute Tag

This example shows you how to modify a single technology-specific Subscriptions attribute tag using the `/subscriptions/cloud/tags/<tagRef>` resource. This PUT request uses administrative privileges to update `tagName` and `aliasName` parameters for the specified tag.

Example: Modifying a Technology Attribute Tag from the Cloud Subscriptions Tags Catalog

Request:

```
PUT /subscriptions/cloud/tags/e4558ad1-0a39-428a-a24e-8e958debfd60
```

Parameters:

```
{
    "tagName": "AWS Environment",
    "aliasName": "awsEnvironment",
    "attributeName": "Resource Tags",
    "technology": "AWS",
    "key": "Env : ",
    "owner": ""
}
```

Example: Deleting Subscription Attribute Tags

This example shows you how to delete a collection of attribute tags from the `/subscriptions/cloud/tags` resource catalog. Keep in mind that you can only delete your own private attribute tags from the catalog. Only **administrative users**[1] can delete any global and any private attribute tags.

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

API Reference Guide for Public Cloud

Example: Deleting a Collection of Attribute Tags from the Cloud Subscriptions Tags Catalog

Request:

```
DELETE /subscriptions/cloud/tags
```

Parameters:

```
[
  {
    "tagRef":     "28eaf716-e66a-4518-a10a-542641b5b155"
  },
  {
    "tagRef": "f9ce84e0-27f2-4c66-96cf-874800c84b42"
  }
]
```

Example: Deleting a Single Subscription Attribute Tag

This example shows you how to delete a single attribute tags from the Cloud Subscriptions Tags catalog. Keep in mind that you can only delete your own private attribute tags from the catalog. Only **administrative users**[1] can delete any global and any private attribute tags.

Example: Deleting a Single Attribute Tag from the Cloud Subscriptions Tags Catalog

Request:

```
DELETE /subscriptions/cloud/tags/28eaf716-e66a-4518-a10a-542641b5b155
```

---

[1]An administrative user is a Densify user in the Administrator user group or with the Analysis Admin role.

Index

F

G

H

I

Densify

200 – 120 East Beaver Creek Rd

Richmond Hill, Ontario

Canada, L4B 4V1

www.Densify.com

Telephone: +1.905.731.0090

Telephone (toll free): +1.866.731.0090

Fax: +1. 289.650.1811