Cloudera Runtime 7.3.2

Atlas Use Cases

Date published: 2020-07-28
Date modified: 2026-03-31

https://docs.cloudera.com/

Legal Notice

Contents

Using metadata for cluster governance

Concepts for collecting, creating, and using metadata.

What is Apache Atlas?

Atlas is a metadata management and governance system designed to help you find, organize, and manage data assets. Atlas creates “entities” or metadata representations of objects and operations in your data lake. You can add business metadata to these entities so you can use business vocabulary to make it easier to search for specific assets.

Apache Atlas uses metadata to create lineage relationships

Atlas reads the content of the metadata it collects to build relationships among data assets. When Atlas receives query information, it notes the input and output of the query and generates a lineage map that traces how data is used and transformed over time. This visualization of data transformations allows governance teams to quickly identify the source of data and to understand the impact of data and schema changes.

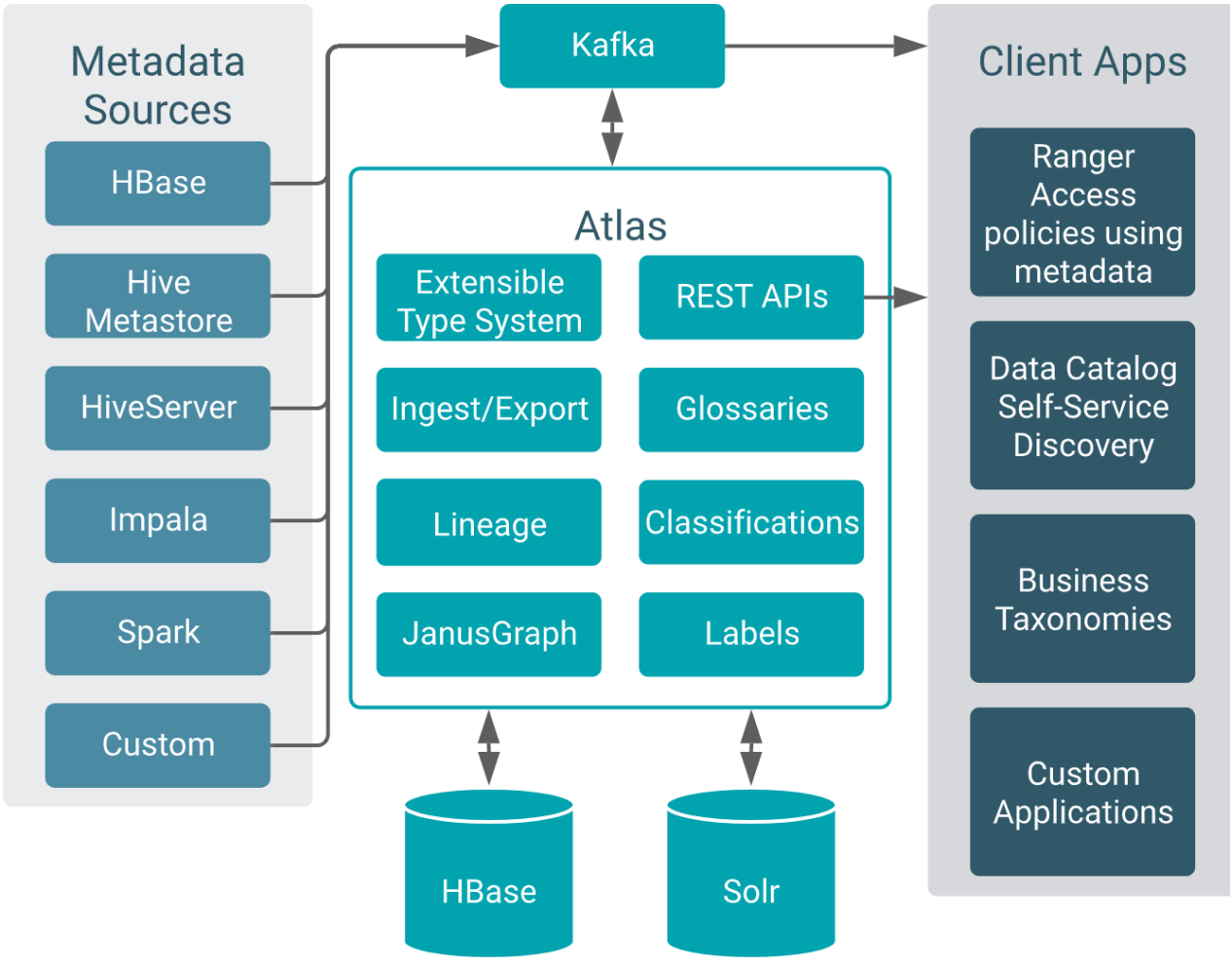
Adding to entity metadata makes searching easier

Atlas manages *classifications* and *labels* that you create and use to enhance the metadata for data assets. You can create and organize classifications and labels to use for anything from identifying data cleansing stages to recording user comments and insights on specific data assets. When you use classifications, the Atlas Dashboard makes it easy to search, group, report, and further annotate the entities you label. Classifications themselves can be organized into hierarchies to make them easier to manage.

Atlas also provides an infrastructure to create and maintain business ontologies to label your data assets. Atlas’ “glossaries” include “terms” so you can build agreed-upon lists for department- or organization-wide vocabulary to identify and manage data. Adding a term gives you a single-click report of entities identified by that term.

Apache Atlas architecture

Atlas runs as an independent service in a Hadoop environment. Many Hadoop data processing and storage services include Atlas add-ons that publish metadata for the services’ activities to a Kafka message topic. Atlas reads the messages and stores them in JanusGraph to model the relationships among entities. The datastore behind JanusGraph is HBase. Atlas stores a search index in Solr to take advantage of Solr’s search functionality.



Pre-defined hooks exist for Hive, Impala, Kafka, NiFi, Spark, and Sqoop.

Atlas also provides “bridges” that import metadata for all of the existing data assets in a given source. For example, if you start Atlas after you’ve already created databases and tables in Hive, you can import metadata for the existing data assets using the Hive bridge. Bridges use the Atlas API to import the metadata rather than publishing messages to Kafka.

If you need a hook or bridge to automate collecting metadata from another source, use the Atlas Java API to create a custom Atlas addon.

Note: Governance through Apache Atlas is just one element of a secure production cluster: Cloudera supports Atlas only when it runs on a cluster where Kerberos is enabled to authenticate users.

Data Stewardship with Apache Atlas

Concepts for collecting, creating, and using metadata.

The value of Atlas metadata for organizing and finding data increases when you augment the generated “technical” metadata by using your organization's business vocabulary. Here’s how you would do that:

- Tools. You can add metadata to entities using labels, classifications, attributes, and a glossary of terms. The glossary allows you to identify synonyms so that the vocabulary from different teams does not get in the way of identifying the same data.

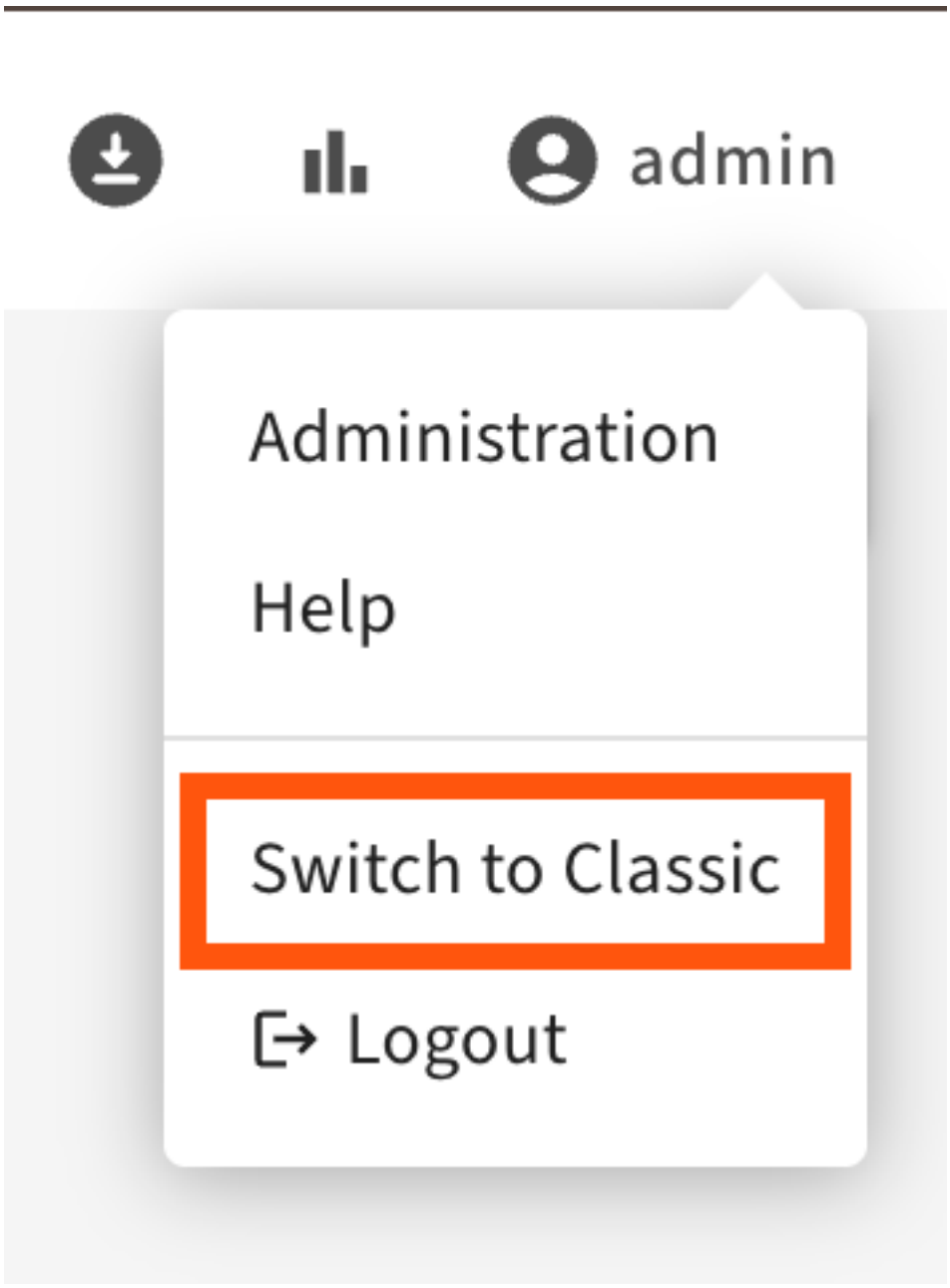
- Planning. Figure out who and how to apply the tools; set up an overall plan for what kinds of metadata you want to apply, design some conventions for how to apply them and who can apply them. Design some processes to oversee metadata as it collects to make sure the results are consistent and useful; identify synonyms and antonyms.
- Examples. This document includes examples of ways you can organize your metadata; strategies that describe how to optimize for specific use cases.

Apache Atlas dashboard tour

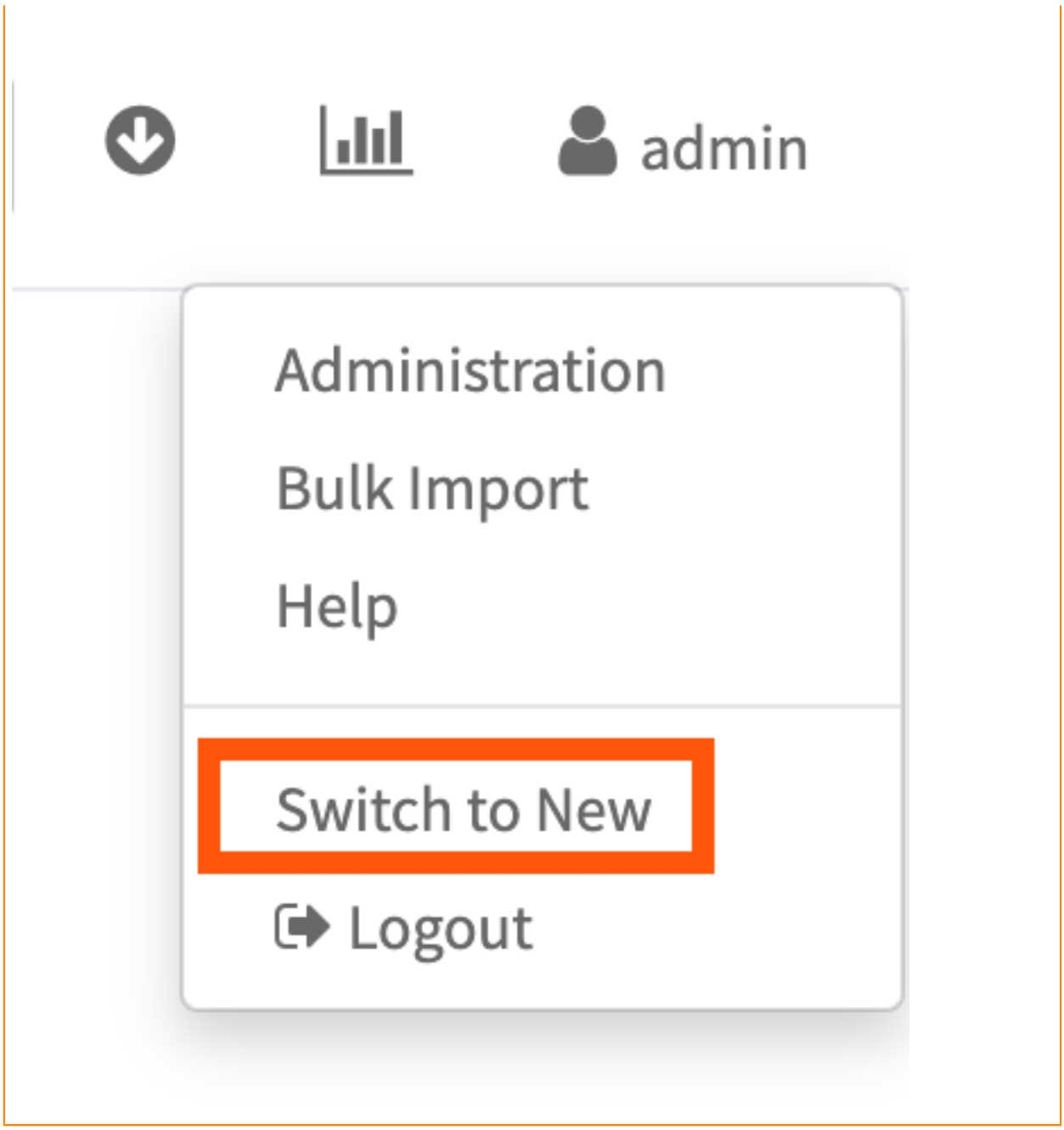
The Apache Atlas dashboard provides a unified metadata management interface for the Classic and New React-based user experiences to perform entity searches, manage classifications, and visualize lineage graphs.

You can switch between the **Classic** and the **New** React-based redesigned user interface.

Figure 1: Switching between Classic and New UI



For Classic User Interface

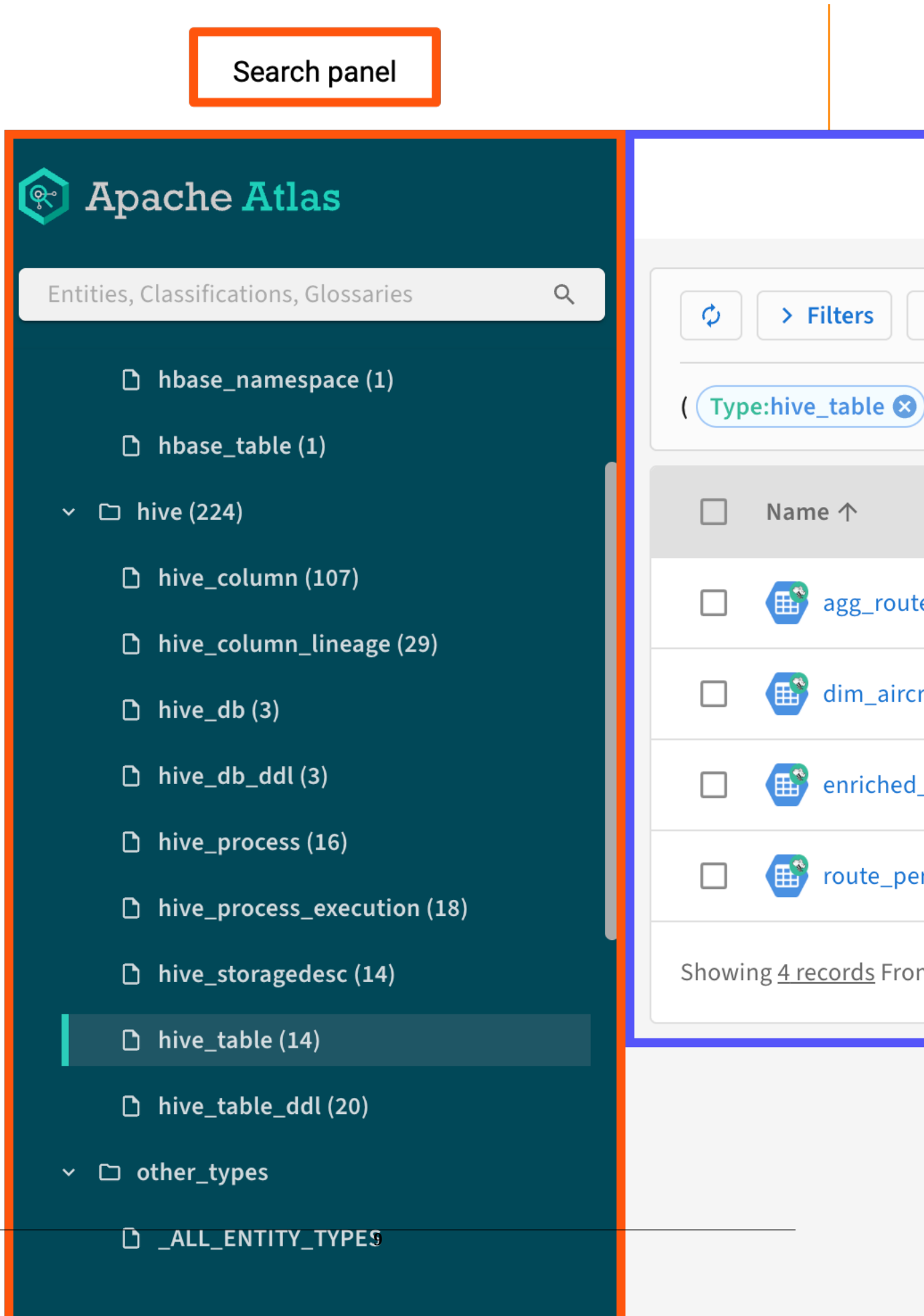


The Atlas user interface is a metadata dashboard with the following parts:

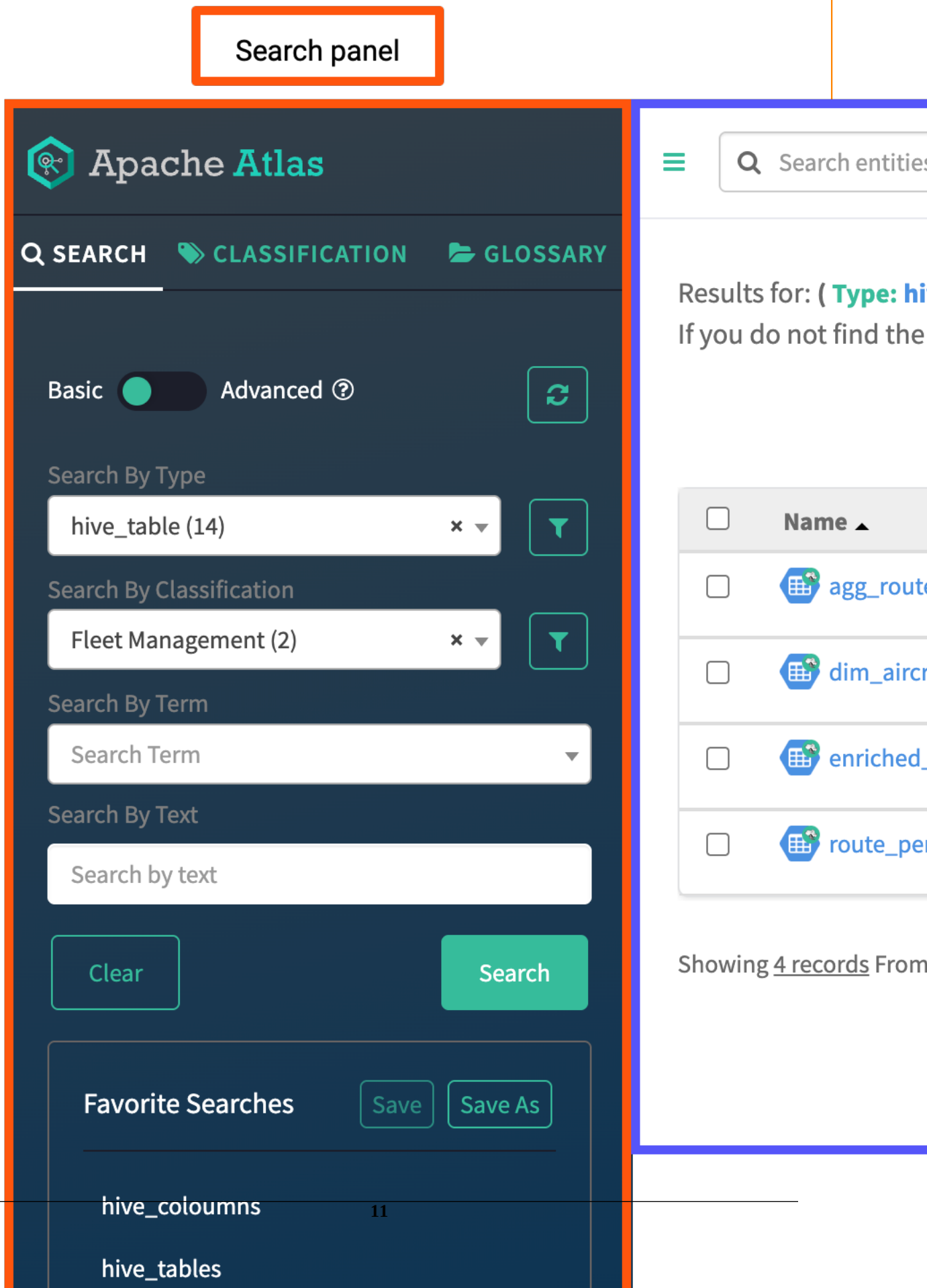
- **Search** panel on the left.
- **Detail** panel on the right where search results are displayed. You can drill into a specific entity to view its the details. Each detail page contains a header section and a series of tabbed panels, all of which are oriented to the metadata for that entity type.

Figure 2: Filtered search results

For New User Interface


Search panel
Apache Atlas
Entities, Classifications, Glossaries
hbase_namespace (1)
hbase_table (1)
hive (224)
hive_column (107)
hive_column_lineage (29)
hive_db (3)
hive_db_ddl (3)
hive_process (16)
hive_process_execution (18)
hive_storagedesc (14)
hive_table (14)
hive_table_ddl (20)
other_types
_ALL_ENTITY_TYPES
Filters
Type:hive_table
Name
agg_route
dim_aircr
enriched_
route_per
Showing 4 records Fron

For Classic User Interface


Search panel
Apache Atlas
SEARCH
CLASSIFICATION
GLOSSARY
Basic
Advanced
Search By Type
hive_table (14)
Search By Classification
Fleet Management (2)
Search By Term
Search Term
Search By Text
Search by text
Clear
Search
Favorite Searches
Save
Save As
hive_coloumns
hive_tables
Search entities
Results for: (Type: hi
If you do not find the
Name
agg_route
dim_aircr
enriched_
route_per
Showing 4 records From

Searching

For New User Interface

The new search panel automatically lists all available entity types, classifications and glossary terms. Clicking an item in one of the categories automatically lists all relevant members.

The Atlas user interface includes radio buttons that enable the following filtering options:

Entities

Displays empty service types

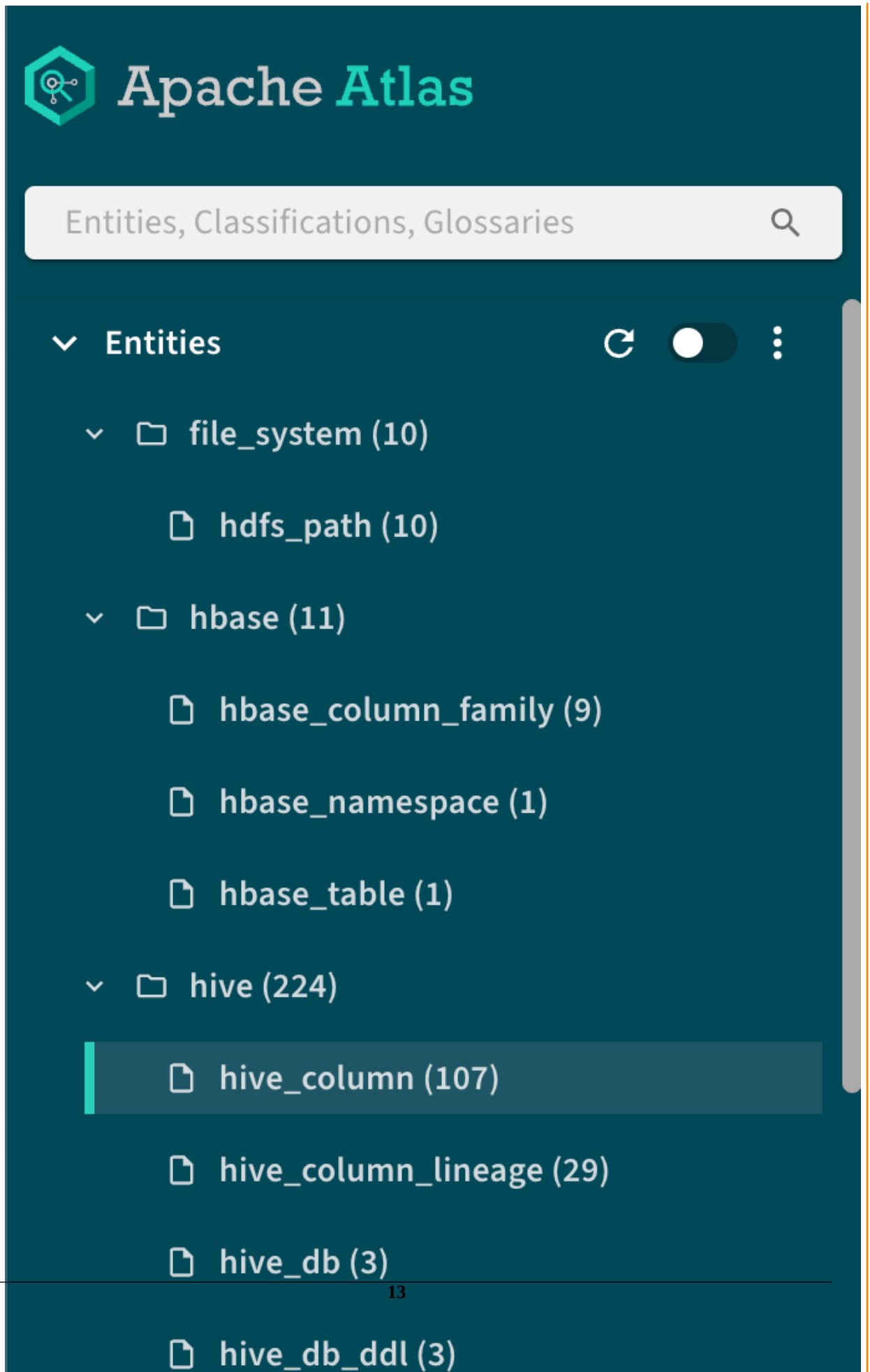
Classifications

Displays unused classifications

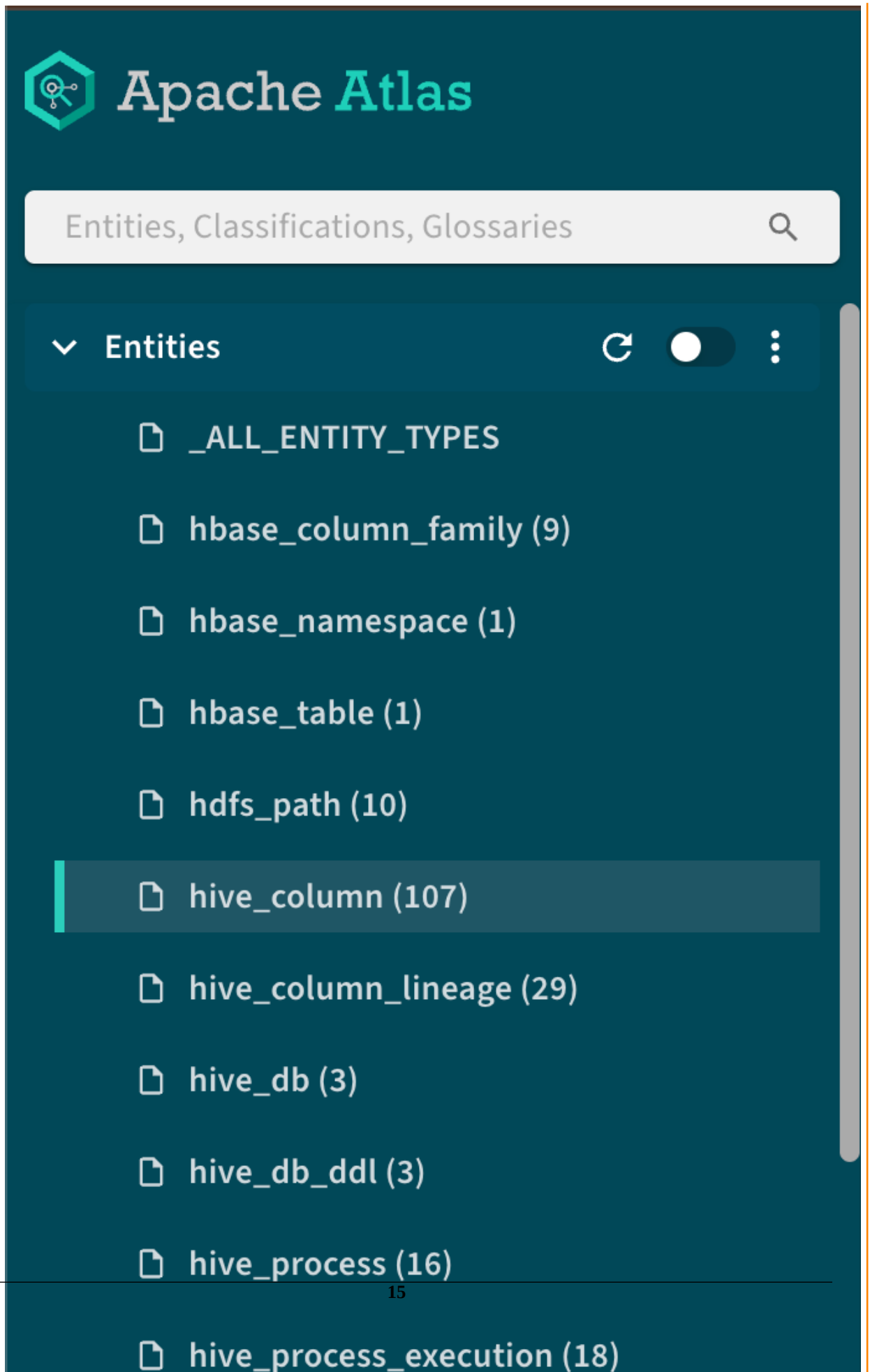
Glossary

Displays category or terms

Figure 3: Search panel in New UI


Apache Atlas
Entities, Classifications, Glossaries
Entities
file_system (10)
hdfs_path (10)
hbase (11)
hbase_column_family (9)
hbase_namespace (1)
hbase_table (1)
hive (224)
hive_column (107)
hive_column_lineage (29)
hive_db (3)
hive_db_ddl (3)

Entities and classifications can be collapsed to a flat tree, by clicking > Show flat tree.


Apache Atlas
Entities, Classifications, Glossaries
Entities
_ALL_ENTITY_TYPES
hbase_column_family (9)
hbase_namespace (1)
hbase_table (1)
hdfs_path (10)
hive_column (107)
hive_column_lineage (29)
hive_db (3)
hive_db_ddl (3)
hive_process (16)
hive_process_execution (18)

For Classic User Interface

The Classic search panel contains the following tabs for searching:

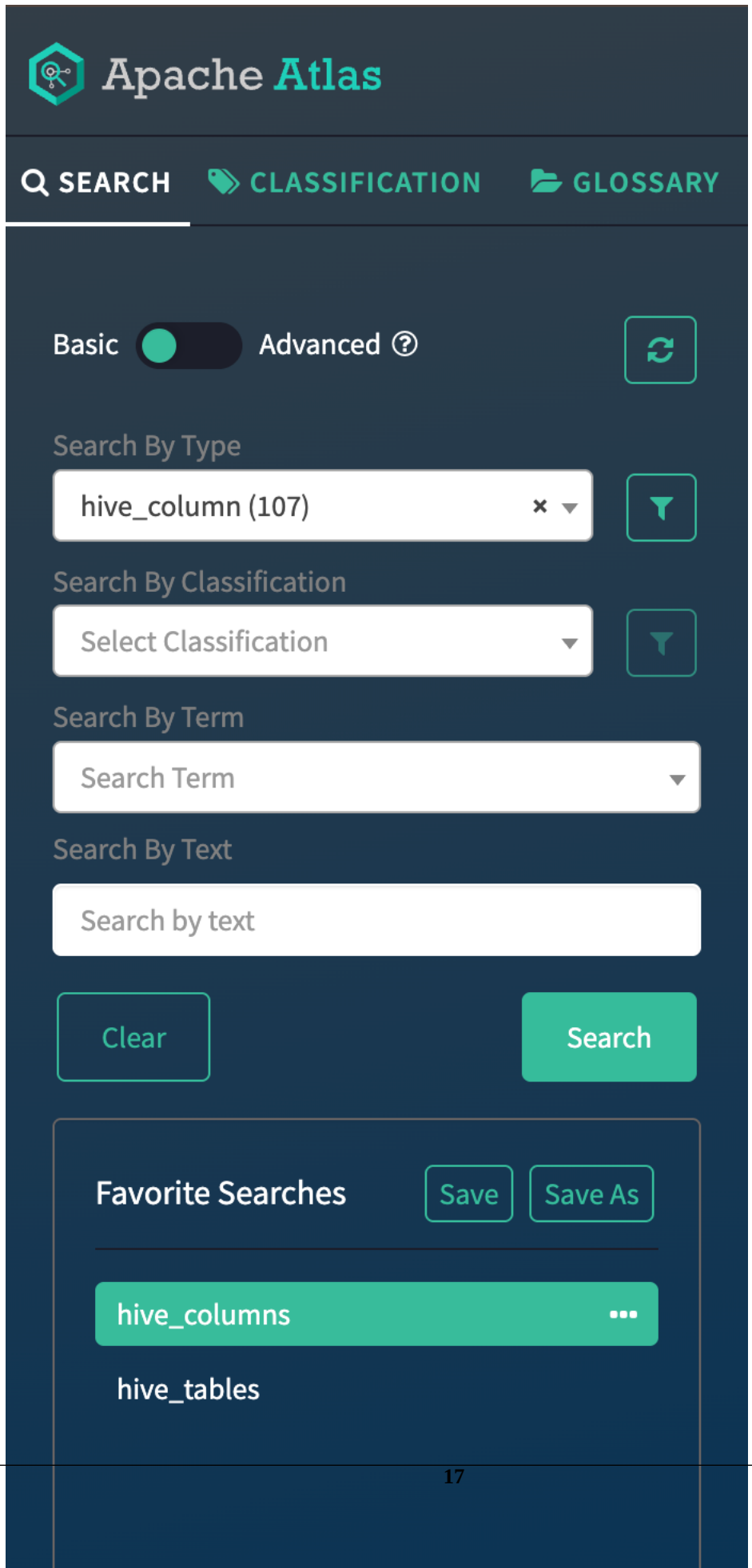
- **Search**

  Select from existing lists of metadata types to narrow the search results. Switching to the Advanced search allows you to enter specific Atlas DSL search queries. Both Basic and Advanced searches can be saved for easy reuse in **Favorite Searches**.
- **Classifications**

  Access predefined searches based on classifications
- **Glossary**

  Search using predefined glossary terms

Figure 4: Search panel in Classic UI


Apache Atlas
SEARCH
CLASSIFICATION
GLOSSARY
Basic
Advanced
Search By Type
hive_column (107)
Search By Classification
Select Classification
Search By Term
Search Term
Search By Text
Search by text
Clear
Search
Favorite Searches
Save
Save As
hive_columns
hive_tables

Classifications

In the **Classification** option, selecting a classification displays all the entities that are marked with that classification. Locate a specific classification using the search box or browse the classification hierarchy that you defined during classification creation. The **Classification** tab displays all related assets, including processes.

For New User Interface

Figure 5: Searching with classifications

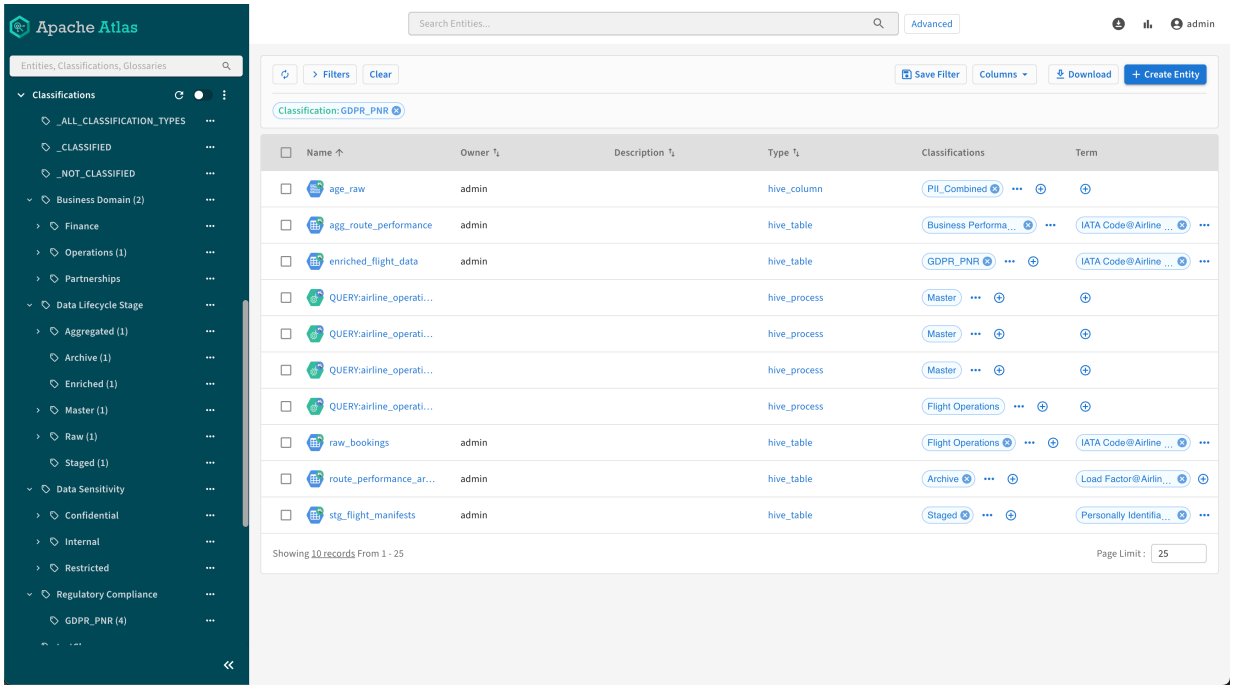
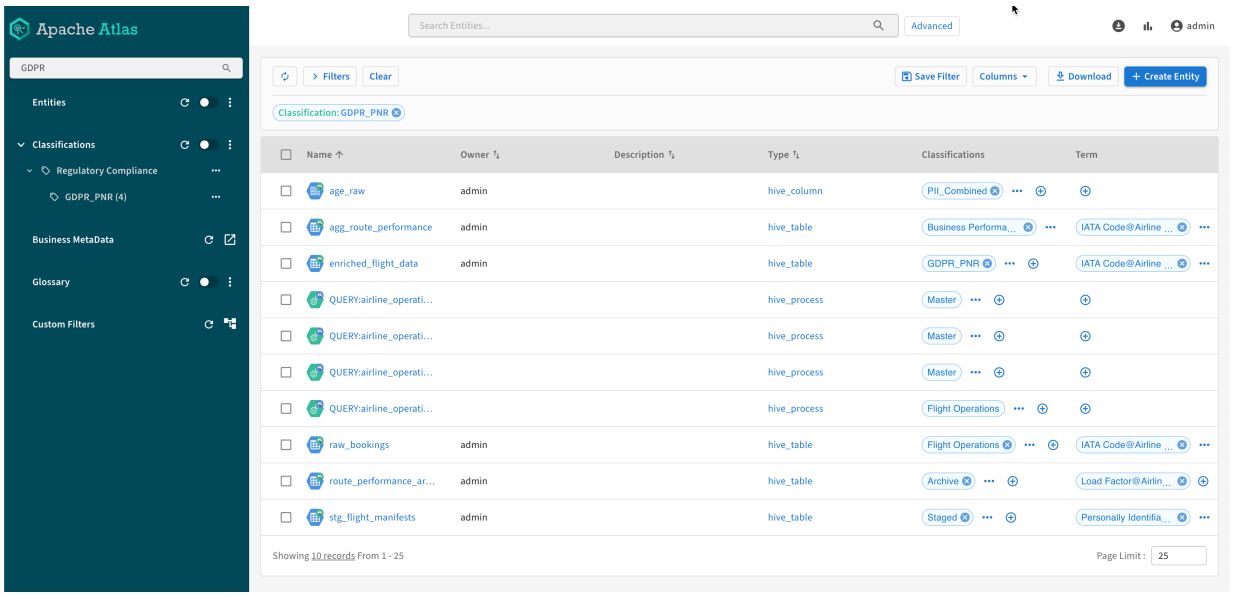


Figure 6: Searching for a classification



For Classic User Interface

Figure 7: Searching with classifications

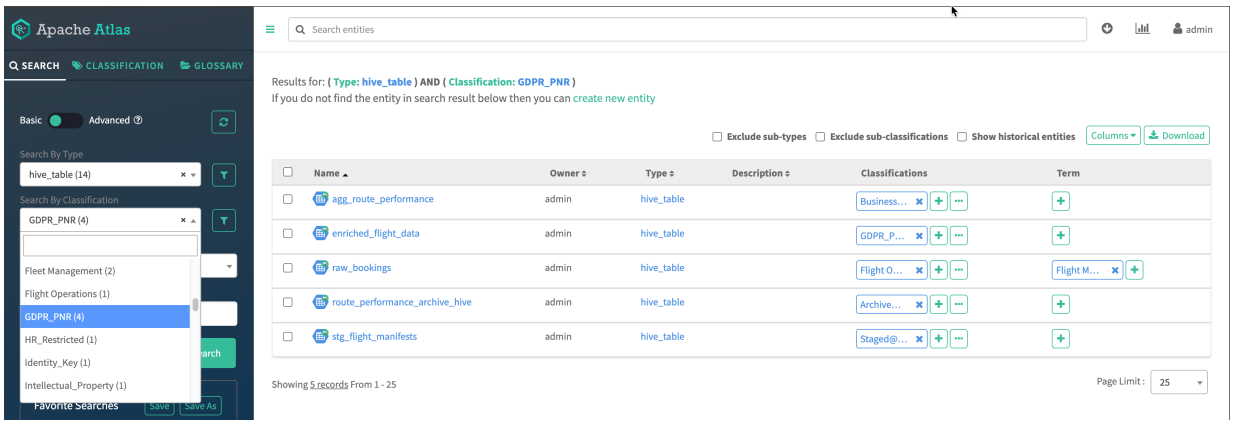
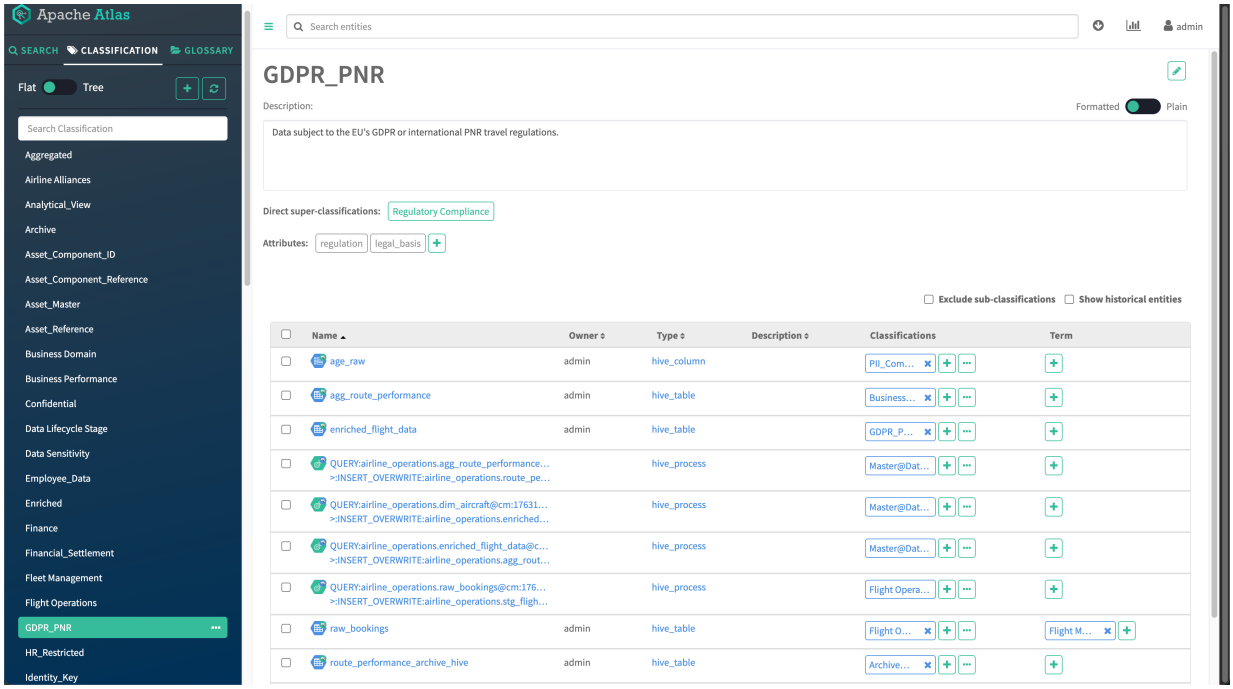


Figure 8: Classification tab

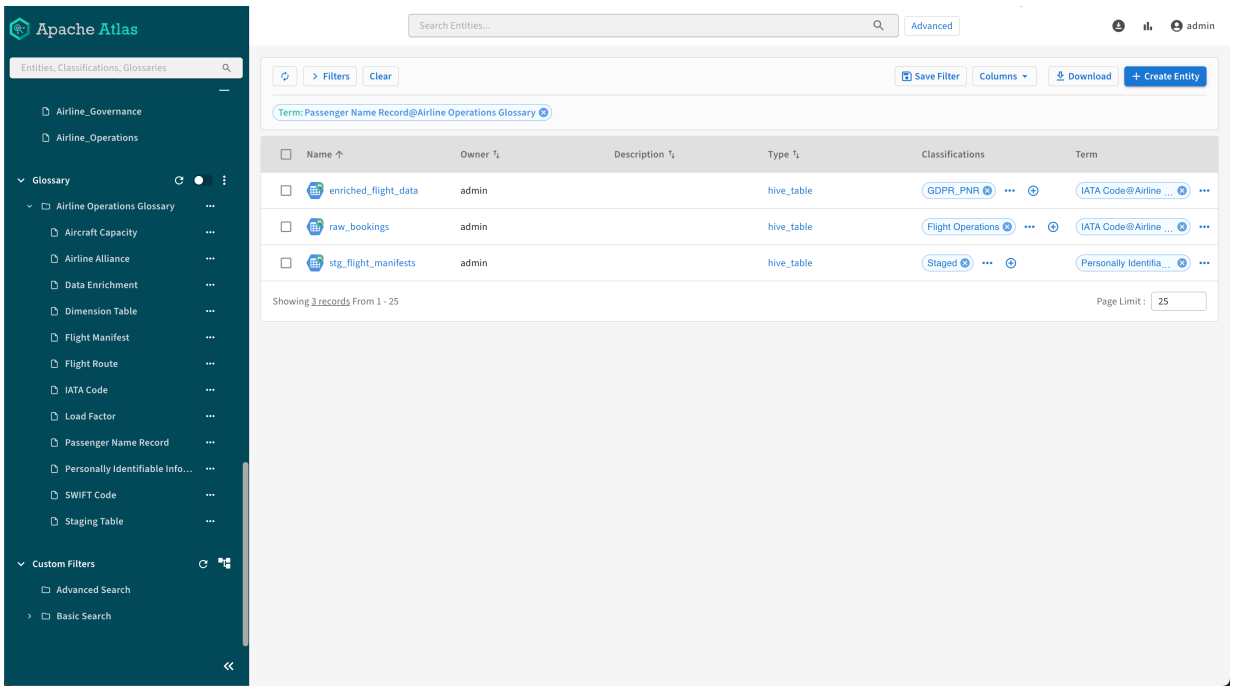


Glossary

In the **Glossary** option, selecting a term displays all the entities that are marked with that term. Locate a specific term using the search box or browse terms by glossary. Use the category view to browse the hierarchy of your organization's business glossary. Selecting a category displays its assigned terms in the display pane. When you select one of those terms, Atlas displays the associated entities.

For New User Interface

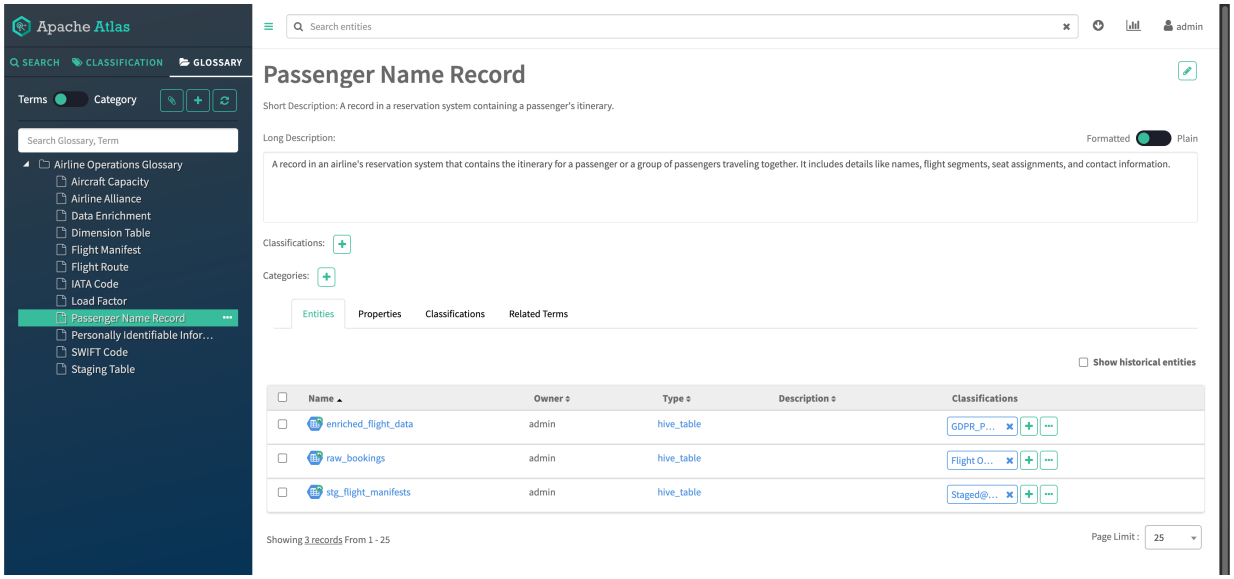
Figure 9: Searching with the glossary



Note: You can download the import template for uploading glossaries by clicking the ⋮ > Download Import Template. You can create glossaries and terms manually by clicking the ⋮ > ✚ Create Glossary icon.

For Classic User Interface

Figure 10: Glossary tab



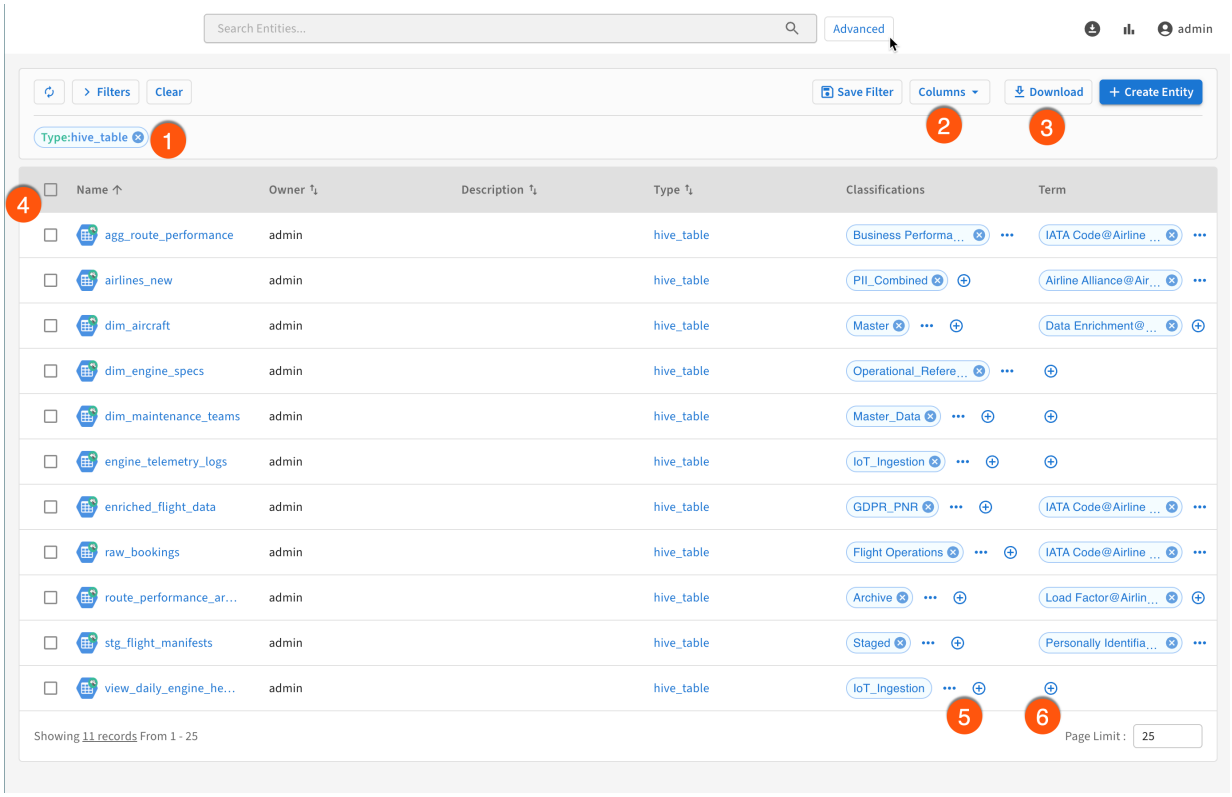
Note: You can download the import template for uploading glossaries by clicking the 📎 icon. You can create glossaries and terms manually by clicking the ✚ icon.

Search results

When Atlas returns search results, the UI displays a paginated list of entities that match the search criteria. You can return to search options to further refine your search or use the available controls to change the search result presentation.

For New User Interface

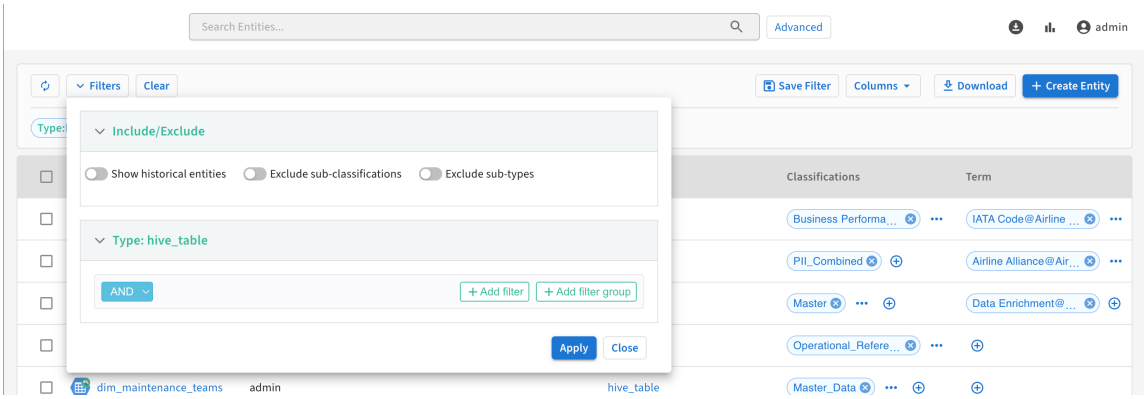
Figure 11: Search results



1. View he search criteria set in the **Search** panel.
2. Control which attributes of the entities are shown in the columns.
3. Download the selected search results as a CSV file.
4. By selecting the checkboxes, mark multiple entities for export or for applying classifications or terms.
5. Add classifications by clicking the ••• icon. By clicking the ✚ icon, view the rest of the classifications, that did not fit the screen.
6. Add terms by clicking the ••• icon. By clicking the ✚ icon, view the rest of the terms, that did not fit the screen.

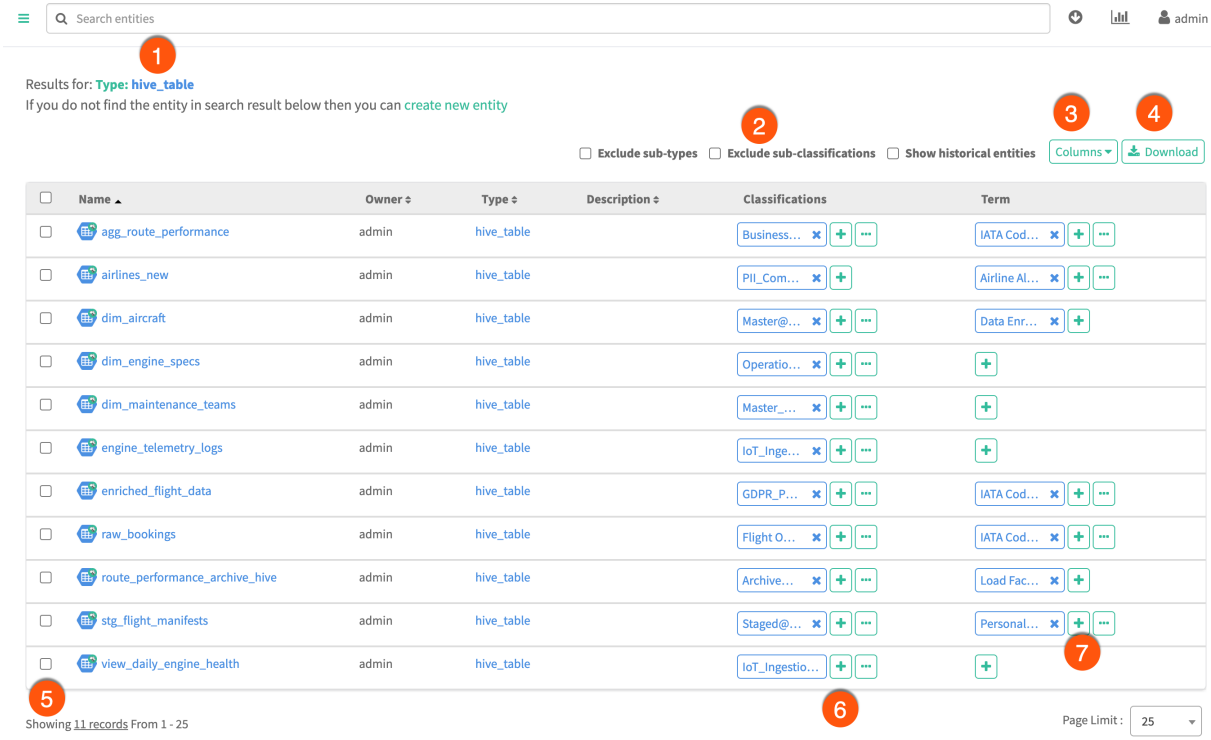
Note: The options to exclude some entity types from the result list is moved to the **Filters**.

Figure 12: Include or exclude entity types



For Classic User Interface

Figure 13: Search results



1. View the search criteria set in the **Search** panel.
2. By selecting the checkboxes, remove the following entity types from the result list:
   - Sub-entities
   - Sub-classifications
   - Entities marked as deleted (historical entities)
3. Control which attributes of the entities are shown in the columns.
4. Download the selected search results as a CSV file.
5. By selecting the checkboxes, mark multiple entities for export or for applying classifications or terms.

6. Add classifications by clicking the + icon. By clicking the ⋮ icon, view the rest of the classifications, that did not fit the screen.
7. Add terms by clicking the + icon. By clicking the ⋮ icon, view the rest of the terms, that did not fit the screen.

Viewing entity details

When you click a link for an entity in the search results, Atlas opens an entity detail page that includes the metadata collected for the entity. The detail page organizes the entity content in the following tabs:

Properties

The Properties tab includes the system metadata collected for this entity and any user-defined properties. It also contains a list of labels applied to the entity. Use values from any of the string data type properties to find this entity using free-text search.

Lineage

The Lineage tab displays a lineage graph for each entity. For data asset entities, the lineage graph displays when the entity functioned as an input or an output for an operation. For process entities, the lineage graph displays all input and output entities used by or produced by the operation.

Relationships

The Relationships tab lists the other entities associated with the current entity. You can view the related entities as a list or as a graph and use this tab to navigate among entities. The input and outp ut relationship types include the entities that make up the lineage.

Classifications

The Classifications tab displays the classifications associated with this entity and are also displayed in the top section of the detail page. This tab allows you to add, update, or remove classifications from the entity.

Note: Managing classifications requires specific privileges to perform classification actions.

Audits

The Audits tab records the changes that occur to entity metadata. It displays when Atlas updated the entity's metadata, including the following changes:

- Classifications added or removed
- Entity attributes updated
- Labels added, updated, or removed
- Relationships added, updated, or removed
- Glossary terms added or removed

Schema

When the current entity is a table, the Schema tab is displayed and lists columns in the table. Use this tab to drill into a specific column or to add classifications to columns without opening the individual column detail page.

Related Information

Using Basic search
Using Free-text Search
Using Search filters
Searching for entities using classifications
Glossaries overview

Apache Atlas metadata collection overview

Actions performed in cluster services create metadata in Atlas.

Atlas provides addons to many Hadoop cluster services to collect metadata when the service performs certain operations. The Atlas addon or “hook” assembles a predefined set of information and sends it to the Atlas server. The Atlas server reads through the metadata and creates entities to represent the data sets and processes described by the metadata. Atlas may create one or many entities for each event it processes. For example, when a user creates a namespace in HBase, Atlas creates a single entity to represent the new HBase namespace. When a user runs a query in HiveServer, Atlas may create many entities, including entities to describe the query itself, any tables involved in the query, entities for each column for each table involved in the query, and so on.

The following table lists the services that are integrated with Atlas by default. For each service, the table lists the events produced by the service that Atlas acknowledges and the entities Atlas produces in response to each event. Note that there isn’t always a one-to-one relationship between the event and an entity: the entities produced from a single event depend on the event itself.

Source	Actions Acknowledged	Entities Created/Updated
HiveServer	ALTER DATABASE CREATE DATABASE DROP DATABASE	hive_db, hive_db_ddl
	ALTER TABLE CREATE TABLE CREATE TABLE AS SELECT DROP TABLE	hive_process, hive_process_execution, hive_table, hive_table_ddl, hive_column, hive_column_lineage, hive_storagedesc, hdfs_path
	ALTER VIEW ALTERVIEW_AS_SELECT CREATE VIEW CREATE VIEW AS SELECT DROP VIEW	hive_process, hive_process_execution, hive_table, hive_column, hive_column_lineage, hive_table_ddl
	INSERT INTO (SELECT) INSERT OVERWRITE	hive_process, hive_process_execution
HBase	alter_async	hbase_namespace, hbase_table, hbase_column_family
	create_namespace alter_namespace drop_namespace	hbase_namespace
	create table alter table drop table drop_all tables	hbase_table, hbase_column_family
	alter table (create column family) alter table (alter column family) alter table (delete column family)	hbase_table, hbase_column_family

Source	Actions Acknowledged	Entities Created/Updated
Impala*	CREATETABLE_AS_SELECT	impala_process, impala_process_execution, impala_column_lineage, hive_db hive_table_ddl
	CREATEVIEW	impala_process, impala_process_execution, impala_column_lineage, hive_table_ddl
	ALTERVIEW_AS_SELECT	impala_process, impala_process_execution, impala_column_lineage, hive_table_ddl
	INSERT INTO INSERT OVERWRITE	impala_process, impala_process_execution
Spark*	CREATE TABLE USING CREATE TABLE AS SELECT, CREATE TABLE USING ... AS SELECT	spark_process
	CREATE VIEW AS SELECT,	spark_process
	INSERT INTO (SELECT), LOAD DATA [LOCAL] INPATH	spark_process

*For these sources, Atlas collects the corresponding asset metadata from HMS. Atlas reconciles the entity metadata received from Kafka messages from each source.

Related Information

Hive Server 2 metadata collection

HBase metadata collection

Impala metadata collection

Spark Metadata Collection

Atlas metadata model overview

Atlas' model represents cluster data assets and operations, and is flexible enough to let you represent objects from other sources.

The flexibility Atlas' metadata model lets you represent the objects and relationships among them so you can produce a map of your data lake. Atlas lets you create new instances of predefined entity types and lets you define new types of entities so you can represent data assets and actions from additional data sources or even services that do not reside in Hadoop. Atlas' building blocks are entities, relationships, classifications, enumerations, and structures.

Entities are a collection of attributes that model or represent a data asset or data action. Entities are the unit that Atlas returns in search results or shows as nodes in a lineage diagram. Labels are modeled as attributes on a given entity instance; you can add user-defined properties to individual entity instances (without affecting the entity type definition).

Relationships describe connections between two entities. You can create relationship definitions with custom attributes to represent behaviors that are specific to your processes. Changes to relationship definitions require changing the model through the Atlas API.

Classifications are named sets of key-value pairs that can be associated with entities. Classifications are distinct from entity attributes:

- Classifications are not part of entity metadata so they are a way to add metadata to entities without updating entity type definitions.
- Classifications can be added to any entity type.
- Atlas can propagate classifications through lineage relationships.
- Classifications can be used in Ranger to drive access policies.

Business Metadata are a set of custom attributes that an administrator can use to extend the metadata stored for each entity type. User access to each set of attributes can be managed using Ranger policies.

Atlas supports defining custom enumerations and data structures as well, similar to those constructs in structured programming languages. Enums can be used in attribute definitions to store lists of predetermined values; structs can be used in attribute definitions to identify more complex data types.