

# Azure Purview REST API

[aka.ms/purviewapi](https://aka.ms/purviewapi)



## Resources

- Blog Post (<https://taygan.co>)
- Data Governance with Azure Purview (<https://aka.ms/purview/L100>)
- Azure Purview Lab Exercises (<https://aka.ms/purviewlab>)
- Azure Purview Best Practices (<https://aka.ms/purview/bb>)
- Azure Purview CLI (<https://aka.ms/purviewcli>)
- Azure Purview API (<https://aka.ms/purviewapi>)



## High-Level Overview

## Azure Purview

### Interface



Purview Studio

Classifications | Glossary | Insights | Lineage | Search | Sensitivity Labels | Sources



Custom Integration

### Platform (REST API Endpoints)



catalog

Apache Atlas 2.0

Entity | Glossary | Lineage | Relationship | Types | Search



scan

Collections | Sources | Scans | Scan Rule Sets



guardian

Data Insights

Assets | Scans | Glossary | Classification  
Sensitivity Labels | File Extensions



**Apache Atlas**



## Apache Atlas

*"Apache Atlas provides open metadata management and governance capabilities for organizations to build a catalog of their data assets, classify and govern these assets and provide collaboration capabilities around these data assets for data scientists, analysts and the data governance team."*

## Apache Atlas Milestones

- 2015-01-28 [Hortonworks Establishes Data Governance Initiative](#)
- 2015-05-06 [Apache Atlas submitted to the Apache Software Foundation](#)
- 2017-06-21 [Apache Atlas graduates to a Top-Level Project](#)
- 2018-06-02 [Apache Atlas 1.0.0](#)
- 2019-05-04 [Apache Atlas 2.0.0](#)
- 2020-07-15 [Apache Atlas 2.1.0 \(Latest\)](#)

# Apache Atlas REST API

## Types



A definition of how a particular type of metadata object is stored and accessed.

## Entity



An instance of an entity “type” (i.e. entityDef) is an entity.

## Glossary



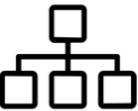
A hierarchical set of business terms that represents your business domain.

## Relationship



Relationships between entities.

## Lineage



Return lineage information about an entity.

## Discovery\*



Data discovery using DSL or full text search.

(\*) Note: Azure Purview does not use default Apache Atlas “Discovery” but rather its own “Advanced Search”.



## Apache Atlas Type System

*"Atlas allows users to define a model for the metadata objects they want to manage. The model is composed of definitions called 'types'. Instances of 'types' called 'entities' represent the actual metadata objects that are managed. The Type System is a component that allows users to define and manage the types and entities. All metadata objects managed by Atlas out of the box (like Hive tables, for e.g.) are modelled using types and represented as entities. To store new types of metadata in Atlas, one needs to understand the concepts of the type system component."*

# Apache Atlas Type Definitions

<h3>classificationDefs</h3>  <p>Classifications can be applied to entities to help describe the content (e.g. <a href="#">Email Address</a>).</p>	<h3>entityDefs</h3>  <p>Captures details of an entity-type (e.g. <a href="#">azure_sql_table</a>).</p>	<h3>enumDefs</h3>  <p>A set of named values (e.g. <a href="#">blob_access_tier</a> - Hot, Cool, Archive, or Unknown).</p>
<h3>relationshipDefs</h3>  <p>Relationships have 2 ends, each of which specify cardinality (e.g. <a href="#">azure_sql_table_columns</a>).</p>	<h3>structDefs</h3>  <p>A struct is a group of attributes (variables) under one name (e.g. <a href="#">blob_soft_deleted_state</a>).</p>	<h3>businessMetadataDefs*</h3>  <p>Business attributes that can be added to existing entity types.</p>

Note #1: Type definitions capture details of a metatype (e.g. Classification, Entity, Relationship, Struct).

Note #2: businessMetadataDefs are not currently used in Azure Purview (\*).

# AtlasEnumDef

REST API	[GET] /v2/types/enumdef/name/{name}	CLASS	EXAMPLE RESPONSE	RESPONSE MAPPED TO CLASS PROPERTIES
PURVIEW CLI	pv types readEnumdef --name 'blob_access_tier'	<b>AtlasEnumDef</b> <pre>defaultValue: string elementDefs: AtlasEnumElementDef[]</pre>	<pre>1  { 2    "category": "ENUM", 3    "guid": "e4d30715-5cf8-4a44-a7e1-1c71ca7b56a1", 4    "createdBy": "admin", 5    "updatedBy": "admin", 6    "createTime": 1614716401911, 7    "updateTime": 1614716401911, 8    "version": 1, 9    "name": "blob_access_tier", 10   "description": "blob_access_tier", 11   "typeVersion": "1.0", 12   "serviceType": "Azure Blob Storage", 13   "lastModifiedTS": "1", 14   "elementDefs": [ 15     { 16       "value": "Unknown", 17       "ordinal": 0 18     }, 19     { 20       "value": "Hot", 21       "ordinal": 1 22     }, 23     { 24       "value": "Cool", 25       "ordinal": 2 26     }, 27     { 28       "value": "Archive", 29       "ordinal": 3 30     } 31   ] 32 }</pre>	<b>AtlasEnumDef</b> <pre>defaultValue: None elementDefs: ["Unknown", "Hot", "Cool", "Archive"]</pre>
AtlasBaseTypeDef	<pre>category: TypeCategory createTime: number createdBy: string dateFormatter: DateFormat description: string guid: string name: string options: map of string serviceType: string typeVersion: string updateTime: number updatedBy: string version: number</pre>	<b>AtlasBaseTypeDef</b> <pre>category: "ENUM" createTime: 1614716401911 createdBy: "admin" dateFormatter: None description: "blob_access_tier" guid: "e4d30715-5cf8-4a44-a7e1-1c71ca7b56a1" name: "blob_access_tier" options: None serviceType: "Azure Blob Storage" typeVersion: "1.0" updateTime: 1614716401911 updatedBy: "admin" version: 1</pre>		

# AtlasStructDef

REST API

[GET] /v2/types/structdef/name/{name}

PURVIEW CLI

pv types readStructdef --name 'blob\_soft\_deleted\_state'

## CLASS

### AtlasStructDef

attributeDefs: AtlasAttributeDef[]



### AtlasBaseTypeDef

category: TypeCategory  
createTime: number  
createdBy: string  
dateFormatter: DateFormat  
description: string  
guid: string  
name: string  
options: map of string  
serviceType: string  
typeVersion: string  
updateTime: number  
updatedBy: string  
version: number

## EXAMPLE RESPONSE

```
1  {
2      "category": "STRUCT",
3      "guid": "60e0b83f-4484-4baa-aacf-b69f44cf8db8",
4      "createdBy": "admin",
5      "updatedBy": "admin",
6      "createTime": 1614716402556,
7      "updateTime": 1614716402556,
8      "version": 1,
9      "name": "blob_soft_deleted_state",
10     "description": "blob_soft_deleted_state",
11     "typeVersion": "1.0",
12     "serviceType": "Azure Blob Storage",
13     "lastModifiedTS": "1",
14     "attributeDefs": [
15         {
16             "name": "deleted",
17             "typeName": "boolean",
18             "isOptional": true,
19             "cardinality": "SINGLE",
20             "valuesMinCount": 0,
21             "valuesMaxCount": 1,
22             "isUnique": false,
23             "isIndexable": false,
24             "includeInNotification": false
25         },
26     ],
27     "options": null,
28     "typeVersion": "1.0"
29 }
```

## RESPONSE MAPPED TO CLASS PROPERTIES

### AtlasStructDef

attributeDefs: ["deleted", "deletedTime", "remainingRetentionDays"]



### AtlasBaseTypeDef

category: "STRUCT"  
createTime: 1614716402556  
createdBy: "admin"  
dateFormatter: None  
description: "blob\_soft\_deleted\_state"  
guid: "60e0b83f-4484-4baa-aacf-b69f44cf8db8"  
name: "blob\_soft\_deleted\_state"  
options: None  
serviceType: "Azure Blob Storage"  
typeVersion: "1.0"  
updateTime: 1614716402556  
updatedBy: "admin"  
version: 1

# AtlasClassificationDef

## REST API

[GET] /v2/types/classificationdef/name/{name}

## PURVIEW CLI

pv types readClassificationdef --name 'MICROSOFT.PERSONAL.EMAIL'

### CLASS

#### AtlasClassificationDef

entityTypes: string[]  
subTypes: string[]  
superTypes: string[]

#### AtlasStructDef

attributeDefs: AtlasAttributeDef[]

#### AtlasBaseTypeDef

category: TypeCategory  
createTime: number  
createdBy: string  
dateFormatter: DateFormat  
description: string  
guid: string  
name: string  
options: map of string  
serviceType: string  
typeVersion: string  
updateTime: number  
updatedBy: string  
version: number

### EXAMPLE RESPONSE

```
1  {
2    "attributeDefs": [],
3    "category": "CLASSIFICATION",
4    "createTime": 1615787943724,
5    "createdBy": "admin",
6    "description": "Email Address",
7    "entityTypes": [],
8    "guid": "1dab37e2-213f-a88d-7e13-064af9561e51",
9    "name": "MICROSOFT.PERSONAL.EMAIL",
10   "options": {
11     "displayName": "Email Address"
12   },
13   "subTypes": [],
14   "superTypes": [],
15   "typeVersion": "1.0",
16   "updateTime": 1615787943724,
17   "updatedBy": "admin",
18   "version": 1
19 }
```

### RESPONSE MAPPED TO CLASS PROPERTIES

#### AtlasClassificationDef

entityTypes: []  
subTypes: []  
superTypes: []

#### AtlasStructDef

attributeDefs: []

#### AtlasBaseTypeDef

category: "CLASSIFICATION"  
createTime: 1615787943724  
createdBy: "admin"  
dateFormatter: None  
description: "Email Address"  
guid: "1dab37e2-213f-a88d-7e13-064af9561e51"  
name: "MICROSOFT.PERSONAL.EMAIL"  
options: {"displayName": "Email Address"}  
serviceType: None  
typeVersion: "1.0"  
updateTime: 1615787943724  
updatedBy: "admin"  
version: 1



## Entities

*"An 'entity' in Atlas is a specific value or instance of an Entity 'type' and thus represents a specific metadata object in the real world. Referring back to our analogy of Object Oriented Programming languages, an 'instance' is an 'Object' of a certain 'Class'."*

# AtlasEntityDef

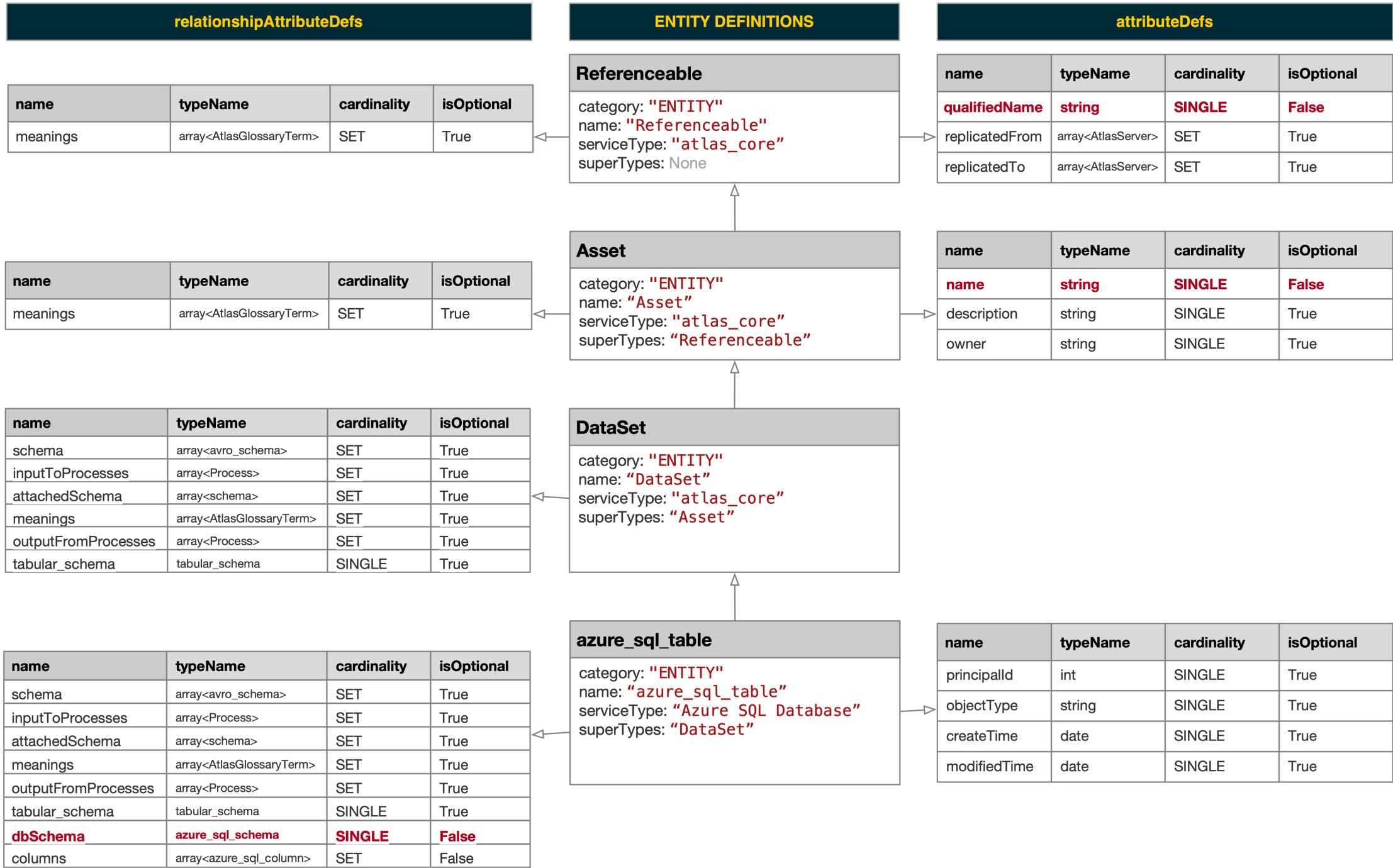
**REST API**

[GET] /v2/types/entitydef/name/{name}

**PURVIEW CLI**

pv types readEntitydef --name 'azure\_sql\_table'

CLASS	EXAMPLE RESPONSE	RESPONSE MAPPED TO CLASS PROPERTIES
<b>AtlasEntityDef</b>  businessAttributeDefs: AtlasAttributeDef[] relationshipAttributeDefs: AtlasRelationshipAttributeDef[] subTypes: string[] superTypes: string[]	<pre>1 { 2   "category": "ENTITY", 3   "guid": "4fab04f5-9d26-42dd-9ed5-b64db9174970", 4   "createdBy": "admin", 5   "updatedBy": "ServiceAdmin", 6   "createTime": 1614716432762, 7   "updateTime": 1617962301153, 8   "version": 3, 9   "name": "azure_sql_table", 10  "description": "azure_sql_table", 11  "typeVersion": "1.0", 12  "serviceType": "Azure SQL Database", 13  "options": { 14    "schemaElementsAttribute": "columns", 15    "purviewEntityExtDef": "{}" 16  }, 17  "lastModifiedTS": "3", 18  "attributeDefs": [ 19    { 20      "name": "principalId", 21      "typeName": "int", 22      "isOptional": true, 23      "cardinality": "SINGLE", 24      "valuesMinCount": 0, 25      "valuesMaxCount": 1, 26      "isUnique": false, 27      "isIndexable": false, 28      "includeInNotification": false 29    }, 30    ... 31  ], 32  "superTypes": [ 33    "DataSet" 34  ], 35  "subTypes": [], 36  "relationshipAttributeDefs": [ 37    { 38      "name": "schema", 39      "typeName": "array&lt;avro_schema&gt;", 40      "isOptional": true, 41      "cardinality": "SET", 42      "valuesMinCount": -1, 43      "valuesMaxCount": -1, 44      "isUnique": false, 45      "isIndexable": false, 46      "includeInNotification": false, 47      "relationshipTypeName": "avro_schema_associatedEntities", 48      "isLegacyAttribute": false 49    }, 50    ... 51  ] 52 }</pre>	<b>AtlasEntityDef</b>  businessAttributeDefs: None relationshipAttributeDefs: ["schema", "inputToProcesses", "dbSchema", "columns", "attachedSchema", "meanings", "outputFromProcesses", "tabular_schema"] subTypes: [] superTypes: ["DataSet"]
<b>AtlasStructDef</b>  attributeDefs: AtlasAttributeDef[]		<b>AtlasStructDef</b>  attributeDefs: ["principalId", "objectType", "createTime", "modifiedTime"]
<b>AtlasBaseTypeDef</b>  category: TypeCategory createTime: number createdBy: string dateFormatter: DateFormat description: string guid: string name: string options: map of string serviceType: string typeVersion: string updateTime: number updatedBy: string version: number		<b>AtlasBaseTypeDef</b>  category: "ENTITY" createTime: 1614716432762 createdBy: "admin" dateFormatter: None description: "azure_sql_table" guid: "4fab04f5-9d26-42dd-9ed5-b64db9174970" name: "azure_sql_table" options: {...} serviceType: "Azure SQL Database" typeVersion: "1.0" updateTime: 1617962301153 updatedBy: "ServiceAdmin" version: 3



attributes	
key	Value
qualifiedName	mssql://sqlsvr8951.database.windows.net/sqldb8951/SalesLT/Address
replicatedFrom	null
replicatedTo	null
<b>name</b>	<b>Address</b>
description	null
owner	null
principalId	0
objectType	U
createTime	1612091789000
modifiedTime	1612091791000



entity	
azure_sql_table	
guid:	"384e62bd-23cd-49c4-b155-63f6f6f60000"
source:	"DataScan"
status:	"ACTIVE"
typeName:	"azure_sql_table"



relationshipAttributes	
key	value
meanings	[{"displayText": "MyTerm", "guid": "6eb3cfe3-1ed6-...", ...}]
schema	[]
inputToProcesses	[]
attachedSchema	[]
outputFromProcesses	[]
tabular_schema	None
<b>dbSchema</b>	{"displayText": "SalesLT", "guid": "b5389526-341d-...", ...}
columns	[{"displayText": "AddressID", "guid": "384e62bd...", ...}, {"displayText": "AddressLine1", "guid": "384e62bd...", ...}]

Address

Azure SQL Table

Edit Refresh

Overview Properties Schema Lineage Contacts Related

Show properties without a value

**Properties - basic**

createTime	Sun Jan 31 2021 11:16:29 GMT+0000 (GMT)
description	-
modifiedTime	Sun Jan 31 2021 11:16:31 GMT+0000 (GMT)
objectType	U
principalId	0
qualifiedName	mssql://sqlsvr8951.database.windows.net/sqldb8951/SalesLT/Address
replicatedFrom	-
replicatedTo	-

Address

Azure SQL Table

Edit Refresh

Overview Properties Schema Lineage Contacts **Related**

Showing 1 to 9 of 9 items

sqlsvr8951.database.windows.net	Azure SQL Server
sqldb8951	Azure SQL Database
<b>SalesLT</b>	Azure SQL Schema

Name

- Address
- Customer
- CustomerAddress
- ProductCategory
- ProductDescription
- ProductModel
- ProductModelProductDescription
- SalesOrderDetail
- SalesOrderHeader

## Relationships

*"A 'relationship' can be established between entities (e.g. Table <---> Columns)." Relationships have 2 ends, each of which specify cardinality, an entityDef type name, and a name.*

# AtlasRelationshipDef

## REST API

[GET] /v2/types/relationshipdef/name/{name}

## PURVIEW CLI

pv types readRelationshipdef --name 'azure\_sql\_table\_columns'

## CLASS

### AtlasRelationshipDef

endDef1: AtlasRelationshipEndDef  
endDef2: AtlasRelationshipEndDef  
propagateTags: PropagateTags  
relationshipCategory: RelationshipCategory  
relationshipLabel: string



### AtlasStructDef

attributeDefs: AtlasAttributeDef[]



### AtlasBaseTypeDef

category: TypeCategory  
createTime: number  
createdBy: string  
dateFormatter: DateFormat  
description: string  
guid: string  
name: string  
options: map of string  
serviceType: string  
typeVersion: string  
updateTime: number  
updatedBy: string  
version: number

## EXAMPLE RESPONSE

```
1  {
2      "category": "RELATIONSHIP",
3      "guid": "037367e8-f379-4078-abeb-675d40c6a0c8",
4      "createdBy": "admin",
5      "updatedBy": "admin",
6      "createTime": 1614716435770,
7      "updateTime": 1614716435770,
8      "version": 1,
9      "name": "azure_sql_table_columns",
10     "description": "azure_sql_table_columns",
11     "typeVersion": "1.0",
12     "serviceType": "Azure SQL Database",
13     "lastModifiedTS": "1",
14     "attributeDefs": [],
15     "relationshipCategory": "COMPOSITION",
16     "propagateTags": "NONE",
17     "endDef1": {
18         "type": "azure_sql_table",
19         "name": "columns",
20         "isContainer": true,
21         "cardinality": "SET",
22         "isLegacyAttribute": false
23     },
24     "endDef2": {
25         "type": "azure_sql_column",
26         "name": "table",
27         "isContainer": false,
28         "cardinality": "SINGLE",
29         "isLegacyAttribute": false
30     }
31 }
```

## RESPONSE MAPPED TO CLASS PROPERTIES

### AtlasRelationshipDef

endDef1: "azure\_sql\_table"  
endDef2: "azure\_sql\_column"  
propagateTags: "NONE"  
relationshipCategory: "COMPOSITION"  
relationshipLabel: None



### AtlasStructDef

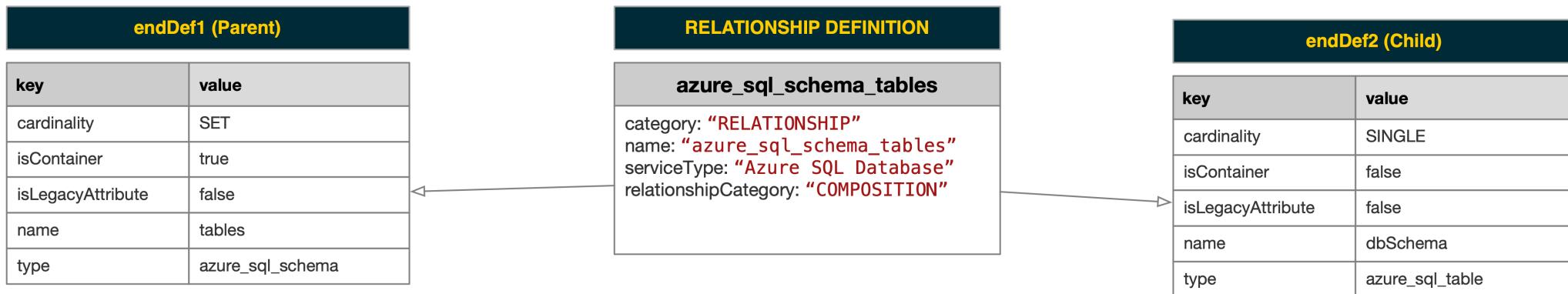
attributeDefs: []



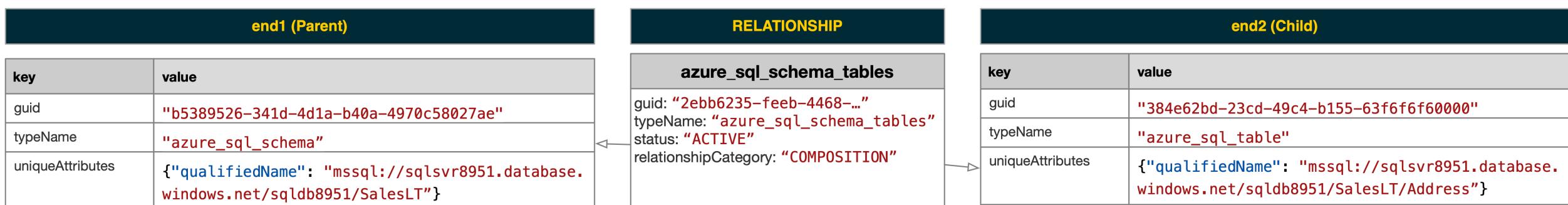
### AtlasBaseTypeDef

category: "RELATIONSHIP"  
createTime: 1614716435770  
createdBy: "admin"  
dateFormatter: None  
description: "azure\_sql\_table\_columns"  
guid: "037367e8-f379-4078-abeb-675d40c6a0c8"  
name: "azure\_sql\_table\_columns"  
options: None  
serviceType: "Azure SQL Database"  
typeVersion: "1.0"  
updateTime: 1614716435770  
updatedBy: "admin"  
version: 1

## Relationship Definition



## Relationship





# Getting Started



## Prerequisites

- An existing **Azure Purview** account.
- You will need sufficient permissions to **register** an application (i.e. service principal) within your Azure AD tenant.
- You will need sufficient permissions to assign the application a **role** (e.g. **Purview Data Curator**) in order for Azure Purview to trust your new service principal.

## Create a Service Principal

1. From the [Azure Portal](#), navigate to **Azure Active Directory**.
2. Select **App registrations**.
3. Select **New registration**.
4. Populate the **Register an application** form.
  - Name (e.g. `purviewapi`)
  - Supported account types (e.g. `Single tenant`)
  - Redirect URI (*this can be left blank*)
5. Click **Register**.

## 6. Copy the values Application (client) ID and the Directory (tenant) ID for later use.

Home > tayganr >

 purviewapi ⚡ ...

Search (Ctrl+/) Delete Endpoints Preview features

Overview Quickstart Integration assistant

Manage

Branding Authentication Certificates & secrets

Essentials

Display name: purviewapi

Application (client) ID: [REDACTED] Copy to clipboard

Directory (tenant) ID: [REDACTED]

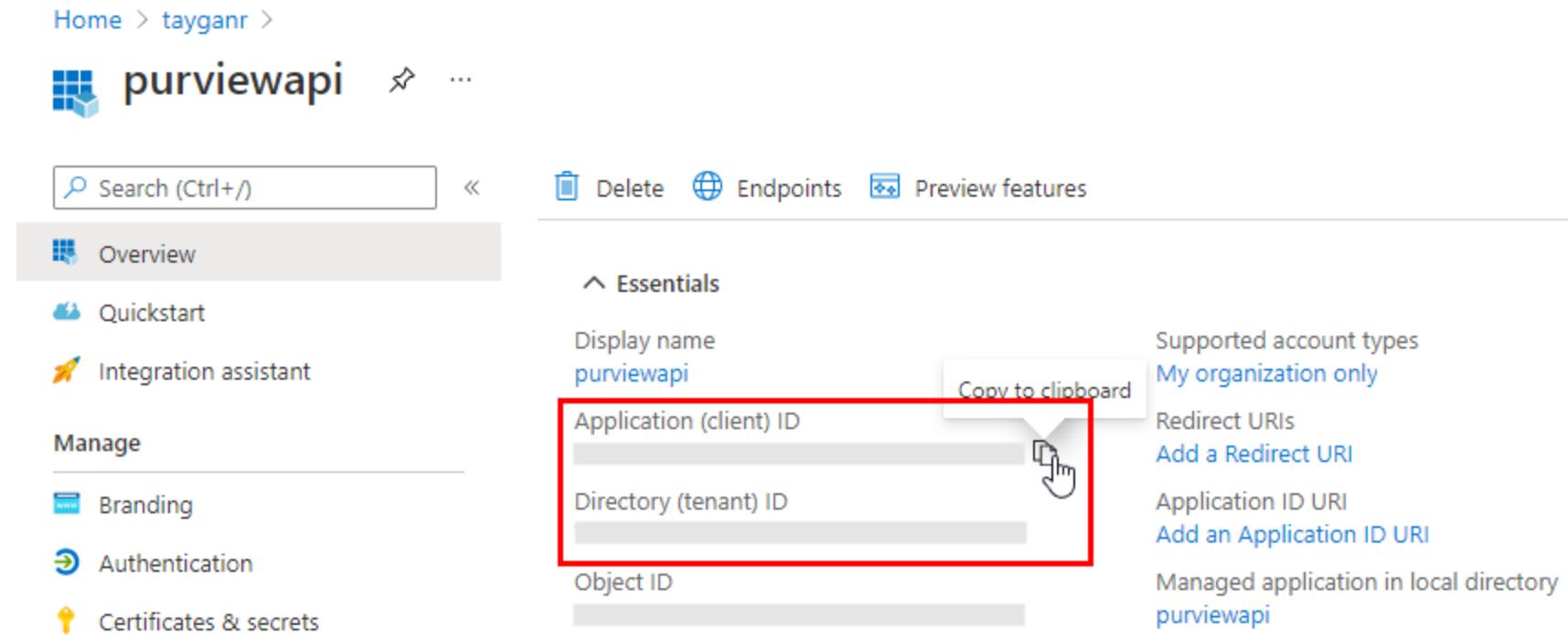
Object ID: [REDACTED]

Supported account types: My organization only

Redirect URIs: Add a Redirect URI

Application ID URI: Add an Application ID URI

Managed application in local directory: purviewapi



## Create a Client Secret

In order to use the service principal, we need generate a password (aka client secret).

1. From the [Azure Portal](#), navigate to **Azure Active Directory**.
2. Select **App registrations**.
3. Select your service principal from the list.
4. Select **Certificates & secrets**.
5. Select **New client secret**.
6. On the **Add a client secret** form, enter a **Description**, select an expiration period under **Expires**, and then select **Add**.

## 7. Copy the value client secret for later use.

 purviewapi | Certificates & secrets ⌂ ⌂

Search (Ctrl+/) Got feedback?

Overview Quickstart Integration assistant

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Manage

Branding Authentication Certificates & secrets Token configuration API permissions Expose an API App roles Owners Roles and administrators | Preview Manifest

Certificates

Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be referred to as public keys.

Upload certificate

Thumbprint	Start date	Expires	ID
No certificates have been added for this application.			

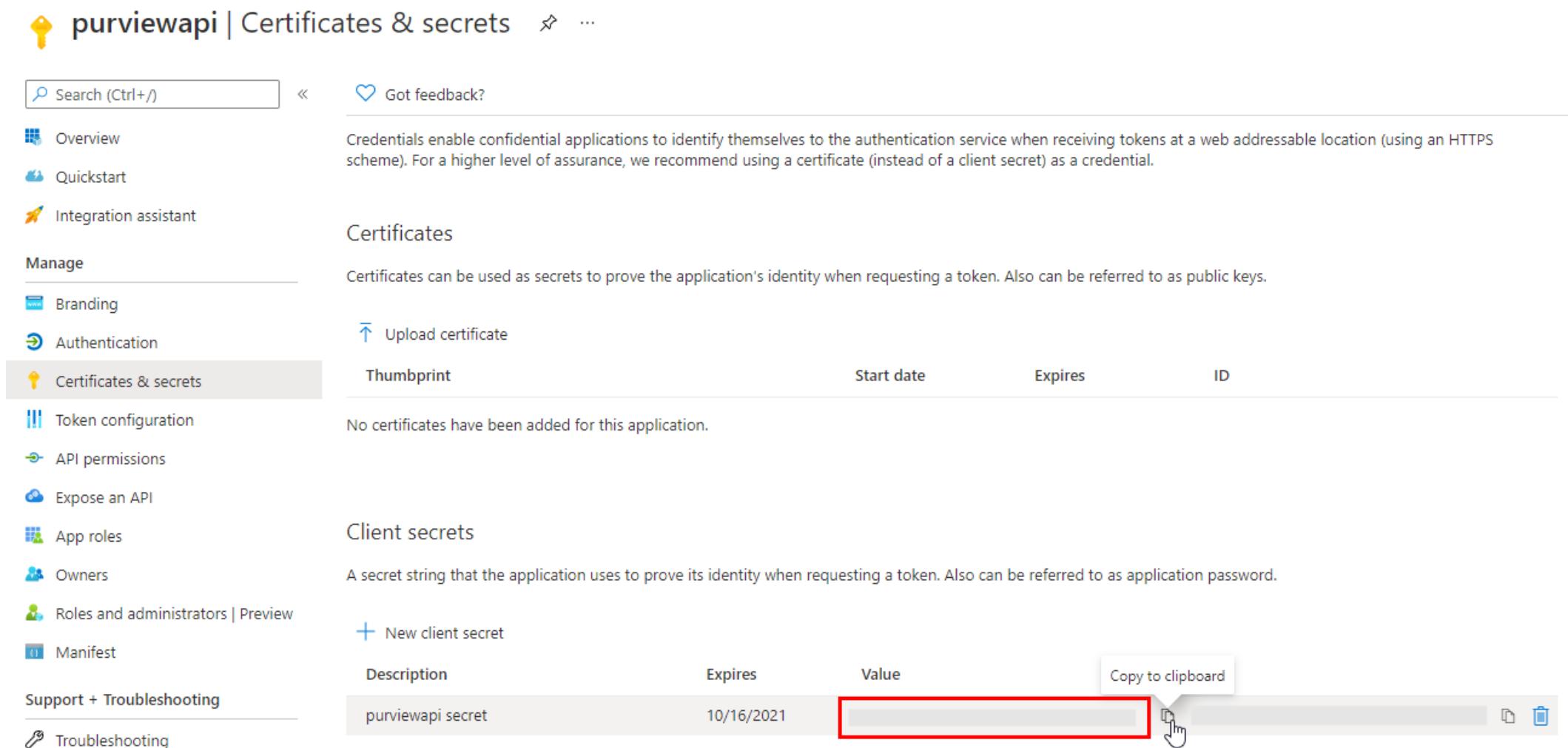
Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

New client secret

Description	Expires	Value	Copy to clipboard
purviewapi secret	10/16/2021	[REDACTED]	

Troubleshooting



## Configure Azure Purview to Trust the Service Principal

1. From the [Azure Portal](#), navigate to your Azure Purview account.
2. Select **Access control (IAM)**.
3. Select **Add role assignments**.
4. For **Role**, select `Purview Data Curator`.
5. For **Assign access to**, leave the default `User, group, or service principal`.
6. For **Select**, enter the name of the service principal (e.g. `purviewapi`) and then click on the name in the results pane.
7. Click **Save**.

# Request an Access Token

## Request

Property	Value
Method	POST
Endpoint	<code>https://login.microsoftonline.com/YOUR_TENANT_ID/oauth2/token</code>
Body Type	x-form-urlencoded

## Body (x-form-urlencoded)

Key	Value
<code>grant_type</code>	client_credentials
<code>client_id</code>	<code>YOUR_CLIENT_ID</code>
<code>client_secret</code>	<code>YOUR_CLIENT_SECRET</code>
<code>resource</code>	<a href="https://purview.azure.net">https://purview.azure.net</a>

## ✉ Request an Access Token (Postman)

The screenshot shows the Postman application interface for making an API request to obtain an access token.

**Request Details:**

- Method:** POST
- URL:** https://login.microsoftonline.com/7...7/oauth2/token
- Body (x-www-form-urlencoded):**

KEY	VALUE	DESCRIPTION
grant_type	client_credentials	
client_id	[REDACTED]	
client_secret	[REDACTED]	
resource	https://purview.azure.net	
Key	Value	Description

**Response Status:** Status: 200 OK

**Response Body (Pretty JSON):**

```
1 {  
2   "token_type": "Bearer",  
3   "expires_in": "86399",  
4   "ext_expires_in": "86399",  
5   "expires_on": "1619115724",  
6   "not_before": "1619029024",  
7   "resource": "https://purview.azure.net",  
8   "access_token": "eyJhdQ10ijodHR  
eyJ...  
A10jE2MTkxMTU3M  
Lm5ldC83MmY5Dh  
l...  
WxWg5tmsPfcu63  
KrdS33kY2p1DmCS  
9 }
```

The response body contains a JSON object with tokens and their expiration details. The access token itself is heavily redacted.

 Request an Access Token (Python)

```
# Set Variables
AZURE_CLIENT_ID = "YOUR_CLIENT_ID"
AZURE_TENANT_ID = "YOUR_TENANT_ID"
AZURE_CLIENT_SECRET = "YOUR_CLIENT_SECRET"

# Prepare Request
data = {
    'grant_type': 'client_credentials',
    'client_id': AZURE_CLIENT_ID,
    'client_secret': AZURE_CLIENT_SECRET,
    'resource': 'https://purview.azure.net',
}
url = 'https://login.microsoftonline.com/{0}/oauth2/token'.format(AZURE_TENANT_ID)

# Get Access Token
import requests
request = requests.post(url, data=data)
response = request.json()
access_token = response['access_token']
print(access_token)
```



## Examples

## ⚙️ 1.1 Retrieve all glossaries

### Request

Property	Value
Method	GET
Endpoint	<code>https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com/api/atlas/v2/glossary</code>

### Headers

Key	Value
Authorization	Bearer YOUR_ACCESS_TOKEN



## 1.2 Retrieve all glossaries (Python)

```
# Set Variables
ATLAS_ENDPOINT = "https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com"
ACCESS_TOKEN = "YOUR_ACCESS_TOKEN"

# Prepare Request
url = '{0}/api/atlas/v2/glossary'.format(ATLAS_ENDPOINT)
headers = {'Authorization': 'Bearer {0}'.format(ACCESS_TOKEN)}

# Get Data
import requests
import json
request = requests.get(url, headers=headers)
response = request.json()
print(json.dumps(response, indent=4))
```



## 1.3 Retrieve all glossaries ([purviewcli](#))

```
pv glossary read
```

## ⚙️ 2.1 Create a custom type (classification)

### Request

Property	Value
Method	POST
Endpoint	<code>https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com/api/atlas/v2/types/typedefs</code>

### Headers

Key	Value
Authorization	Bearer YOUR_ACCESS_TOKEN
Content-Type	application/json

## Body (Example Payload)

```
{  
    "classificationDefs": [  
        {  
            "category": "CLASSIFICATION",  
            "description": "This is a custom classification.",  
            "name": "TAYGAN.CUSTOM.CLASSIFICATION"  
        }  
    ]  
}
```



## 2.2 Create a custom classification (Python)

```
# Set Variables
ATLAS_ENDPOINT = "https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com"
ACCESS_TOKEN = "YOUR_ACCESS_TOKEN"

# Prepare Request
url = '{0}/api/atlas/v2/types/typedefs'.format(ATLAS_ENDPOINT)
headers = {
    'Authorization': 'Bearer {0}'.format(ACCESS_TOKEN),
    'Content-Type': 'application/json'
}
data = {
    "classificationDefs": [
        {
            "category": "CLASSIFICATION",
            "description": "This is a custom classification.",
            "name": "TAYGAN.CUSTOM.CLASSIFICATION"
        }
    ]
}

# Create Type
request = requests.post(url, headers=headers, json=data)
response = request.json()
print(json.dumps(response, indent=4))
```

## ⚙️ 3.1 Delete a custom type (classification)

### Request

Property	Value
Method	DELETE
Endpoint	<code>https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com/api/atlas/v2/types/typedef/name/{name}</code>

### Headers

Key	Value
Authorization	Bearer YOUR_ACCESS_TOKEN



## 3.2 Delete a custom classification (Python)

```
# Set Variables
ATLAS_ENDPOINT = "https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com"
ACCESS_TOKEN = "YOUR_ACCESS_TOKEN"

# Prepare Request
classification_name = 'TAYGAN.CUSTOM.CLASSIFICATION'
url = '{0}/api/atlas/v2/types/typedef/name/{1}'.format(ATLAS_ENDPOINT, classification_name)
headers = {'Authorization': 'Bearer {0}'.format(ACCESS_TOKEN)}

# Delete Type
request = requests.delete(url, headers=headers)
print(request.status_code)
```

## ⚙️ 4.1 Get an entity

### Request

Property	Value
Method	GET
Endpoint	<code>https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com/api/atlas/v2/entity/guid/{guid}</code>

### Headers

Key	Value
Authorization	Bearer YOUR_ACCESS_TOKEN



## 4.2 Get an entity (Python)

```
# Set Variables
ATLAS_ENDPOINT = "https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com"
ACCESS_TOKEN = "YOUR_ACCESS_TOKEN"

# Prepare Request
guid = 'f0a9c131-8cad-4fa9-928d-f4f6f6f60000'
url = '{0}/api/atlas/v2/entity/guid/{1}'.format(ATLAS_ENDPOINT, guid)
headers = {'Authorization': 'Bearer {0}'.format(ACCESS_TOKEN)}

# Get Entity
request = requests.get(url, headers=headers)
response = request.json()
print(json.dumps(response, indent=4))
```

## ⚙️ 5.1 Create an entity

### Request

Property	Value
Method	POST
Endpoint	<code>https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com/api/atlas/v2/entity</code>

### Headers

Key	Value
Authorization	Bearer YOUR_ACCESS_TOKEN
Content-Type	application/json

## Body (Example Payload)

```
{  
  "entity": {  
    "typeName": "azure_sql_table",  
    "attributes": {  
      "qualifiedName": "mssql://sqlsvr8951.database.windows.net/sqlDb8951/SalesLT/CustomEntity",  
      "name": "CustomEntity",  
      "description": "This is a custom entity."  
    }  
  }  
}
```



## 5.2 Create an entity (Python)

```
# Set Variables
ATLAS_ENDPOINT = "https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com"
ACCESS_TOKEN = "YOUR_ACCESS_TOKEN"

# Prepare Request
url = '{0}/api/atlas/v2/entity'.format(ATLAS_ENDPOINT)
headers = {
    'Authorization': 'Bearer {0}'.format(ACCESS_TOKEN),
    'Content-Type': 'application/json'
}
data = {
    "entity": {
        "typeName": "azure_sql_table",
        "attributes": {
            "qualifiedName": "mssql://sqlsvr8951.database.windows.net/sql8951/SalesLT/CustomEntity",
            "name": "CustomEntity",
            "description": "This is a custom entity."
        }
    }
}

# Create Type
request = requests.post(url, headers=headers, json=data)
response = request.json()
print(json.dumps(response, indent=4))
```

## ⚙️ 6.1 Delete an entity

### Request

Property	Value
Method	DELETE
Endpoint	<code>https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com/api/atlas/v2/entity/guid/{guid}</code>

### Headers

Key	Value
Authorization	Bearer YOUR_ACCESS_TOKEN

## 6.2 Delete an entity (Python)

```
# Set Variables
ATLAS_ENDPOINT = "https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com"
ACCESS_TOKEN = "YOUR_ACCESS_TOKEN"

# Prepare Request
guid = 'e7714771-86c6-4083-84be-3df6f6f60000'
url = '{0}/api/atlas/v2/entity/guid/{1}'.format(ATLAS_ENDPOINT, guid)
headers = {'Authorization': 'Bearer {0}'.format(ACCESS_TOKEN)}

# Delete Type
request = requests.delete(url, headers=headers)
print(request.status_code)
```

## ⚙️ 7.1 Get a relationship

### Request

Property	Value
Method	GET
Endpoint	<code>https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com/api/atlas/v2/relationship/guid/{guid}</code>

### Headers

Key	Value
Authorization	Bearer YOUR_ACCESS_TOKEN

## 7.2 Get a relationship (Python)

```
# Set Variables
ATLAS_ENDPOINT = "https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com"
ACCESS_TOKEN = "YOUR_ACCESS_TOKEN"

# Prepare Request
guid = '86e9c889-5fd4-4f44-b086-9fa9e974f9d0'
url = '{0}/api/atlas/v2/relationship/guid/{1}'.format(ATLAS_ENDPOINT, guid)
headers = {'Authorization': 'Bearer {0}'.format(ACCESS_TOKEN)}

# Get Entity
request = requests.get(url, headers=headers)
response = request.json()
print(json.dumps(response, indent=4))
```

## ⚙️ 8.1 Create a relationship (custom lineage)

### Request

Property	Value
Method	POST
Endpoint	<code>https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com/api/atlas/v2/relationship</code>

### Headers

Key	Value
Authorization	Bearer YOUR_ACCESS_TOKEN
Content-Type	application/json

## Body (Example Payload)

```
{  
    "typeName": "process_dataset_outputs",  
    "end1": {  
        "guid": "8cf4916b-a0a6-4c21-9efe-1e6c43fe886c",  
        "typeName": "adf_copy_activity",  
        "uniqueAttributes": {  
            "qualifiedName": "/subscriptions/2c334b6c-e5...."  
        }  
    },  
    "end2": {  
        "guid": "7ccc6d67-7a15-42a6-8fd8-39f6f6f60000",  
        "typeName": "azure_sql_table",  
        "uniqueAttributes": {  
            "qualifiedName": "mssql://sqlsvr8951.database.windows.net/sqlDb8951/Taygan/TwitterOne"  
        }  
    }  
}
```

## 8.2 Create a relationship (custom lineage)

```
# Set Variables
ATLAS_ENDPOINT = "https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com"
ACCESS_TOKEN = "YOUR_ACCESS_TOKEN"

# Prepare Request
url = '{0}/api/atlas/v2/relationship'.format(ATLAS_ENDPOINT)
headers = {
    'Authorization': 'Bearer {0}'.format(ACCESS_TOKEN),
    'Content-Type': 'application/json'
}
data = {
    "typeName": "process_dataset_outputs",
    "end1": {
        "guid": "8cf4916b-a0a6-4c21-9efe-1e6c43fe886c",
        "typeName": "adf_copy_activity",
        "uniqueAttributes": {
            "qualifiedName": "/subscriptions/2c334b6c-e5...."
        }
    },
    "end2": {
        "guid": "7ccc6d67-7a15-42a6-8fd8-39f6f6f60000",
        "typeName": "azure_sql_table",
        "uniqueAttributes": {
            "qualifiedName": "mssql://sqlsvr8951.database.windows.net/sqlDb8951/Taygan/TwitterOne"
        }
    }
}

# Create Type
request = requests.post(url, headers=headers, json=data)
response = request.json()
print(json.dumps(response, indent=4))
```

## ⚙️ 9.1 Delete a relationship

### Request

Property	Value
Method	DELETE
Endpoint	<code>https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com/api/atlas/v2/relationship/guid/{guid}</code>

### Headers

Key	Value
Authorization	Bearer YOUR_ACCESS_TOKEN

## 9.2 Delete a relationship (Python)

```
# Set Variables
ATLAS_ENDPOINT = "https://YOUR_PURVIEW_ACCOUNT.catalog.purview.azure.com"
ACCESS_TOKEN = "YOUR_ACCESS_TOKEN"

# Prepare Request
guid = 'dab534c8-ace7-49be-95a1-def4e394c74a'
url = '{0}/api/atlas/v2/relationship/guid/{1}'.format(ATLAS_ENDPOINT, guid)
headers = {'Authorization': 'Bearer {0}'.format(ACCESS_TOKEN)}

# Delete Type
request = requests.delete(url, headers=headers)
print(request.status_code)
```