Implementation Guide

Automated Data Analytics on AWS

Automated Data Analytics on AWS: Implementation Guide

Table of Contents

Solution to ingest, transform, query, and manage datasets from multiple sources

Publication date: *August 2022* (last update: April 2024)

The Automated Data Analytics on AWS solution provides an end-to-end data platform for ingesting, transforming, managing and querying datasets. This helps analysts and business users manage and gain insights from data without deep technical experience using Amazon Web Services (AWS). It has an open-sourced architecture with connectors to commonly used AWS services, along with third-party data sources and services. This solution also provides an user interface (UI) to search, share, manage, and query datasets using standard SQL commands.

This solution is applicable for a variety of use-cases:

- Ingesting, transforming, and querying disparate datasets.
- Providing simplified access to data for non-technical business users.
- Managing organization-wide data governance and access policies.

This implementation guide describes the architectural considerations and configuration steps for deploying the Automated Data Analytics on AWS solution in the AWS Cloud. It includes links to an AWS CloudFormation template that launches, configures, and runs the AWS services required to deploy this solution using AWS best practices for security and availability. This implementation guide also includes instructions on how to use the solution web UI, and a developer guide for steps to deploy the CDK and use the ADA APIs.

Use this navigation table to quickly find answers to these questions:

If you want to . . .	Read . . .
Know the cost for running this solution. The estimated cost for running this solution in the Asia-Pacific (AP-Southeast-2) Region is **USD $1246.46** per month.	Cost
Understand the security considerations for this solution.	Security

If you want to . . .	Read . . .
Know how to plan for quotas for this solution. Use AWS Service Quotas in your AWS console to verify your AWS Lambda concurrent executions	Quotas
Know which AWS Regions are supported for this solution.	Supported AWS Regions
View or download the AWS CloudForm ation template included in this solution to automatically deploy the infrastructure resources (the "stack") for this solution.	AWS CloudFormation template
View details about the features of the solution web UI and how to use them for working with datasets.	Using the ADA solution
Know how to deploy the solution using CDK, and how to use the ADA APIs.	Developer guide

The guide is intended for IT architects, developers, engineers, and technology professionals who have practical experience architecting in the AWS Cloud.

Features and benefits

The Automated Data Analytics on AWS solution provides the following features:

Data lifecyle management

- Automates the process of ingesting, transforming, governing, and querying diverse datasets.

Pre-built connectors

- Wizard-based connectors for Amazon S3, Amazon Kinesis Stream, Amazon CloudWatch, Amazon CloudTrail, File Upload, Google Cloud Storage, Google Analytics, Google BigQuery, MySQL5, PostgreSQL, Microsoft SQL Server, DynamoDB, and MongoDB.

Self-serve analytics

- Anyone with SQL skills can quickly and easily derive insights from their data.

For more information, refer to the Automated Data Analytics on AWS solution page.

Use cases

Simplify Data Analytics at scale

Automated Data Analytics on AWS simplifies the management and analysis of data, providing an end-to-end platform used for ingesting, transforming, governing, and querying datasets, from a range of different data sources, using an intuitive standalone user interface. This enables analysts to easily manage and gain insights from data without any deep technical experience using the AWS platform.

Software developers with a basic knowledge of AWS products and services can set up a data platform for a team of analysts in a few hours, and easily and securely manage the data access permissions as well as the PII data.

Concepts and definitions

This section describes key concepts and defines terminology specific to this solution:

Dataset

A *dataset* is a singular collection of data, such as a database table.

Data product

A *Data product* is a dataset that has successfully been imported into Automated Data Analytics on AWS and is ready to be queried.

Query workbench

Mechanism to run SQL-like queries on data products that have been successfully imported.

Domain

A *domain* is a user defined collection of data products. For example, this might be a team or a project. Domains provide a structured way for users to organize their data products and manage access permissions.

Schema transform

A *transform* is the process of converting data from one schema format to another.

Governance attribute

Governance attributes are specific business classifications for data fields. Governance attributes could be either automatically detected PII data types or custom defined types. You can assign user group level access to each individual governance attribute.

Personally identifiable information (PII)

PII is a textual reference to personal data that could be used to identify an individual. PII examples include addresses, bank account numbers, and phone numbers.

For a general reference of AWS terms, see the AWS glossary in the AWS General Reference.

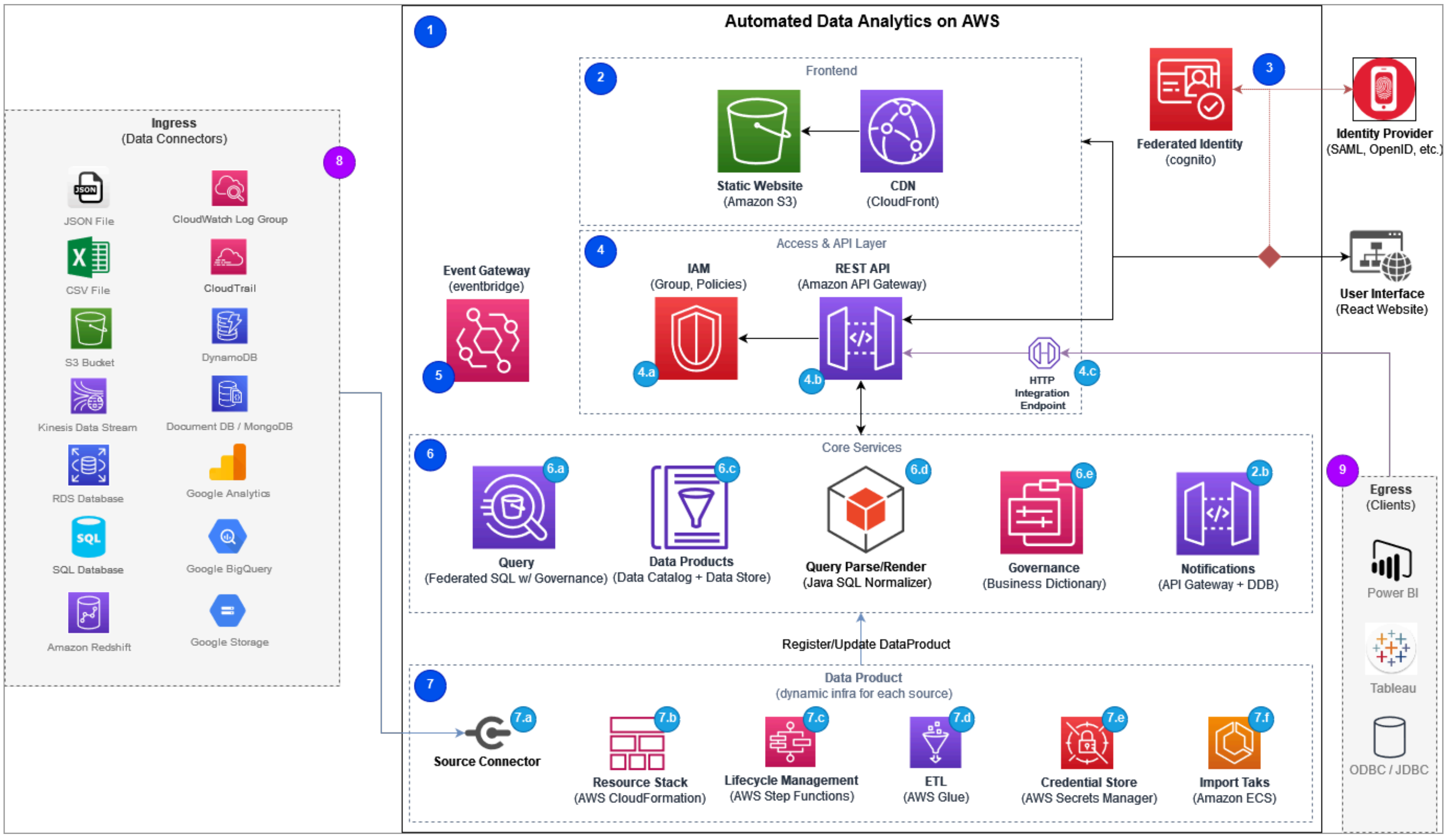
Architecture overview

The Automated Data Analytics on AWS solution automates the building of data pipelines that are optimized for the size, frequency of update, and type of data. These data pipelines handle the data ingestion, transformations, and queries.

This solution creates and integrates a combination of AWS services required to perform these tasks, abstracted through a user interface. These services include AWS Glue crawlers, jobs, workflows and triggers, along with S3 buckets, IAM integration, and other services. For querying data, Athena and SparkSQL Glue jobs are integrated into Automated Data Analytics on AWS as the engine to allow SQL queries and transformations.

Deploying this solution with the default parameters builds the following environment in your AWS account.

Architecture diagram



Automated Data Analytics on AWS architecture diagram

> **Note**
>
> AWS CloudFormation resources are created from AWS Cloud Development Kit (AWS CDK) constructs.

The high-level process flow for the solution components deployed with the AWS CloudFormation template is as follows. The numbers below match the number designated in the architecture diagram.

1. The AWS CloudFormation template provisions the following infrastructure and services provided by the solution.

   > **Note**
   >
   > Other resources in the diagram are example integrations supported by the deployed solution, but are not provisioned or managed by the solution; these external resources are for reference only. In addition to the following resources, all Amazon DynamoDB tables and Amazon Simple Storage Service (Amazon S3) buckets have unique AWS KMS keys for encryption.

2. Frontend
   - Static website**:** The solution uses Amazon CloudFront for distribution, and is protected by AWS WAF. It also uses an Amazon S3 bucket to host and serve the web front end, including the HTML pages, CSS stylesheets, and JavaScript code.
   - Notifications**:** An Amazon DynamoDB table is used to manage and provide persistent notifications in the user interface, along with Amazon API Gateway REST API resources (resource, method, model), AWS Lambda handler (NodeJS), and an Amazon EventBridge rule for mapping events to notifications.
3. Federated Identity**:** An Amazon Cognito user pool manages federating and storing users from external identity providers (IDPs). An AWS Lambda function uses NodeJS to handle custom authorization of federated authentication from identity providers and maps federated users to Automated Data Analytics on AWS's Identity and Access Management model. The solution provisions a *root administrator* user in the Amazon Cognito user pool based on email and phone number provided in the CloudFormation template parameters:
   - The *root administrator* is the only user directly managed by the user pool, and

- All other users are federated through the IDP.

4. Access and API Layer
    - Identity and Access Management (IAM):
        - An Amazon DynamoDB table to store group policy statements.
        - An Amazon Cognito user pool for managing federated user authentication.
    - Amazon API Gateway (REST API) for federating requests and access to all underlying services and resources.
        - AWS Lambda as custom authorizer to control API access for federated users and machines.
        - An Amazon CloudFront distribution network to cache and AWS WAF to protect the API.
    - Amazon API Gateway (HTTP API) is used for proxying egress requests from external clients (for example, Tableau, or PowerBI) via an Amazon Cognito client credentials and facilitating the request and response to support client formats.
5. Event Gateway
    - A dedicated Amazon EventBridge event bus and gateway is used for event-driven application messaging between microservices, and to propagate and persist notifications to users.
6. Core solution services
    - Data Product service: Stores details about data products and manages the creation of dynamic infrastructure used to ingest, transform, and manage various data sources.
        - AWS Lambda functions for handling API requests (NodeJS).
        - Amazon DynamoDB data stores.
        - AWS Step Functions for managing the lifecycle of data products.
        - Amazon S3 buckets for storing processed data, user-defined scripts, and file uploads.
        - AWS Glue tables and resources for handling the data extract, transform, and load (ETL) processing.
        - AWS Lambda function utilizing AWS CDK and CloudFormation to deploy *dynamic infrastructure* for each data product (NodeJS).
    - Query service**:** Responsible for executing *governed* federated queries, storing/managing saved queries, and maintaining query caching.
        - AWS Lambda functions for handling API requests (NodeJS).
        - Amazon Athena for performing federated queries which stores results in Amazon S3 buckets.
        - AWS Step Functions to orchestrate the asynchronous life-cycle of query execution.

  - Amazon DynamoDB data stores for saved queries, query history, and query caching.
- Query Parse/Render service**:** Responsible for SQL query manipulation. This is decoupled from query service to provide flexibility in SQL parsing library.
  - AWS Lambda functions for handling API requests (NodeJS & Java).
- Governance service: Allows you to define common terms or classifications of data throughout the entire business and define governance policies based on user groups.
  - AWS Lambda functions for handling API requests (NodeJS).
  - Amazon DynamoDB data stores for storing governance metadata.

7. Dynamic Infrastructure: Each data product deploys a dedicated AWS CloudFormation stack with varying resources depending on source type and data, along with Amazon EventBridge rules for integration with core services.
   - AWS Lambda functions for handling source import.
   - AWS CloudFormation stack to manage resources.
   - AWS Step Functions for orchestrating lifecycle management.
   - AWS Glue crawlers, data catalogues, and jobs for ETL.
   - AWS Secrets Manager to store external credentials.
   - Amazon ECS tasks for processing large data ingestion jobs.
   - Amazon Athena and Amazon Comprehend for detecting PII entities.
8. Ingress (Data Connectors): Automated Data Analytics on AWS supports multiple source data connectors out-of-the-box including Amazon S3, Amazon Kinesis Stream, Amazon CloudWatch, Amazon CloudTrail, Amazon Redshift, File Upload, Google Cloud Storage, Google Analytics, Google BigQuery, MySQL5, Oracle, PostgreSQL, Microsoft SQL Server, DynamoDB, or MongoDB. The resources required to support these data sources are only deployed during the creation of a new data product of the given type; there are no idle resources.
9. Egress (Third Party Tools): Automated Data Analytics on AWS support both JDBC and ODBC standards for consuming data from common clients.

> **Note**
>
> AWS CloudFormation resources are created from AWS Cloud Development Kit (AWS CDK) constructs.

AWS Well-Architected design considerations

We designed this solution with best practices from the AWS Well-Architected Framework, which helps customers design and operate reliable, secure, efficient, and cost-effective workloads in the cloud. The Well-Architected Framework includes six pillars plus domain-specific lenses, hands-on labs, and the AWS Well-Architected Tool. (available at no charge in the AWS Management Console).

This section describes how we applied the design principles and best practices of the Well-Architected Framework when building this solution.

Operational excellence

This section describes how we architected this solution using the principles and best practices of the operational excellence pillar.

- All Lambda functions send logging output to Amazon CloudWatch.
- Access to S3 buckets and CloudFront Distribution is logged into a dedicated access log bucket.
- A comprehensive CloudWatch dashboard is provided to monitor the operational status of underlying services.
- Amazon Step Functions past runtime outcomes can be reviewed.

Security

This section describes how we architected this solution using the principles and best practices of the security pillar.

- All inter-service communications use AWS Identity and Access Management (IAM) roles.
- Users are authenticated with OIDC code flow with signed JWT token as credentials.
- All roles used by the solution follow least-privilege access. That is, they only contain the minimum permissions required so that the service can function properly.
- All S3 buckets have encryption at REST activated with custom keys from AWS KMS.
- Access to data storage buckets is logged into a dedicated access log bucket.
- AWS WAF is applied on both CloudFront Distribution as well as APIGateway APIs to mitigate potential attack.
- All sensitive information is managed in AWS Secrets Manager.

Reliability

This section describes how we architected this solution using the principles and best practices of the reliability pillar.

- The solution uses AWS serverless services wherever possible to ensure high availability and recovery from service failure.
- All compute processing uses Lambda functions or ECS Fargate.
- Data is stored in Amazon S3 and DynamoDB tables, so it persists in multiple Availability Zones by default.
- The solution uses Step Functions to orchestrate the workflow and provide retries for steps in the workflow.
- All custom code uses AWS Software Development Kit (AWS SDK) and benefits from automatic retries and back-off for application programming interface (API) calls..

Performance efficiency

This section describes how we architected this solution using the principles and best practices of the performance efficiency pillar.

- The solution uses serverless compute and data resources throughout the architecture.
- You can launch the solution in any Region that supports AWS services used in this solution.
- The solution is developed with AWS CDK and managed with AWS CloudFormation stacks. By implementing complete Infrastructure as Code approach, it allows easy upgrading and resources management.
- The solution leverages as many AWS managed services as possible including AWS Glue, AWS Athena, S3, AWS Comprehend, and AWS Cognito.

Cost optimization

This section describes how we architected this solution using the principles and best practices of the cost optimization pillar.

- Only Lambda functions and AWS ECS Fargate are used for compute needs and only charged for what is used.

- Full serverless architecture and automatic scalability to scale out when demand is high and scale in when demand is low.
- Application provides built-in Cost Explorer for customer to easily monitor costs from the application itself.

Sustainability

This section describes how we architected this solution using the principles and best practices of the sustainability pillar.

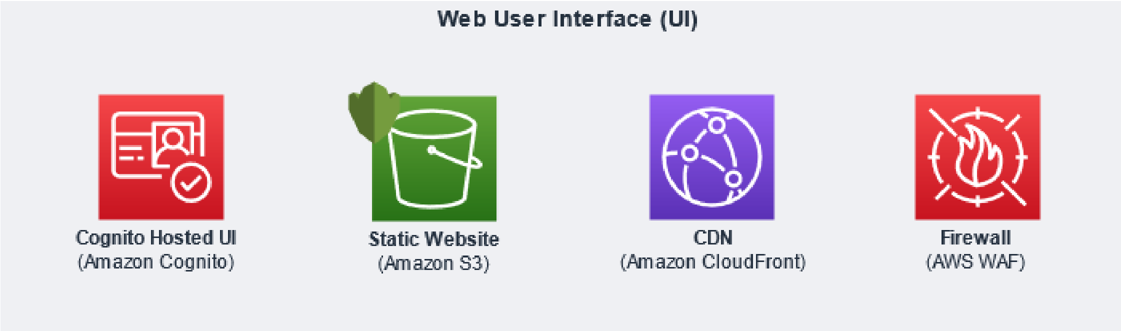
- Serverless resources are used for compute and data storage.
- Most data storage is maintained in a S3 bucket that you can remove easily.

Architecture details

This section describes the components and AWS services that make up this solution and the architecture details on how these components work together.

- Web User Interface (UI)
- Notification and events
- Identity and Access Management
- REST API
- HTTP API
- Data Product service
- Query service
- Governance service
- Cost Explorer service

Web user interface (UI)

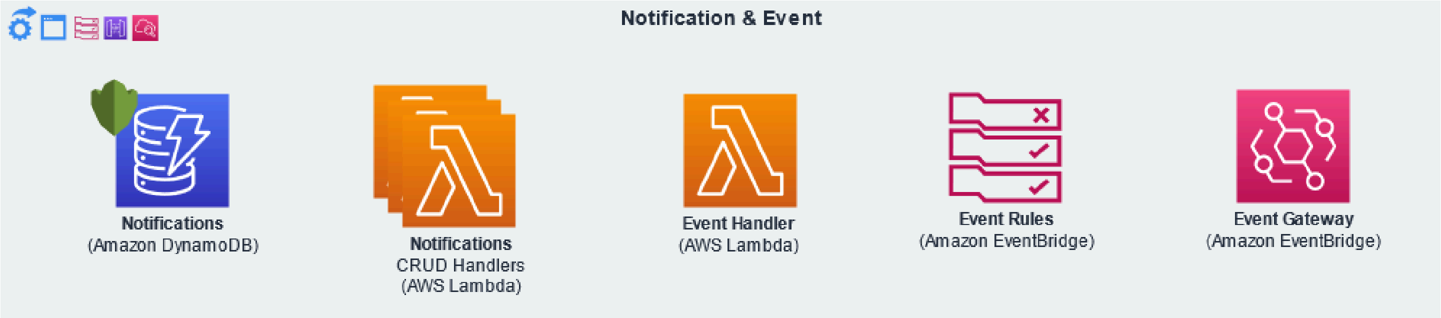


Web UI components

This solution provisions a user interface (UI) to ingest, transform, query, and manage datasets by deploying a static single page web application in Amazon S3 fronted by an Amazon CloudFront distribution secured through AWS WAF.

An Amazon Cognito hosted UI handles the user sign-in, but does not support user sign-up flows.

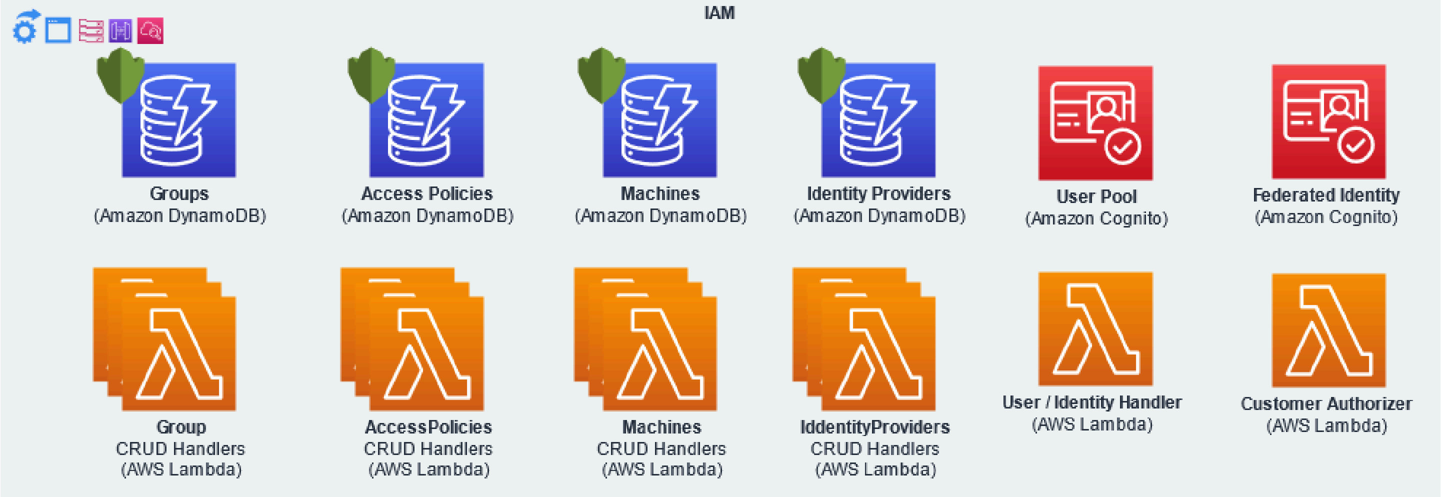
Notification and events



Notifications and events components

Automated Data Analytics on AWS follows event-driven patterns for decoupling service-to-service integration and notifying the end user about important events. This solution provisions an Amazon EventBridge event bus that is shared between all micro-services and used to publish and subscribe to asynchronous events. Each micro-service provisions its own event rules to receive events across service boundaries. Important end-user events are filtered from the event-bus and propagated to a persistent notification store in Amazon DynamoDB to help the end user to receive and acknowledge events between sessions.

Identity and Access Management (IAM)



Identity and Access Management components

User management

When you install this solution for the first time, it creates a *root_admin* user in Amazon Cognito User Pool based on the **adminEmail** parameter provided and Amazon Cognito User Pool sends a temporary password to the specified email address. Using these credentials, the admin user can sign in to the web UI at the deployed endpoint. Once signed in to the UI, the admin user can configure the solution to integrate with external Identity Providers (IdP) to allow additional users access through federated sign-in. With the exception of the *root_admin*, all users within Automated Data Analytics on AWS are federated users managed through their respective IdP.

Groups

Automated Data Analytics on AWS uses the concept of groups to manage user access to the API, resource entities, and queried data across a collection of users. Upon deployment, the solution provisions three system groups: **Default Users, Power Users**, and **Administrators**. The system groups are initially provisioned as read-only, read-write, and full access, respectively, with full access enabling additional functionality such as managing IdP, generating cost reports, etc.

Power Users and Administrators can manage group members and create/manage additional groups; the system groups are required by the solution and cannot be deleted. Groups maintain an explicit list of members that belong to the group, and users can belong to multiple groups. Users are granted membership to a group either through request/approval flow initiated by the user in the UI, or by the group owner/administrator adding the user to the group through the UI. Groups are managed in the backend through Amazon DynamoDB.

The Default Users group also supports automatic membership for all authenticated users to simplify providing basic access for all federated users. This is disabled by default and can be configured through the Default Users details page in the UI.

Authentication mechanism

Automated Data Analytics on AWS uses an [Amazon Cognito User Pool](#) for authentication through the web user interface (UI) and Amazon API Gateways (REST and HTTP). The solution supports federated user authentication through an Identity Provider (IdP) and client credentials for machine-to-machine access (for example, Tableau, PowerBI).

> **Note**
>
> The Automated Data Analytics on AWS solution only supports OpenID Connect and SAML2.0 authentication.

After you configure the IdP, additional users can sign in through the UI using federated sign-in, utilizing the OpenID Connect code flow through Amazon Cognito Hosted-UI. Once authenticated, Amazon Cognito provides a JSON Web Token (JWT) to the web UI that is provided with all subsequent API requests. If a valid JWT is not provided, the API request will fail and return a HTTP 403 Forbidden response.

Using the web UI, federated users can create access for clients (For example, Tableau, PowerBI), which provisions an Amazon Cognito app client that can be used for machine-to-machine access using the *client credentials* flow. Once authenticated, Amazon Cognito provides a JSON Web Token (JWT) to the client that will be provided with all subsequent API requests. In case the access is provided through the Athena Proxy endpoint (CloudFront endpoint for ODBC connections), the user will not be required to call Amazon Cognito to retrieve a valid JWT access token, since this action will be performed internally by Athena Proxy. If a valid JWT is not provided, the API request will fail and return a HTTP 403 Forbidden response.

Authorization and data access mechanism

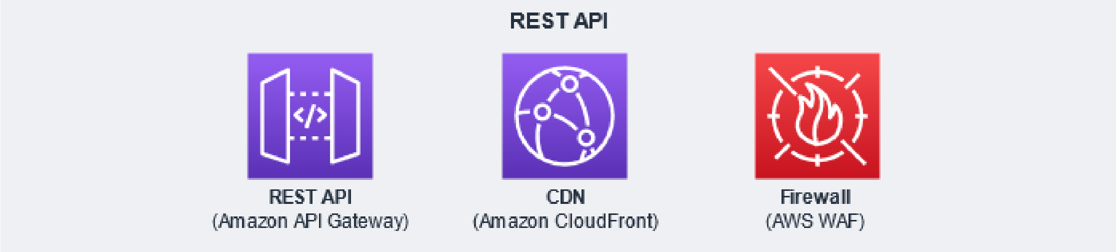
Automated Data Analytics on AWS uses several layers of authorization to ensure least-privileged permissions are applied throughout the solution and provide granular access controls of the API, entities, and data.

Federated user authorization is based on group membership and the respective group policies determined during authentication flow. External machines (clients) authorization is based on Amazon Cognito *client credentials* that are associated with the federated user that created the client.

Service-to-service calls originating within the solution are authorized based on providing encrypted shared API key managed by AWS KMS in request header. High-level authorization is managed through groups that define a collection of least-privileged Amazon IAM policy statements stored in Amazon DynamoDB. A user can belong to multiple groups and their authorization is based on the merging of all group policy statements that the user belongs to. All group policy statements are allow based only and grant the user access to API routes. By default, users not belonging to any group are denied all access, with the exception of the routes for retrieving their own user profile and permissions. All API routes require the user to be authenticated, with unauthenticated users receiving a HTTP 403 Forbidden response.

Entity level access is managed through permission tables stored in DynamoDB. Access is mapped from entity to group to access to the read-only, read-write, and full permissions, with non-existing record resulting in no access. For instance, there is a DynamoDB table for data product permissions that defines the access policy to apply to each data product entity per group. Entity level authorization is enforced by the respective AWS Lambda handler resolving the API request to validate the caller has adequate permissions to perform the requested action against the requested entity. If the caller's access is insufficient to perform the action, the API request will fail and return a HTTP 403 Forbidden response.

REST API

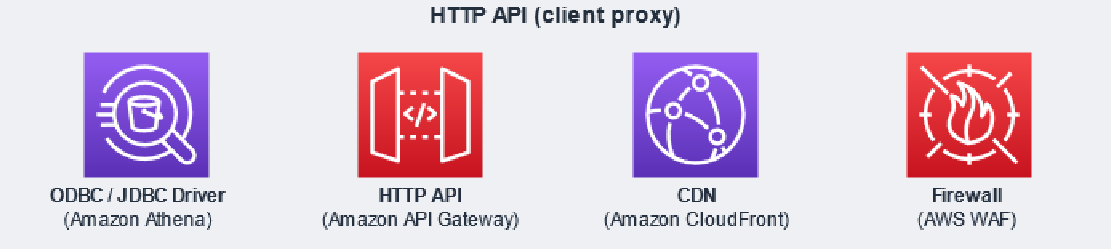


REST API components

Automated Data Analytics on AWS provisions a single Amazon API Gateway REST API to handle requests to underlying data provided by the decoupled micro-services. The REST API is fronted with an Amazon CloudFront distribution and further secured by AWS WAF. Each micro-service provisions Amazon API Gateway Resources, Methods, Models, and Integrations within its service scope into the single Amazon API Gateway REST API.

The REST API requires that all requests include Authorization headers with authenticated user credentials, no unauthorized access is allowed as described above.

HTTP API (client proxy)

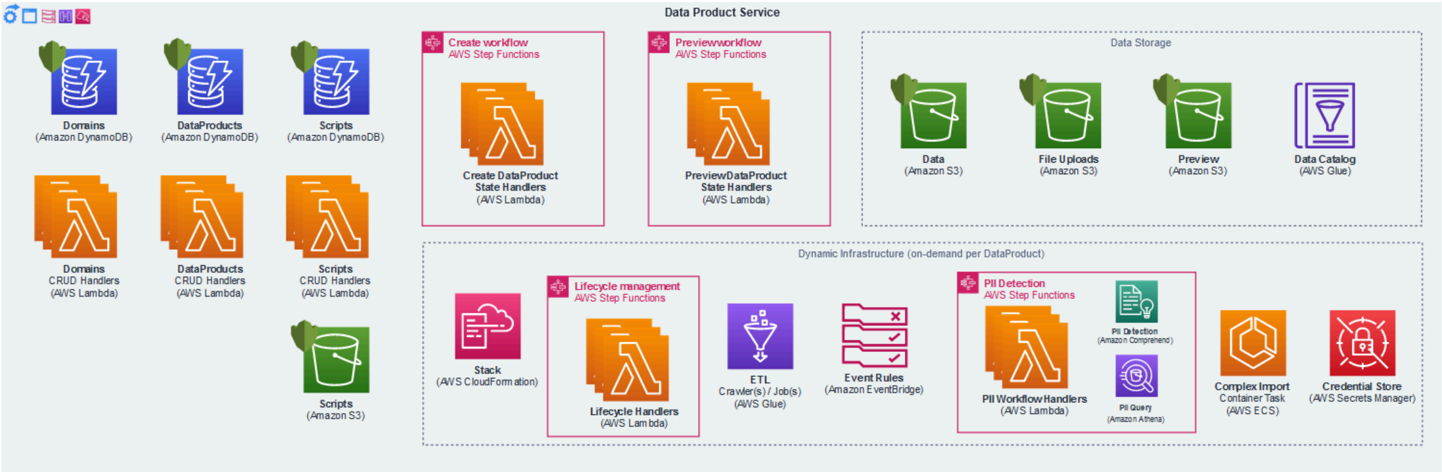


HTTP API components

Automated Data Analytics on AWS provisions an Amazon API Gateway HTTP API to provide egress access to underlying data and governed query execution capabilities for external clients (For example, Tableau, PowerBI). The HTTP API is fronted with Amazon CloudFront distribution and secured with AWS WAF.

This client proxy endpoint enables using the Amazon Athena ODBC / JDBC Drivers to securely query data within Automated Data Analytics on AWS using OAuth *client credential* flow and executing governed queries based on federated user authorization. See Data Egress below for more details.

Data Product service



Data product service components

At the center of this solution is the concept of a data product, which is a logical grouping of a single dataset or multiple relational datasets, all the relevant metadata for the datasets, and the provisional state of the individual resources. The Data Product service manages all the metadata, lifecycles, and integration with other services in the context of the data entities. Metadata is stored in Amazon DynamoDB tables, encrypted with an individual AWS KMS key, and handled through individual AWS Lambda functions for CRUD operations. The service also provisions two AWS Step Functions to handle complex orchestration of the asynchronous lifecycle of data products and schema preview.

This solution provides a mechanism for grouping Data Products by domain; each data product must belong to a single domain. Domains can be used to group data products by business organization, projects, teams, etc.

Customer data managed through this solution is stored in a single Amazon S3 bucket that is partitioned by folders at the data product level. Data files and custom transform scripts directly uploaded through the UI are stored in two separate Amazon S3 buckets. Preview results, including inferred schema and sample data, are stored in a separate Amazon S3 bucket. All Amazon S3 buckets within this solution are encrypted with individual AWS KMS keys.

This solution uses AWS Glue crawlers to extract the schema from datasets and stores this metadata in a shared AWS Glue data catalog. The transformed and processed results are stored in the shared data bucket mentioned above.

Dynamic infrastructure

The above resources are considered *static infrastructure* within the solution as they are provisioned during initial deployment. In addition to this *static infrastructure*, this solution also provisions what it considers *dynamic infrastructure*, which are resources that are provisioned on-demand after the initial deployment. The Data Product service provisions and de-provisions these additional resources during the creation and deletion processes of individual data products, which are managed through the deployment of individual AWS CloudFormation stacks that manage the additional resources necessary for the given type of data product.

Each data product stack includes an AWS Step Function and relevant AWS Lambda functions for managing the lifecycle of the data product and related resources. Depending on the underlying dataset being managed, either a single or multiple AWS Glue crawlers and jobs are provisioned to handle data ETL. Each data product also registers a number of Amazon EventBridge rules to handle event-based integration with the rest of the solution.

In some cases, sensitive credentials are required to access the external data source. When the data product requires storing credentials, it will provision a secret in AWS Secrets Manager to securely store and manage credentials.

Most data import tasks are handled within asynchronous AWS Lambda functions and AWS Glue crawler workflows, but in some cases data import requires custom handling and the solution provisions AWS ECS tasks to handle these complex imports workflows.

Automatic PII detection

Automated Data Analytics on AWS provides automatic PII detection and governance classification during data imports using Amazon Comprehend text analysis for PII entities. When enabled on a data product, it will also provision an AWS Step Function and relevant AWS Lambda functions for detecting PII entities through Amazon Comprehend DetectPiiEntities API that is executed through an Amazon Athena UDF. The PII entity results from Amazon Comprehend are mapped to governance attributes within the solution governance layer to enable automatic PII governance.

Transforms

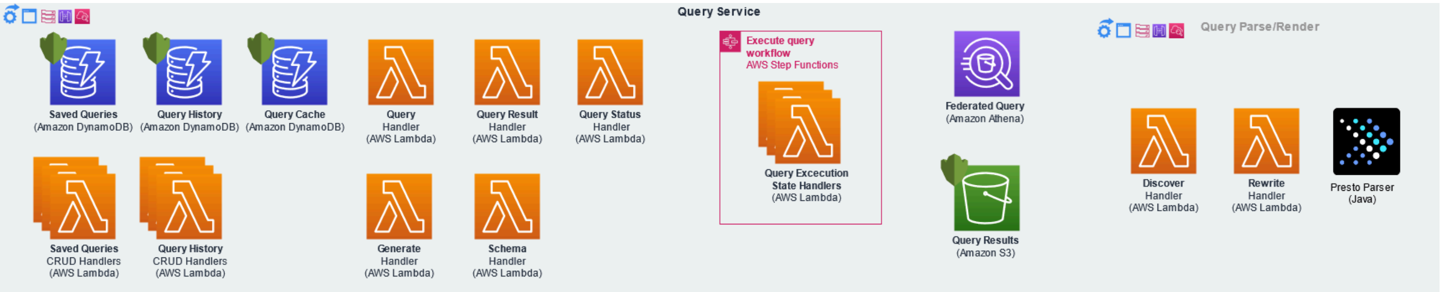
Automated Data Analytics on AWS provides the ability to transform data or its schema through transform scripts. The solution suggests transforms to apply based on the data provided (for example, JSON data is suggested to be relationalized to run SQL queries). Transform scripts are written as a single Python function, and are executed as AWS Glue Jobs during data import. At a high level, transform scripts define an `apply_transform` method which accepts an AWS Glue Dynamic Frame, and returns one or more AWS Glue Dynamic Frames as output. The below details the inputs available for a transform script (passed as kwargs so only those used need be declared):

- `input_frame`: The AWS Glue Dynamic Frame to transform. This could be the data from the data product's source, or the result of a previous transform.
- `glue_context`: The AWS Glue Context object
- `spark_context`: The Apache Spark Context object
- `temp_s3_path`: An S3 path (of the form s3://bucket/key) where the script is permitted to write temporary files if necessary
- `transformation_ctx`: A unique string for the current transformation, that can be prepended to any transformation_ctx parameters required for performing AWS Glue transformations.

The following example demonstrates a simple transform which adds a prefix to column names using AWS Glue's built in ApplyMapping transform.

```
from awsglue.transforms import ApplyMapping
    """
    Sample script to add a prefix of 'aws_' to all column names
    """
    def apply_transform(input_frame, transformation_ctx, **kwargs):
    mappings = [(
    "`{}`".format(field.name),
    field.dataType.jsonValue()["dataType"],
    "`aws_{}`".format(field.name),
    field.dataType.jsonValue()["dataType"]
    ) for field in input_frame.schema()]
    return [ApplyMapping.apply(
    frame=input.frame,
    mappings=mappings,
    transformation_ctx=transformation_ctx,
    )]
```

Query service


Query Service
Saved Queries (Amazon DynamoDB)
Query History (Amazon DynamoDB)
Query Cache (Amazon DynamoDB)
Query Handler (AWS Lambda)
Query Result Handler (AWS Lambda)
Query Status Handler (AWS Lambda)
Saved Queries CRUD Handlers (AWS Lambda)
Query History CRUD Handlers (AWS Lambda)
Generate Handler (AWS Lambda)
Schema Handler (AWS Lambda)
Execute query workflow
AWS Step Functions
Query Excecution State Handlers (AWS Lambda)
Federated Query (Amazon Athena)
Query Results (Amazon S3)
Query Parse/Render
Discover Handler (AWS Lambda)
Rewrite Handler (AWS Lambda)
Presto Parser (Java)

Query service components

Automated Data Analytics on AWS provides an abstraction around Amazon Athena federated queries to simplify complex SQL queries across disparate data sources with managed governance rules. The Query service manages all the relevant metadata for queries, including saved queries, query history, and query caching that enhance the user experience and performance. This service provisions multiple Amazon DynamoDB tables, encrypted with individual AWS KMS keys, and relevant AWS Lambda functions for CRUD operations, along with AWS Step Functions and AWS Lambda functions to manage the asynchronous lifecycle of query runs.

Query runs are asynchronous. When a query is run against the API, or through the UI, the SQL query is initially parsed, de-referenced, and governance is applied before sending the query to Amazon Athena to federate against the underlying resources. The query run results are stored in an Amazon S3 bucket that is provisioned by this solution, which also acts as the storage for caching query results. Queries are cached based on the hashing of the governed query being executed, providing users with the same level of access to all underlying data to retrieve the results from the cache when available and not stale. A query's run results are considered stale if the related data products have imported new data since the cached query was run.

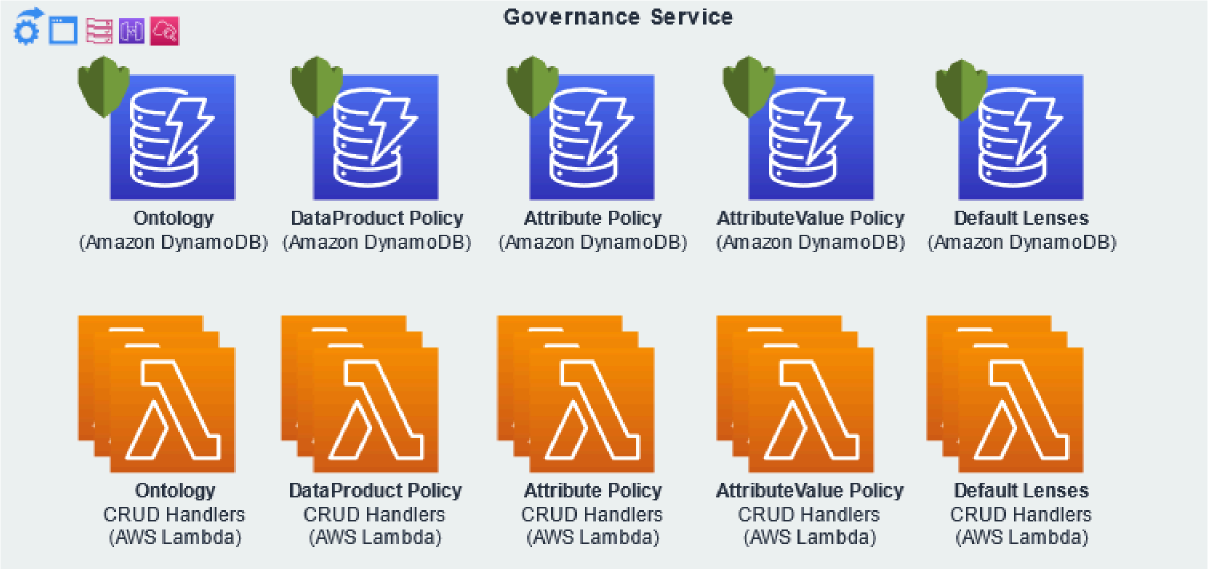
End-users can save queries either privately or with others. Saved queries are partitioned by namespaces in an Amazon DynamoDB table, with each unique user receiving their own namespace for private queries, while shared queries are associated with a domain and are available to all users.

Query Parse/Render service

Automated Data Analytics on AWS decouples the parsing of SQL queries from the rest of query execution to enable updating/replacing the underlying SQL parser that is used. This solution uses the Java implementation of Presto SQL Parser that is used by Amazon Athena and therefore supports the same querying and data formats.

In addition to parsing and de-referencing the query, the parse/render service is responsible for applying the correct governance to the query based on the access of the requesting caller.

Governance service


Governance Service
Ontology (Amazon DynamoDB)
DataProduct Policy (Amazon DynamoDB)
Attribute Policy (Amazon DynamoDB)
AttributeValue Policy (Amazon DynamoDB)
Default Lenses (Amazon DynamoDB)
Ontology CRUD Handlers (AWS Lambda)
DataProduct Policy CRUD Handlers (AWS Lambda)
Attribute Policy CRUD Handlers (AWS Lambda)
AttributeValue Policy CRUD Handlers (AWS Lambda)
Default Lenses CRUD Handlers (AWS Lambda)

Governance service components

Automated Data Analytics on AWS provides a single pane to manage governance across all data through a unified dictionary (ontology). This is used to define common terms or classifications of data throughout the entire business and define governance policies based on user groups. For example, there is a single term for ‘email’ classification, but a separate policy for each group’s’ access to data classified as “email”. As such, a user belonging to a group with the “redacted” governance policy defined for emails will see all emails in redacted form regardless of where the data is sourced. A user’s data access policy is derived from merging all the groups that a member belongs to and granting the user the most permissive access amongst their associated group policies for each business term in their requested query.

The solution supports column-level, row-level, and entity-level policies, along with default policies to be applied when no matching policy definition is found. The concept of business terms within the solution are called Governance attributes. Attribute Policies define column-level governance, Attribute Value Policies define row-level governance, and data product policies define entity-level access to a given data product. Governance metadata is stored in across Amazon DynamoDB tables, encrypted with individual AWS KMS keys, and related AWS Lambda functions for CRUD operations.

Attribute Policies (column-level) define mappings of Governance attributes (business terms) to lenses by user group, where a lens is an Amazon Athena User-Defined Function (UDF) with specific data handling purpose in context of governance. The solution provisions three UDF functions in the form of governance lenses: `CLEAR, HASHED,` and `HIDDEN`.

- `CLEAR` text lens acts as a pass-through lens in which the raw value is unmodified.
- `HASHED` text lens creates a deterministic tokenization of the value that is consistent between queries and sources to maintain relationships; the hashing includes a unique salt per account to prevent rainbow table brute-force attacks.
- `HIDDEN` text lens drops the entire column from results. For example, there could be a definition for Group A users to see all emails as hashed text, which would apply to all queries across all data.

Attribute Value policies (row-level) define mappings of attributes (business terms) to a SQL statement by user group. The SQL statement is used as a filtering mechanism that when matched, will drop the matching row. As an example, an attribute value policy could be defined to filter out all 'premium' user data for all groups except the premium customer support group.

Data product policies (entity-level) define basic access control against data products by user group with Read, Read/Write, and Full access support.

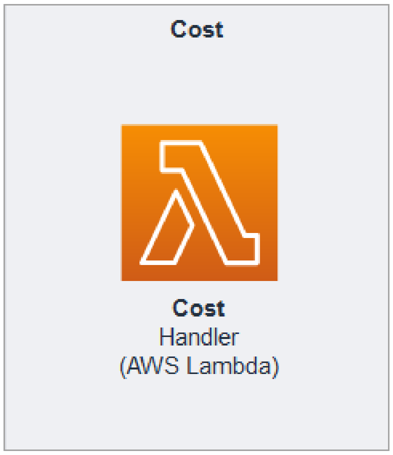
For more information on how to set up governance, refer to the Setting up governance section.

By default, when a data product is created, it has the following access policy defined; `Administrators = Full`.

- To view and query a data product, a user must have at least a `Read` level access.
- To isolate very sensitive or confidential data, you can create a custom user group with an explicit list of members that should have access, along with adding this group to the data product permissions and removing all other group access will render the data product completely hidden except from members of this custom group.

For more information, refer to the Data Product permissions section.

Cost Explorer service



Cost explorer service components

Automated Data Analytics on AWS integrates with the AWS Cost Explorer to provide a service-level breakdown of the utility spend across all resources provisioned by the solution. This feature is available to members of the administrator group through the UI.

To activate this feature, follow the instructions listed in the Cost Explorer section.

Data handling

Data ingress

Automated Data Analytics on AWS has an open-sourced architecture with connectors to commonly used AWS services and 3rd-party data sources that enable importing and transforming data through a wizard-based UI. Once data has been imported into the solution, it can be further governed and queried through the unified UI. One of the core principles of the solution is to not grant itself access to data it does not provision, as this could circumvent existing access policies. As such, it is the responsibility of the data owner to provide the necessary access prior to importing data. Except for the most basic data examples, such as importing existing S3 data with no transformations being applied, the solution will first import the source data into Amazon S3 before performing ETL.

Note

This solution supports Kinesis Data Stream as a data source input. Amazon Kinesis Data Streams is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Kinesis Data Streams. For more information on how to monitor API calls to Kinesis Data Stream, refer to the Logging using CloudTrail topic.

Most connectors support fast schema inference and preview during creation to allow managing and querying the data as quickly as possible, however this is not always possible depending on the source type. In cases where not supported, the full dataset must be crawled and ETL completed before querying or managing dataset is available. See the following table for details on which source types support preview. Some source types also support querying the original source data directly, which provides quick federated query access to raw data prior to completing the full import. However, this functionality is only available to the creator of the data product. See the table below for details on which source types support source query.

Connector support

Automated Data Analytics on AWS currently supports the following connectors:

Note

AWS will periodically build and release connectors for additional data sources and integrate connectors built by the open source community through GitHub.

Connector	Description	Preview*	Source Query**
Amazon Kinesis Data Stream	Source data from an existing Kinesis DataStream within the same account in the same region. If data stream is encrypted using custom managed key, decrypt access is required for the Automated Data Analytics application.		

Connector	Description	Preview*	Source Query**
Amazon CloudWatch	Source data from Amazon CloudWatch logs. The connector queries CloudWatch Logs Groups with the specific CloudWatch query and then imports the result logs into Automated Data Analytics on AWS. It also supports incremental importing by schedule.	X	X
Amazon Redshift	Source data from an existing Amazon Redshift Cluster or Amazon Redshift Serverless table in an AWS account.		
Amazon S3	Source data from existing Amazon S3 data supports the same data file types and formats as AWS Glue. The provided object path must be readable by the Automated Data Analytics on AWS application.	X	X

Connector	Description	Preview*	Source Query**
AWS CloudTrail	Source data from AWS CloudTrail. It supports filtering on importing for different CloudTrail Event Types and also supports importing from a different account. The connector also supports incremental daily import when it is set to scheduled mode. Due to data volumes that could be potentially massive, the automatic PII detection feature is currently unavailable for CloudTrail connector.	X	X

Connector	Description	Preview*	Source Query**
DynamoDB	Source data from AWS DynamoDB tables. It supports importing data from DynamoDB tables from a different account. The data is transferred into a S3 bucket that is managed by Automated Data Analytics for AWS.	X	X
File Upload	Source data from file upload supports .csv, .json, .par and .gz file formats through the UI. The uploaded files are stored in a shared Amazon S3 buckets for all uploads and only accessible through the solution. Once a file is uploaded as source data, it is treated the same as Amazon S3 sourced data.	X	X

Connector	Description	Preview*	Source Query**
Google Analytics	Source data from Google Analytics supports import of analytics dimensions and metrics. It supports both full import and incremental import. The authentication requires a service account to be provisioned in order to connect to the API for continuous import. For details on creating service account for Google Analytics, refer to [Create a client ID for Google Analytics API]() topic.	X	

Connector	Description	Preview*	Source Query**
Google BigQuery	Source data from Google BigQuery supports any queries that is runnable within BigQuery. Service account that has read permission to the BigQuery API is required for authentication through Automated Data Analytics on AWS. To provision a service account in Google BigQuery, refer to Managing service accounts page for more details.	X	

Connector	Description	Preview*	Source Query**
Google Storage	Data from Google Storage supports folder level import. The connector uses RSync API to synchronize between source bucket in Google Storage and destination in shared S3 Bucket managed by Automated Data Analytics on AWS. Data removed from the source bucket will also be removed from the destination.		
Microsoft SQL Server	Source data from a Microsoft SQL Server database. The connector uses JDBC to connect to the target database and import the data from the specific table into a S3 bucket managed by Automated Data Analytics on AWS.	X	

Connector	Description	Preview*	Source Query**
MongoDB	Source data from MongoDB or Amazon DocumentDB. The connector supports server TLS and client certifica te. It also offers a bookmark field to support incremental importing.	X	X
MySQL5	Source data from MySQL Server 5. The connector uses JDBC to connect to the target database and import the data from the specific table into a S3 bucket managed by Automated Data Analytics on AWS.	X	
Oracle	Source data from Oracle databases. The connector uses JDBC to connect to the target database and import the data from the specific table into a S3 bucket managed by Automated Data Analytics on AWS		

Connector	Description	Preview*	Source Query**
PostgreSQL	Source data from a PostgreSQL database. The connector uses JDBC to connect to the target database and import the data from the specific table into a S3 bucket managed by Automated Data Analytics on AWS.	X	

* Indicates the source type supports **preview** feature for fast schema inference and managing schema during full import process.

** Indicates the source type supports **source query** feature to grant creator direct query access to the original source data through queries.

Querying data

Automated Data Analytics on AWS provides a unified mechanism for governing federated queries, simplifies complex SQL queries through identifiers, provides caching to lower latency, while enabling saving and sharing queries across organizations.

SQL identifiers

Automated Data Analytics on AWS provides a common hierarchy for identifying and referencing entities within SQL query syntax based on namespaces. Each data product is created within a domain and has a unique name within that domain, which combined, formulate the unique SQL identifier for the data product. For example, a `user_sessions` data product created in a `marketing` domain is referenced by the `marketing.user_sessions identifier`.

Data products may combine multiple datasets, such as for JSON relationalized data. In this case, the nested datasets (tables) are identified by appending the table name to the data product identifier. Following the example above, if `user_sessions` has two nested tables of *clicks* and *orders*, they would be identified as `marketing.user_sessions.clicks` and `marketing.user_sessions.orders` respectively.

When there is a single dataset, there is no need to append the table name to the end of the SQL identifier, as this is handled automatically.

For source types that support the *source query* feature, the creator of the data product is able to directly query the source using the namespace of source prefixed to the SQL identifier of the data product. The naming convention after the source namespace stays the same, however, the underlying schema and tables are likely different, given there are no transformations applied. For example, to query the source from the above example, use `source.marketing.user_sessions.clicks`. The query results from *source query* are un-governed queries and return the raw source data, which may contain PII or other sensitive information. For this reason, only the creator is able to perform these queries.

There is data within the solution that is contextually relevant and accessible only for the calling users, such as saved queries as described below. This solution reserves the namespace **my** for caller based SQL query identifiers, such as `my.queries.example`.

Query governance

Automated Data Analytics on AWS provides a single pane of governance across all data at the business level rather than at the entity level to minimize governance overhead and provide consistency in how data is handled. Governance is managed by defining governance attributes and mapping user groups to governance policies against each attribute. For more information on how this is structured, refer to the *Governance service*. Rather than applying governance policies at the data product level, governance is based on mapping the data product schema columns to business attributes. When an end user runs a query, this solution will rewrite the SQL query to apply governance. It does this by:

- destructing the query into explicit column names
- looking up governance attributess based on column names
- mapping governance attributes to governance policy based on the callers group memberships
- wrapping respective columns to governance UDF for given policy
- execute the governed query in Amazon Athena
- return the governed results

For example, an administrator can create ‘name’, ‘email’, and ‘postcode’ governance attributes in namespace called personal, and apply group governance policy mapping as shown in the table below. A user then creates a new data product (`example.contacts`) that contains PII data (`columns contact_name, contact_email, contact_zip`) and maps the schema columns to the respective governance attributes as shown in the table below. When a *User* in the *Default* group queries this data product with SELECT * FROM `example.contacts`

1) query is destructed into explicit column names of SELECT `contact_name, contact_email, contact_zip` FROM `example.contacts`

2) columns are mapped to respective governance attributes such that `contact_name` -> personal:name

3) the *Default* user group policy is applied to columns such that `contact_name -> HASHED`

4) SQL query is rewritten as SELECT `HASHED(contact_name), HIDDEN(contact_email), CLEAR(contact_zip)` FROM `example.contacts`

5) query is run

6) returns the governed results to user.

Attribute	Group	Policy		Column	Attribute
personal: name	Default	HASHED		contact_name	personal: name
	Power	CLEAR		contact_email	personal: email
personal: email	Default	HIDDEN		contact_zip	personal: postcode
	Power	HASHED			
personal: postcode	Default	CLEAR			
	Power	CLEAR			

Input Query	Group	Executed Query
SELECT * FROM example.contacts	Default	SELECT HASHED(contact_name), HIDDEN(contact_email), CLEAR(contact_zip) FROM example.contacts
	Power	SELECT CLEAR (contact_name), HASHED(contact_email), CLEAR(contact_zip) FROM example.contacts

A user can belong to multiple groups with multiple governance policy mappings defined; when multiple policies are available for a user the user will be granted the *most permissive* policy. Using the example above, if the user belongs to both the *Default* and *Power* user groups, they will receive the policies for *Power* user, since they are more permissive. The governance policies are ordered from least-to-most permissiveness as follows: HIDDEN, HASHED, and CLEAR. For more details on governance policy UDF specifics, see the Governance table.

Currently, this solution only supports applying governance policies to the system user groups: *Default, Power*, and *Administrator*. Support for custom groups will be added in future versions.

Query caching

To reduce query execution times and reduce execution costs, Automated Data Analytics on AWS provides a caching mechanism for queries that ensures the governance is maintained based on the calling user's access.

Queries are cached based on the rewritten SQL query rather than the initial input SQL query, such that if users share the exact same governance policy mapping, they will benefit from query caching. After the input query is rewritten with governance applied, the resulting query is looked up in the cache, and if the underlying data products have not been updated since the cache time, the results are returned directly from the cached results without running the query.

Saved queries

Automated Data Analytics on AWS enables users to save queries, share saved queries with other users, and reference saved queries within other SQL queries to easily build complex queries.

Saved queries are stored within a namespace and either private to the current user only or shared with other users through a domain. All saved queries can be referenced within SQL following `{namespace}.queries.{queryId}` syntax as the identifier.

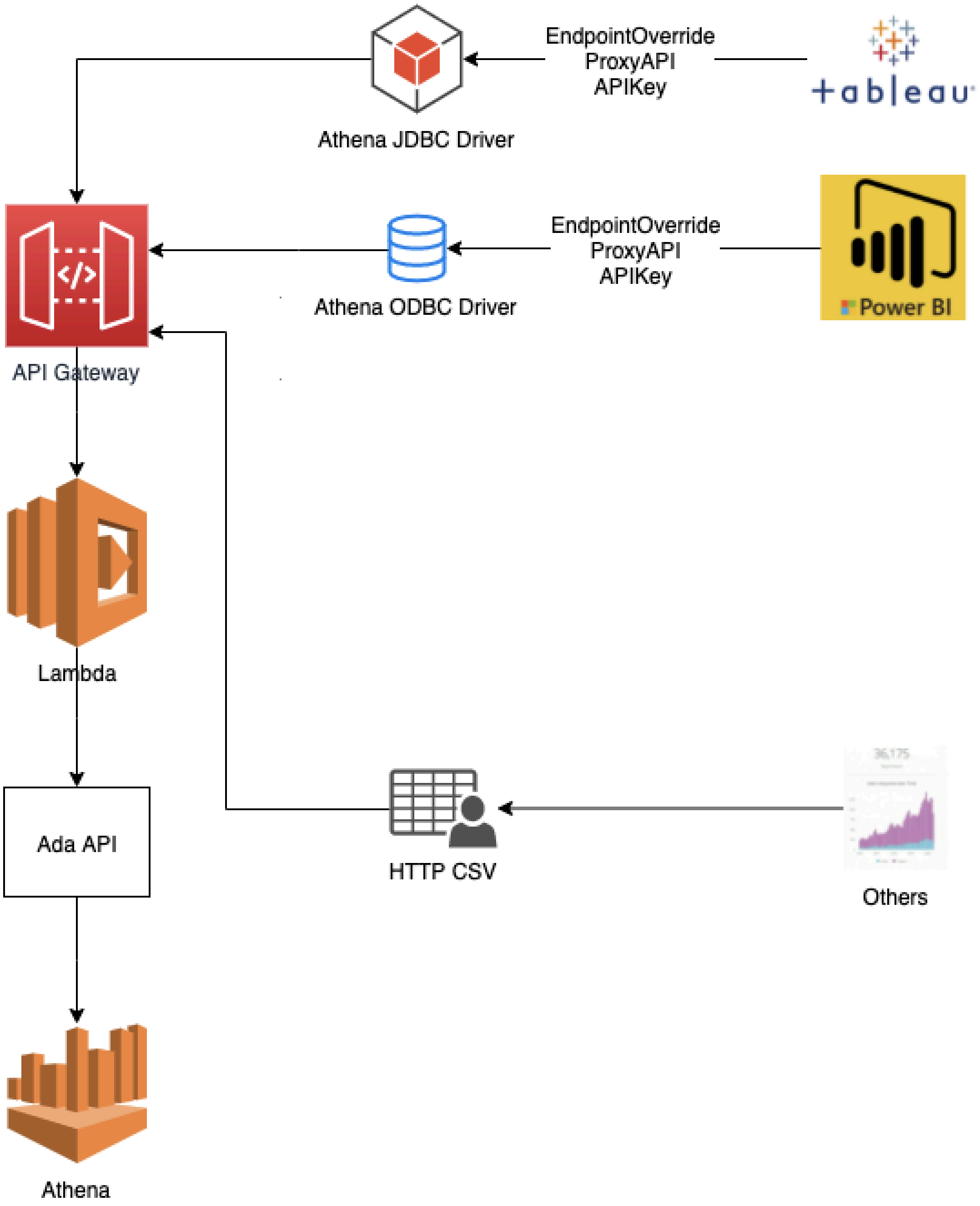
When a SQL query is run, that references a saved query, the referenced saved query is deconstructed as a windowed table and replaced prior to applying governance and executing the rewritten query. For example, if a user saves `SELECT * FROM example.contacts as all_contacts` and then runs `SELECT contact_name, contact_email FROM my.queries.all_contacts`, the query will first be expanded to `SELECT contact_name, contact_email FROM (SELECT contact_name, contact_email, contact_zip FROM example.contacts)` before governance is applied and the query runs.

It is possible to build very complex SQL queries based on multiple saved queries. You must consider performance, as each saved query is run as a subquery and common queries could be accessing the same underlying table without benefitting from a single query.

Private queries are stored in unique namespace for each user and only accessible by that user. Private queries are referenced through the reserved namespace of `my.queries` that maps to the calling users unique namespace. For example, `my.queries.example1` references the calling users example1 saved query.

Shared queries are stored within a domain and accessible to all users through the Domain. Each domain has a reserved keyword of queries that is used to identity saved queries within the domain within SQL. For example, `Domain1.queries.example1` would reference the save query example within the `Domain1 Domain`.

Data egress



Data egress components

Automated Data Analytics on AWS provides egress endpoints to commonly used third-party Business Intelligence platforms through Athena JDBC or ODBC connections.

This enables Automated Data Analytics on AWS to perform DML (filtering, joining) operations using the third-party tool by visual mapping and custom SQL queries. Through visual mapping, the underlying queries associated with the visual mapping, such as joins and aggregation, gets passed onto the data source as queries instead of run locally within the visualization engine. Users can write custom SQL queries that get passed to Automated Data Analytics on AWS API endpoint without additional translations. Data governance is enforced in this process, built on top of a simple HTTP proxy with client credentials.

Client applications, such as Tableau and PowerBI, connect to Automated Data Analytics on AWS via the standard Athena JDBC/ODBC drivers (AthenaJDBC42_2.0.25.1001 and ODBC 1.1.15). The Athena JDBC/ODBC driver supports custom endpoints override Athena JDBC/ODBC Drive Endpoint Override. Automated Data Analytics on AWS leverages this endpoint override to proxy requests from third-party applications to Automated Data Analytics on AWS. For more information on how to configure endpoints, refer to the Connecting to third party tools section. The proxy endpoint is built on API Gateway (HTTP API) and backed by Lambda to validate routes requests to Automated Data Analytics on AWS.

For authentication and authorization, an API key generated from Automated Data Analytics on AWS solution web UI is used to exchange access tokens through Cognito oauth2 request. Each API key maps to a Cognito App Client and the client ID is stored in Automated Data Analytics on AWS token table for mapping to individual user identities.

For example, when requesting for a list of Automated Data Analytics on AWS domains, the Athena driver sends the command for listing Data Catalogs `AmazonAthena.ListDataCatalogs`. The requests received by the proxy endpoints are translated to an HTTP API `callapi.listDataProductDomains` in the solution. Domains are returned and mapped back to Data Catalogs before returning the response back to the Athena driver.

Below is a list of mappings between Athena Driver commands and HTTP API endpoints in Automated Data Analytics on AWS:

Athena	Athena Driver Commands	Automated Data Analytics on AWS interface	HTTP endpoints
Catalog	ListDataCatalogs	Domain	listDataProductDomains
Database	ListDatabases	Data Product	listDataProductDomainDataProducts

Athena	Athena Driver Commands	Automated Data Analytics on AWS interface	HTTP endpoints
Table	ListTableMetadata	Dataset	getDataProductDomainDataProduct
	GetTableMetadata		getDataProductDomainDataProduct
Query	StartQueryExecution	Query	postQuery
	GetQueryExecution		getQueryStatus
	GetQueryResults		listQueryResultsAsAthenaResults

API Gateway

HTTP API is used as `EndpointOverride` connection property in Athena as the JDBC/ODBC driver does not support subpath.

For example, `EndpointOverride=example.execute-api.com:443` will work but `EndpointOverride=example.execute-api.com:443/prod` will fail.

As the HTTP API default stage does not form part of the URL path, integration with JDBC/ODBC driver is supported.

Proxy Lambda

A proxy Lambda is used to intercept the Athena requests coming from the client. A typical request header consists of an Authorization token and x-amz-target argument. The proxy lambda function extracts the API-key from Authorization token and the Athena API from x-amz-target. Mappings between Athena and Automated Data Analytics on AWS hierarchy is as follows.

Athena	Automated Data Analytics on AWS
Catalog	Domain

Athena	Automated Data Analytics on AWS
Database	Data Product
Table	DataSet

Note

Automated Data Analytics on AWS always adds *AwsDataCatalog* as a default catalog, as the JDBC driver hard codes the evaluation in the driver. If that catalog does not persist, it throws an exception before requesting further.

To parse, Automated Data Analytics on AWS uses the following Athena command proxy intercepts and handles.

AmazonAthena.CreatePreparedStatement
AmazonAthena.GetQueryExecution
AmazonAthena.GetQueryResults
AmazonAthena.GetTableMetadata
AmazonAthena.GetWorkGroup
AmazonAthena.ListDataCatalogs
AmazonAthena.ListDatabases
AmazonAthena.ListTableMetadata
AmazonAthena.StartQueryExecution
AmazonAthena.StopQueryExecution

Athena Proxy API Forwarding

The Athena Proxy translates Athena requests and then forward it to the REST endpoints of the solution. This includes requests to retrieve Domain, Data Product and query execution. One exception to this is the `GetQueryResults` request. Instead of translating the Athena results to Automated Data Analytics on AWS then sending back to Athena again, a special API is created: `[API Domain]/[execution-id]/result-as-athena`.

This bypasses the type conversion and sends the Athena results (after ontology mapping and governance rules are applied) back to the client. This minimizes data type conversion errors and improves the end-to-end performance.

Athena Proxy API Authentication

The implementation of client authentication is through a standard API key.

Each API key maps to an App Client in Amazon Cognito. API keys are sent through HTTP header from original request and then exchanged for an access token via oauth2 client credentials flow by passing client id and secret combination against the Amazon Cognito client App.

> **Note**
>
> Currently, the maximum number of Amazon Cognito App Client supported by each Amazon Cognito User Pool is 1,000 by default, but can be increased to 10,000 on request. This decides the maximum number of API keys that can exist concurrently per solution deployment which is capped by this number.

AWS services in this solution

Core services

AWS service	Description
AWS Lambda	Handling API requests (NodeJS and Java).
Amazon Simple Storage Service	Storing processed data, user-defined scripts, and file uploads.
AWS Step Functions	- Managing the lifecycle of data products.

AWS service	Description
	- Orchestrate the asynchronous life-cycle of query execution.
AWS Glue	Tables and resources for handling the data extract, transform, and load (ETL) processing.
Amazon DynamoDB	- Saved queries, query history, and query caching. - Storing governance metadata.
Amazon Athena	Performing federated queries which stores results in Amazon S3 buckets.

Supporting services

AWS service	Description
Amazon CloudFront	The solution uses Amazon CloudFront for distribution, and is protected by AWS WAF
Amazon Simple Storage Service	Host and serve the web front end, including the HTML pages, CSS stylesheets, and JavaScript code.
Amazon DynamoDB	Manage and provide persistent notifications in the user interface, along with Amazon API Gateway REST API resources (resource , method, model), AWS Lambda handler (NodeJS), and an Amazon EventBridge rule for mapping events to notifications.
Amazon Cognito user pool	Manages federating and storing users from external identity providers (IDPs).
Amazon API Gateway	- Federating requests and access to all underlying services and resources.

AWS service	Description
	• Proxying egress requests from external clients (for example, Tableau, PowerBI) via an Amazon Cognito client credentials and facilitating the request and response to support client formats
Amazon EventBridge	Event-driven application messaging between microservices, and to propagate and persist notifications to users.
AWS Lambda	Handling source import.
AWS CloudFormation	Manage solution resources.
AWS Step Functions	Orchestrating lifecycle management.
AWS Glue	Used for crawlers, data catalogues, and jobs for ETL.
AWS Secrets Manager	Secrets to store external credentials.
Amazon ECS	Processing large data ingestion jobs.
Amazon Athena and Amazon Comprehend	Detecting PII entities.

Plan your deployment

This section describes the cost, security, Region, and quota considerations for planning your deployment.

Cost

You are responsible for the cost of the AWS services used while running this solution. As of April 2023, the cost for running this solution with the default settings in the **Asia Pacific (AP-Southeast-2)** is approximately **USD $1246.46 per month.**

See the pricing webpage for each AWS service used in this solution.

We recommend creating a budget using the AWS Cost Explorer to help manage costs. Prices are subject to change. For full details, see the pricing webpage for each AWS service used in this solution.

> **Note**
>
> Starting from ADA release v1.1, we added support for customers to allow private network connectivity (non-Internet) between ADA to others VPCs or on-premises network so that users can import data sources such as relational databases or MongoDB from their internal network. As a result of the greater connector choices, usage of the solution incurs a small monthly additional cost. See the Cost table below for more information.

> **Note**
>
> The cost for running the Automated Data Analytics on AWS solution is based on variety of factors, such as the number and sizes of datasets, number and complexity of transformations, and frequency of updates. For example, a large dataset that is running multiple chained transformations with automatic updates when new data is detected will result in higher costs than smaller datasets without any transformations and updates that are triggered on demand.

Cost table

The following table provides a sample cost breakdown for deploying this solution with the default parameters in the Asia Pacific (AP-Southeast-2) Region for one month.

Scenarios	Ingest size: 1GB/day, Query size: 5GB/day	Ingest size: 1GB/hr, Query size: 800mb/hr	Ingest size: 5GB/hr, Query size: 2.7GB/hr
AWS service	Monthly cost (USD)	Monthly cost (USD)	Monthly cost (USD)
Amazon Comprehend	$9.00	$249.30	$1614.96
AWS CloudTrail	$162.95	$247.68	$467.51
Amazon EC2 - Other (NAT Gateway)	$87.11	$142.11	$448.77
AWS Lambda	$158.38	$158.39	$158.55
AWS Key Management Service (AWS KMS)	$112.48	$112.93	$113.63
AWS Glue	$0.60	$24.19	$112.74
AWS WAF	$34.87	$34.89	$34.88
Amazon Kinesis	$31.25	$31.25	$31.25
Amazon GuardDuty	$0.89	$6.05	$30.03
AWS Secrets Manager	$4.79	$7.36	$9.75
Amazon CloudWatch	$5.46	$6.71	$22.30
Amazon Elastic Container Service (Amazon ECS)	$0.14	$3.63	$22.77
AWS Step Functions	$0.14	$3.75	$18.28

Scenarios	Ingest size: 1GB/day, Query size: 5GB/day	Ingest size: 1GB/hr, Query size: 800mb/hr	Ingest size: 5GB/hr, Query size: 2.7GB/hr
Amazon Simple Storage Service (Amazon S3)	$2.76	$1.07	$2.26
Amazon Athena	$0.00	$0.63	$1.32
Amazon Elastic Container Registry (Amazon ECR)	$0.43	$0.43	$0.43
Amazon DynamoDB	$0.05	$0.09	$0.21
Amazon Data Firehose	$0.01	$0.02	$0.01
Amazon CloudWatch Events	$0.00	$0.00	$0.01
AWS X-Ray	$0.00	$0.00	$0.00
Amazon CloudFront	$0.00	$0.00	$0.00
Amazon EC2-Instances	$0.00	$0.00	$0.00
Amazon API Gateway	$0.00	$0.00	$0.00
Amazon Cognito	$0.00	$0.00	$0.00
Amazon Simple Notification Service (Amazon SNS)	$0.00	$0.00	$0.00
Amazon Simple Queue Service (Amazon SQS)	$0.00	$0.00	$0.00

Scenarios	Ingest size: 1GB/day, Query size: 5GB/day	Ingest size: 1GB/hr, Query size: 800mb/hr	Ingest size: 5GB/hr, Query size: 2.7GB/hr
AWS Transit Gateway	$101.40 (with 1VPC/on-premises connection and 1GB/day data ingress	$216 (with 3 VPC/on-premises connectio ns and 1GB/hr data ingress)	$626.40 (with 10 VPC/on-premises connections and 5GB/hr data ingress)
Total cost	**~$712.60**	**~$1246.46**	**~$3715.86**

Note

This cost estimate does not account for the cost of any 3rd party services, such as data egress.

Security

When you build systems on AWS infrastructure, security responsibilities are shared between you and AWS. This shared responsibility model reduces your operational burden because AWS operates, manages, and controls the components including the host operating system, the virtualization layer, and the physical security of the facilities in which the services operate. For more information about AWS security, visit AWS Cloud Security.

IAM roles

AWS Identity and Access Management (IAM) roles allow customers to assign granular access policies and permissions to services and users on the AWS Cloud. Automated Data Analytics on AWS creates IAM roles that grant the solution's constructs to access Regional resources provisioned by the solution, such as:

- IAM roles used by the Lambda functions that implements the APIs to read and write data in S3 buckets and DynamoDB tables or
- IAM roles used by AWS Glue crawlers and jobs to read and write data in S3 buckets.

Though following the principle of least privilege on all IAM roles provisioned, due to the complexity of this solution, the Automated Data Analytics on AWS solution requires more exclusive control and access over the resources within the account it is deployed. It is recommended to deploy and operate this solution in its own dedicated AWS account instead of sharing the same AWS account with other cloud workloads.

IAM resource policies

AWS Identity and Access Management (IAM) resource-based policies are JSON policy documents that you attach to a resource such as an Amazon S3 bucket. These policies grant the specified principal permission to perform specific actions on that resource and defines under what conditions this applies. Automated Data Analytics on AWS supports granting the solution access to resources that are not provisioned by the solution (external resources), such as granting the solution read access to an Amazon S3 bucket to import source data. Automated Data Analytics on AWS uses session tagging to allow for external resource policies to manage granular access of the solution, groups, and users through PrincipalTag conditions.

Principal	Policy Condition

Solution	`"Condition": {`
	`    "StringLike": {`
	`    "aws:PrincipalTag/ada:service":`
	`    "*"`
	`    }`
	`    }`

Service (*query* or *data-product*)	`"Condition": {`
	`    "StringLike": {`
	`    "aws:PrincipalTag/ada:service":`
	`    "data-product"`
	`    }`
	`    }`

Group	`"Condition": {`
	`    "ForAnyValue:StringLike": {`
	`    "aws:PrincipalTag/ada:groups": [`
	`    "*:admin:*",`
	`    "*:power-user:*"`

Principal	Policy Condition

	`    ]`
	`    }`
	`    }`

User	`"Condition": {`
	`    "ForAnyValue:StringLike": {`
	`    "aws:PrincipalTag/ada:user": [`
	`    "user-id-1",`
	`    "user-id-2"`
	`    ]`
	`    }`
	`    }`

This is an example policy that grants *power-user* users read access to an Amazon S3 bucket for creating data products and delegated read access to the Amazon S3 bucket to Automated Data Analytics on AWS when querying the data product.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Grant Automated Data
  Analytics on AWS User access",
  "Effect": "Allow",
            "Action": "s3:Get*",
            "Resource": "arn:aws:s3:::<bucket-name>/*",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::<ada-account>:root"
                ]
            },
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalTag/ada:user": "<ada-user>"
                }
            }
        },
        {
            "Sid": "Grant Automated Data
```

```
Analytics on AWS Group access",
"Effect": "Allow",
          "Action": "s3:Get*",
          "Resource": "arn:aws:s3:::<bucket-name>/*",
          "Principal": {
              "AWS": [
                  "arn:aws:iam::<ada-account>:root"
              ]
          },
          "Condition": {
              "StringLike": {
                  "aws:PrincipalTag/ada:groups": "*:power-user:*"
              }
          }
      },
      {
          "Sid": "Grant Automated Data
Analytics on AWS Group access in one of group",
"Effect": "Allow",
          "Action": "s3:Get*",
          "Resource": "arn:aws:s3:::<bucket-name>/*",
          "Principal": {
              "AWS": [
                  "arn:aws:iam::<ada-account>:root"
              ]
          },
          "Condition": {
              "ForAnyValue:StringLike": {
                  "aws:PrincipalTag/ada:groups": [
                      "*:power-user:*",
                      "*:some-custom-group:*"
                  ]
              }
          }
      },
      {
          "Sid": "Grant Automated Data
Analytics on AWS Federated Query access",
"Effect": "Allow",
          "Action": "s3:Get*",
          "Resource": "arn:aws:s3:::<bucket-name>/*",
          "Principal": {
              "AWS": [
                  "arn:aws:iam::<ada-account>:root"
```

```
                ]
            },
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalTag/ada:service": "query"
                }
            }
        },
        {
            "Sid": "Grant access from any
  Automated Data Analytics on AWS microservice",
  "Effect": "Allow",
            "Action": "s3:Get*",
            "Resource": "arn:aws:s3:::<bucket-name>/*",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::<ada-account>:root"
                ]
            },
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalTag/ada:service": "*"
                }
            }
        }
    ]
}
```

Amazon Cognito

Automated Data Analytics on AWS uses Amazon Cognito user and identity pools. User pools are user directories that provide sign-in functionality for the web users. Identity pools provide AWS credentials to grant the web users access to other AWS services, such as the ability to access data stored in Amazon S3. After a successful user pool sign-in, Automated Data Analytics on AWS's web solution UI receives user pool tokens from Amazon Cognito. These tokens are used to control access to server-side resources. For example, the API Gateway instance is configured with a Cognito authorizer that validates web requests for the presence of a proper token (for example, signed by the user pool and hasn't expired).

This solution provisions a *root_admin* user when first deployed that is managed by the user pool and uses the Amazon Cognito Hosted UI to sign-in. All other users are managed by external identity providers configured by the *root_admin* through federated sign-in. The Cognito user pool handles the federated sign-in with the identity provider to authenticate users and return tokens based on user identity.

This solution also supports machine-to-machine access through Cognito app clients managed by the user pool to enable *client credential* flow.

Amazon CloudFront

This solution deploys a web user interface (UI) hosted in an Amazon S3 bucket. To help reduce latency and improve security, this solution deploys an Amazon CloudFront distribution with an origin access identity. For more information, refer to Restricting Access to Amazon S3 Content by Using an Origin Access Identity in the *Amazon CloudFront Developer Guide*.

This solution also deploys two APIs in the AWS APIGateway. One API is the REST API that provides access to ADA backend and the other API is an HTTP API that serves as the Athena Proxy for egress connection with third party analytics tools.

AWS WAF

This solution deploys AWS WAF, a web application firewall that helps protect the solution against common web exploits that might affect availability, compromise security, or consume excessive resources. AWS WAF provides control over how traffic reaches the solution, such as using security rules that block requests that don’t originate in an allow-list of CIDR IP range.

The solution supports specifying the allow-list of CIDR IP ranges for AWS Cloud Development Kit (AWS CDK) (AWS CDK) deployments to further secure all solution endpoints by restricting access. See CDK Deployment for more details on enabling AWS WAF IP allow-list.

Amazon API Gateway

You can access the REST APIs and HTTP APIs deployed by this solution from the web UI or via third party analytics tools to consume services and query data. Both API endpoints are protected by either Amazon Cognito authentication or the API key issued by the solution.

- Use AWS WAF to activate an allow-list of IP range that API calls can originate from.
- Use the API Gateway resource policy to create an allow-list of IP range that API calls can originate from.
- Use mutual TLS authentication for API Gateway to allow calls from trusted parties only.

- Configure the API to be private using Amazon VPC endpoint to limit access to callers within a particular VPC or on-premises connecting via Direct Connect or VPN.
- Use IAM to restrict access to the API.

Design considerations

We have designed the Automated Data Analytics on AWS solution in accordance with the following principles and considerations:

Serverless and well-architected

- The AWS solution has been designed and built in alignment with the AWS Well-Architected framework with focus on the serverless lens using event-driven design principles. This was done to ensure the solution can scale without constraints while being performant and cost effective.

Automated deployment

- The AWS solution can be deployed via a CloudFormation template with fully inspectable resources through CDK. This greatly simplifies the deployment process for users with an estimated completion time of approximately 1 hour. The solution is open-source and made available through GitHub. This allows users to customize the solution to fit their use case and contribute back to the community.

Data lifecycle management

- The AWS solution is designed to manage the entire data lifecycle, from ingestion to storage for both raw and processed data, with user defined policies for refreshing source datasets. These include on-demand, scheduled, and automatic.

On-Demand pricing

- Automated Data Analytics on AWS offers pay-as-you-go pricing. With Automated Data Analytics on AWS, you pay only pay for the underlying services it uses for as long as you use them, without any long-term contracts or complex licensing. You only pay for the services you consume, and once you stop using them, there are no additional costs or termination fees.

SQL as a first class citizen

- This solution allows users to query data with standard SQL statements as the primary query language. This enables a broad range of non-technical users to derive insights from their data without deep expertise in data engineering.

Identity management

- This solution support integration with external identity providers out of the box to facilitate federated access from a customer's identity system. It provides a way for customer to seamlessly integrate it into their existing infrastructure.

Analyst persona centric UI/UX

- Automated Data Analytics on AWS's user interface iss designed and built with the analyst persona in mind and fine-tuned through direct customer feedback about how they use their data. A key focus in this was simplicity and abstracting away any technical overhead and maintenance.

Data governance

- Automated Data Analytics on AWS has a set of core constructs to automate and simplify data governance across entire an entire organization. Users are assigned to groups with user-defined row and column level controls around visibility to various types of data. Personally identifiable information (PII) is automatically identified and redacted to ensure privacy and compliance is upheld.

Extensible architecture for connectors

- Automated Data Analytics on AWS provides an extensible connector architecture to source data from various data sources for data ingress. it also provides egress integration to allow third party business intelligence tools such as Tableau, and PowerBI to consume the data from it. It offers a large set of pre-built ingress connectors for both AWS services and third party data sources such as Google Cloud. It is expected that the variety of ingress connectors as well as egress integrations will grow in time based on feedback from customers and contributions from the open-source community.

Supported AWS Regions

This solution uses the **AWS Comprehend** service, which is not currently available in all AWS Regions. You must launch this solution in an AWS Region where AWS Comprehend is available. For the most current availability of AWS services by Region, see the AWS Regional Services List.

Automated Data Analytics on AWS is supported in the following AWS Regions:

Region code	Region name
us-east-1	US East (N. Virginia)
us-east-2	US East (Ohio)
us-west-2	US West (Oregon)
ap-south-1	Asia Pacific (Mumbai)
ap-northeast-1	Asia Pacific (Tokyo)
ap-northeast-2	Asia Pacific (Seoul)
ap-southeast-1	Asia Pacific (Singapore)
ap-southeast-2	Asia Pacific (Sydney)
ca-central-1	Canada (Central)
eu-west-1	Europe (Ireland)
eu-west-2	Europe (London)
eu-central-1	Europe (Frankfurt)

Quotas

Service quotas, also referred to as limits, are the maximum number of service resources or operations for your AWS account.

Sufficient AWS Lambda concurrent executions limit

Use AWS Service Quotas in your AWS console to verify your AWS Lambda concurrent executions. The *Applied quota value* in your account should be greater or equal to the *AWS default quota value* (which is 1000). If the *Applied quota value* is less than 1000, select the **Request quota increase** button to request an increase to this value to at least 1000 before deploying the solution. For more information, refer to the AWS Lambda Developer Guide..

Deploy the solution

This solution uses AWS CloudFormation templates and stacks to automate its deployment. The CloudFormation templates describe the AWS resources included in this solution and their properties. The CloudFormation stacks provisions the resources that are described in the templates.

AWS CloudFormation template

You can download the CloudFormation template for this solution before deploying it.

View template

AdaOneClick.template: Use this template to launch the solution and all associated components, including the solution web UI. This template creates an AWS CodeBuild project that facilitates the full deployment via the Cloud Development Kit (CDK) which includes building and vending the necessary ECS container images utilized by the solution.

> **Note**
>
> AWS CloudFormation resources are created from AWS Cloud Development Kit (AWS CDK) constructs.

This AWS CloudFormation template deploys Automated Data Analytics on AWS in the AWS Cloud. You must meet the following prerequisites before launching the stack:

> **Note**
>
> If you have previously deployed this solution, see Update the solution section for update instructions.

Prerequisites

- **A CDK bootstrapped AWS account:** You must bootstrap your AWS CDK environment in the target region you want to deploy, using the AWS CDK toolkit's cdk bootstrap command. From the command line, authenticate into your AWS account, and run `cdk bootstrap 'aws://<YOUR ACCOUNT NUMBER>/<REGION>'`. For more information, refer to the AWS CDK's How to bootstrap page.

- **Sufficient AWS Lambda concurrent executions limit**: Use AWS Service Quotas in your AWS console to verify your AWS Lambda concurrent executions. The *Applied quota value* in your account should be greater or equal to the *AWS default quota value* (which is 1000). If the *Applied quota value* is less than 1000, select the **Request quota increase** button to request an increase to this value to at least 1000 before deploying the solution. For more information, refer to the AWS Lambda Developer Guide.

Deployment process overview

Before you launch the solution, review the cost, architecture, network security, and other considerations discussed earlier in this guide.

Time to deploy: Approximately 1 hour

The automated deployment deploys Automated Data Analytics on AWS with the default configuration settings, and can be used for evaluation and production purposes. However, to fine tune the advanced settings for better performance or customize the solution to your specific environment, it is recommended to download the source code from the GitHub repository and build and deploy the solution with AWS CDK. For more information, refer to the CDK Deployment section.

> **Note**
>
> This solution includes an option to send anonymized operational metrics to AWS. We use this data to better understand how customers use this solution and related services and products. AWS owns the data gathered though this survey. Data collection is subject to the AWS Privacy Policy.
> To opt out of this feature, download the template, update `sendAnonymousData` to **No** before you deploy the solution in AWS CloudFormation console. For more information, see the Anonymized data collection section of this guide.

Launch the stack

Follow the step-by-step instructions in this section to configure and deploy the solution into your account.

Time to deploy: Approximately 1 hour

> **Note**
>
> You are responsible for the cost of the AWS services used while running this solution. For more details, visit the Cost section in this guide, and refer to the pricing webpage for each AWS service used in this solution.

1. Sign in to the AWS Management Console and select the button to launch the **AdaOneClick.template** AWS CloudFormation template.

   Launch solution

2. The template launches in the US East (N. Virginia) Region by default. To launch the solution in a different AWS Region, use the Region selector in the console navigation bar.

   > **Note**
   >
   > This solution uses the AWS Comprehend service, which is not currently available in all AWS Regions. You must launch this solution in an AWS Region where AWS Comprehend is available. For the most current availability by Region, refer to the Supported AWS Regions section.

3. On the **Create stack** page, verify that the correct template URL is in the **Amazon S3 URL** text box and choose **Next**.
4. On the **Specify stack details** page, assign a name to your solution stack. For information about naming character limitations, refer to IAM and STS Limits in the *AWS Identity and Access Management User Guide*.

> **Note**
>
> We recommend naming the stack with the version number included in the name. For example, *ADA-deploy-v110*. Avoid using the name ADA as this name is used for the main stack deployed with the solution.

5. Under **Parameters**, review the parameters for this solution template and modify them as necessary. This solution uses the following default values.

Parameter	Default	Description
BootstrapVersion	`/cdk-bootstrap/hnb659fds/version`	Version of the CDK Bootstrap resources in this environment, automatically retrieved from SSM Parameter Store.
adminEmail	*`<Requires input>`*	The email address of the administrator. This has to be a valid address, you will receive the temporary password to this address.
adminPhoneNumber	*`<Requires input>`* Default is +1555555023	The phone number of the administrator. Must be a valid phone number that can receive OTP messages if MFA is enabled. Change the default value to your phone number if you want to enable this functionality.
adminMFA	ON	Indicates if Multi-Factor Authentication (MFA) is enabled for root administrator.

Parameter	Default	Description
advancedSecurityMode	ENFORCED	The advanced security mode for Cognito UserPool.
autoAssociateAdmin	true	Indicates if the admin role is automatically mapped to the users from external identity provider. This email must match the admin email address provided as a parameter during the deployment.
sendAnonymousData	Yes	Send anonymous operation al metrics to AWS. We use this data to better understan d how customers use this solution and related services and products.

6. Choose **Next**.
7. On the **Configure stack options** page, select **Next**.
8. On the **Review** page, review and confirm the settings. Check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.
9. Choose **Create stack** to deploy the stack.

   You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should receive a `CREATE_COMPLETE` status in approximately 1 hour.

   This CloudFormation template will deploy a stack with the name specified and containing resources to bootstrap the deployment of the main stack of the solution which will show up shortly after this stack is completed on deployment. The main stack of the solution is named **ADA** and it contains the Automated Data Analytics on AWS solution resources.

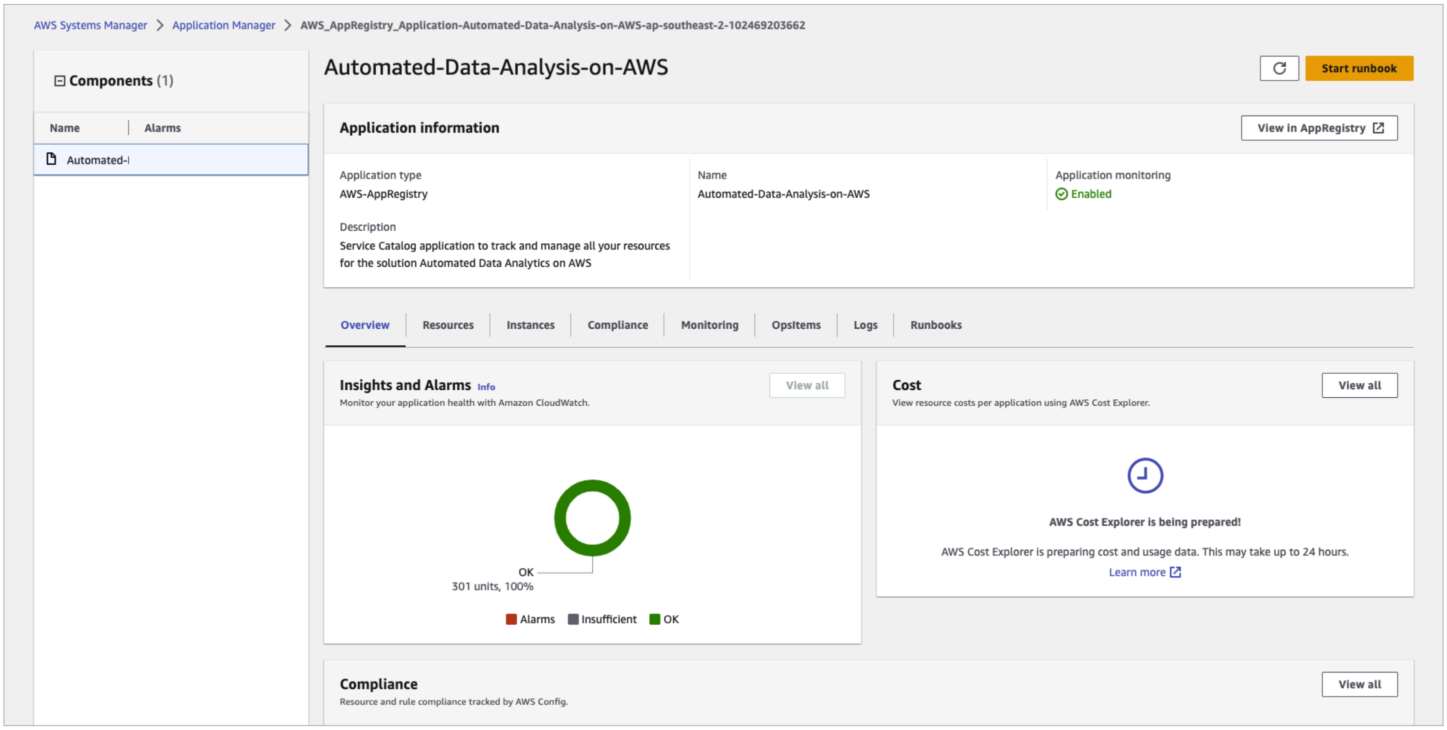
Monitoring the solution with Service Catalog AppRegistry

The Automated Data Analytics on AWS solution includes a Service Catalog AppRegistry resource to register the CloudFormation template and underlying resources as an application in both Service Catalog AppRegistry and AWS Systems Manager Application Manager.

AWS Systems Manager Application Manager gives you an application-level view into this solution and its resources so that you can:

- Monitor its resources, costs for the deployed resources across stacks and AWS accounts, and logs associated with this solution from a central location.
- View operations data for the resources of this solution in the context of an application. For example, deployment status, CloudWatch alarms, resource configurations, and operational issues.

The following figure depicts an example of the application view for the Automated Data Analytics on AWS stack in Application Manager.



AWS Systems Manager > Application Manager > AWS_AppRegistry_Application-Automated-Data-Analysis-on-AWS-ap-southeast-2-102469203662

Components (1)

Name	Alarms
Automated-l	

Automated-Data-Analysis-on-AWS

Application information

Application type
AWS-AppRegistry

Description
Service Catalog application to track and manage all your resources for the solution Automated Data Analytics on AWS

Name
Automated-Data-Analysis-on-AWS

Application monitoring
Enabled

Insights and Alarms Info
Monitor your application health with Amazon CloudWatch.

Cost
View resource costs per application using AWS Cost Explorer.

AWS Cost Explorer is being prepared!

AWS Cost Explorer is preparing cost and usage data. This may take up to 24 hours.

Compliance
Resource and rule compliance tracked by AWS Config.

Automated Data Analytics on AWS stack in Application Manager

Note

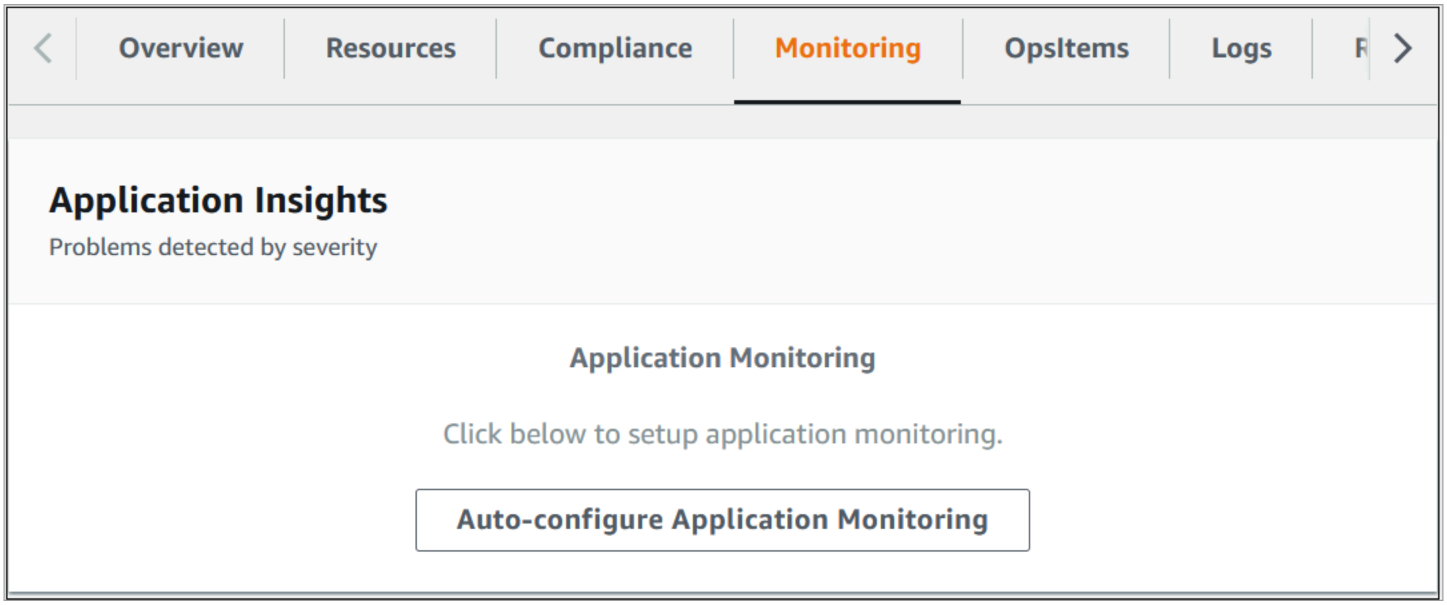
You must activate CloudWatch Application Insights, AWS Cost Explorer, and cost allocation tags associated with this solution. They are not activated by default.

Activate CloudWatch Application Insights

1. Sign in to the Systems Manager console.
2. In the navigation pane, choose **Application Manager**.
3. In Applications, choose AppRegistry applications.
4. In **AppRegistry applications**, search for the application name for this solution and select it.

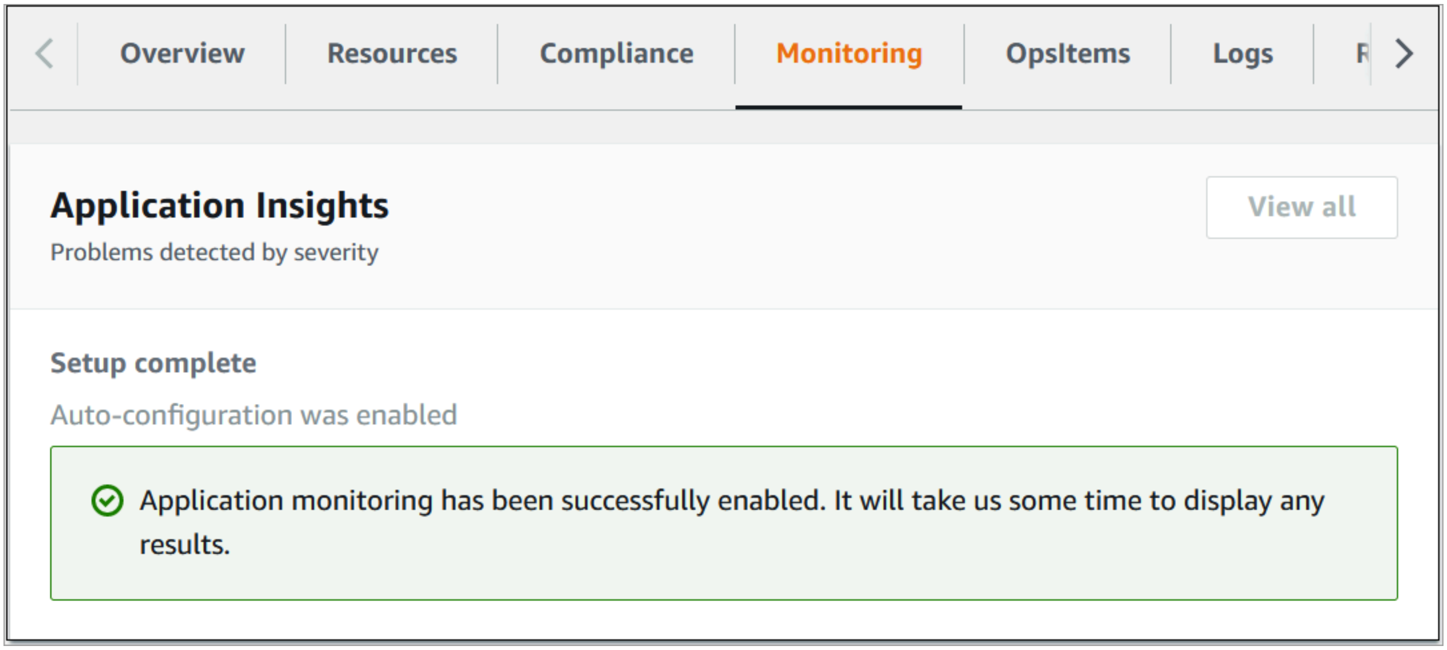
The next time you open Application Manager, you can find the new application for your solution in the **AppRegistry application** category.

1. In the **Components** tree, choose the application stack you want to activate.
2. In the Monitoring tab, in Application Insights, select **Auto-configure Application Monitoring**.


Overview
Resources
Compliance
Monitoring
OpsItems
Logs
R
Application Insights
Problems detected by severity
Application Monitoring
Click below to setup application monitoring.
Auto-configure Application Monitoring

Auto configure application monitoring

Monitoring for your applications is now activated and the following status box appears:



Application Insights

Problems detected by severity

Setup complete

Auto-configuration was enabled

Application monitoring has been successfully enabled. It will take us some time to display any results.

Application monitoring activated

Activate AWS Cost Explorer

You can see the overview of the costs associated with the application and application components within the Application Manager console through integration with AWS Cost Explorer which must be first activated. Cost Explorer helps you manage costs by providing a view of your AWS resource costs and usage over time. To activate Cost Explorer for the solution:

1. Sign in to the AWS Cost Management console.
2. In the navigation pane, select **Cost Explorer**.
3. On the **Welcome to Cost Explorer** page, choose Launch Cost Explorer.

The activation process can take up to 24 hours to complete. Once activated, you can open the Cost Explorer user interface to further analyze cost data for the solution.

Activate cost allocation tags associated with the solution

After you activate Cost Explorer, you must activate the cost allocation tags associated with this solution to see the costs for this solution. The cost allocation tags can only be activated from the management account for the organization. To activate cost allocation tags:

1. Sign in to the AWS Billing and Cost Management and Cost Management console.
2. In the navigation pane, select **Cost Allocation Tags**.
3. On the **Cost allocation tags** page, filter for the AppManagerCFNStackKey tag, then select the tag from the results shown.
4. Choose **Activate**.

The activation process can take up to 24 hours to complete and the tag data to appear.

Update the solution

If you have previously deployed the solution, follow the following steps to update the current deployment with the latest released version.

1. Sign in to the AWS CloudFormation console, and select the correct account and region. The console displays stack named **Ada** and another stack named when the previous deployment was made. For example, *AdaDeployer-v110*.
2. Sign in to the AWS Management Console and select the button to launch the **AdaOneClick.template** AWS CloudFormation template.

   Launch solution.
3. The template launches in the US East (N. Virginia) Region by default. To launch the solution in a different AWS Region, use the Region selector in the console navigation bar.

   > **Note**
   >
   > This solution uses the AWS Comprehend service, which is not currently available in all AWS Regions. You must launch this solution in an AWS Region where AWS Comprehend is available. For the most current availability by Region, refer to the Supported AWS Regions section.
4. On the **Create stack** page, verify that the correct template URL is in the **Amazon S3 URL** text box and choose **Next**.
5. On the **Specify stack details** page, assign a name to your solution stack. For information about naming character limitations, refer to IAM and STS Limits in the *AWS Identity and Access Management User Guide*.

   > **Note**
   >
   > Enter a name for the stack as recommended in the Launch the stack section with the new version number. For example to update from version v1.1.0 to v1.2.0, enter *AdaDeployer-v120* as the name.
6. Under **Parameters**, review the parameters for this solution template and modify them as necessary. This solution uses the following default values.

Parameter	Default	Description
BootstrapVersion	`/cdk-bootstrap/hnb659fds/version`	Version of the CDK Bootstrap resources in this environment, automatically retrieved from SSM Parameter Store.
adminEmail	*`<Requires input>`*	The email address of the administrator. This has to be a valid address, you will receive the temporary password to this address.
adminPhoneNumber	*`<Requires input>`* Default is +1555555023	The phone number of the administrator. Must be a valid phone number that can receive OTP messages if MFA is enabled. Change the default value to your phone number if you want to enable this functionality.
adminMFA	ON	Indicates if Multi-Factor Authentication (MFA) is enabled for root administrator.
advancedSecurityMode	ENFORCED	The advanced security mode for Cognito UserPool.

Parameter	Default	Description
autoAssociateAdmin	true	Indicates if the admin role is automatically mapped to the users from external identity provider. This email must match the admin email address provided as a parameter during the deployment.
sendAnonymousData	Yes	Send anonymous operation al metrics to AWS. We use this data to better understan d how customers use this solution and related services and products.

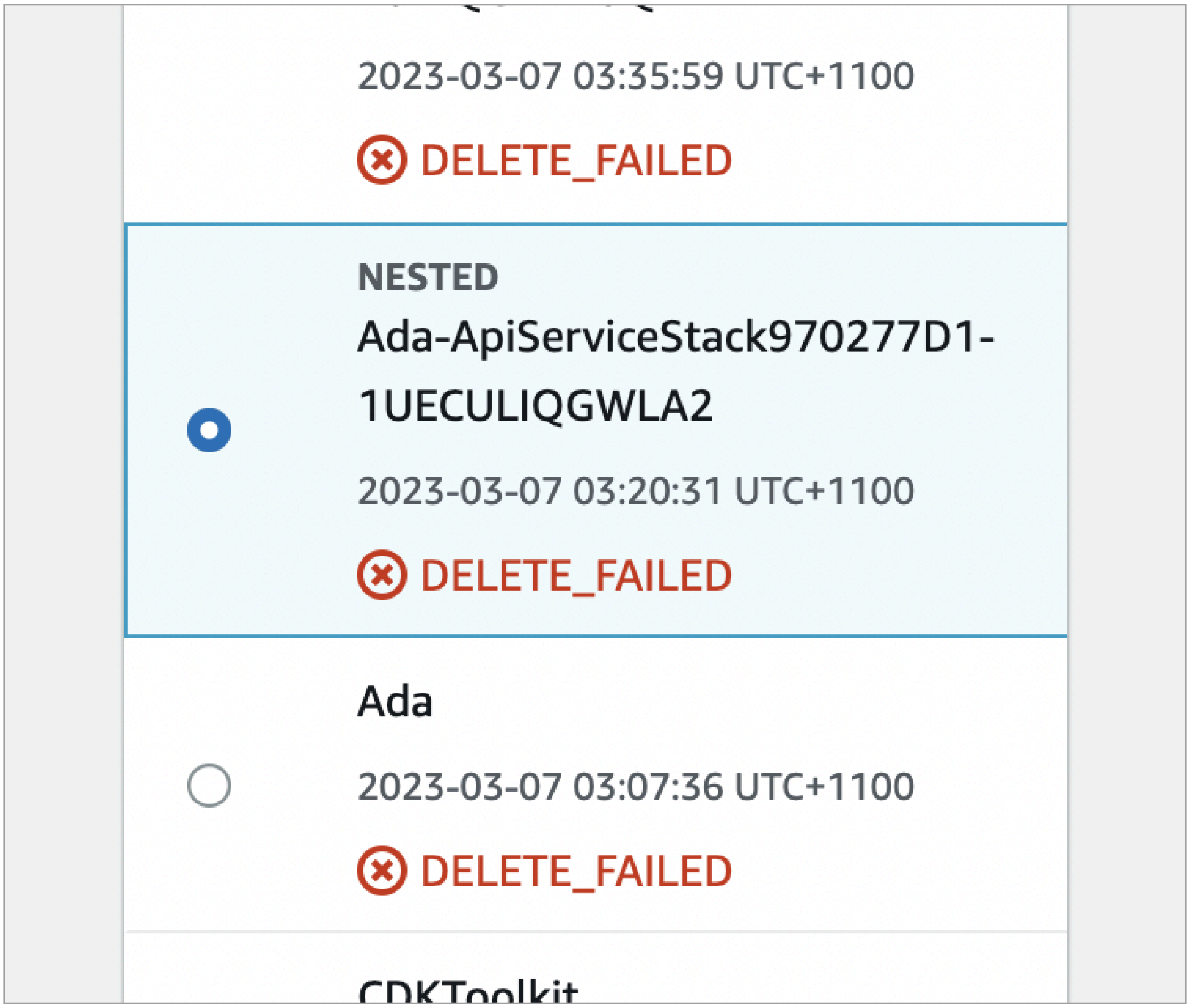
7. Choose **Next**.
8. On the **Configure stack options** page, select **Next**.
9. On the **Review** page, review and confirm the settings. Check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.
10. Choose **Create stack** to deploy the stack.

This CloudFormation template will deploy a stack with the name specified and containing resources to bootstrap the upgrade of the main Ada stack. After the update is complete, the main Ada stack will start upgrading to the new version. You can view the status of the stack in the AWS CloudFormation console in the **Status** column.

Troubleshooting

Nested stacks fail to delete during teardown

During teardown, you may see some of the nested stacks fail to delete, which in turn causes the **Ada** root stack to fail.



2023-03-07 03:35:59 UTC+1100

ⓧ DELETE_FAILED

● NESTED Ada-ApiServiceStack970277D1-1UECULIQGWLA2

2023-03-07 03:20:31 UTC+1100

ⓧ DELETE_FAILED

○ Ada

2023-03-07 03:07:36 UTC+1100

ⓧ DELETE_FAILED

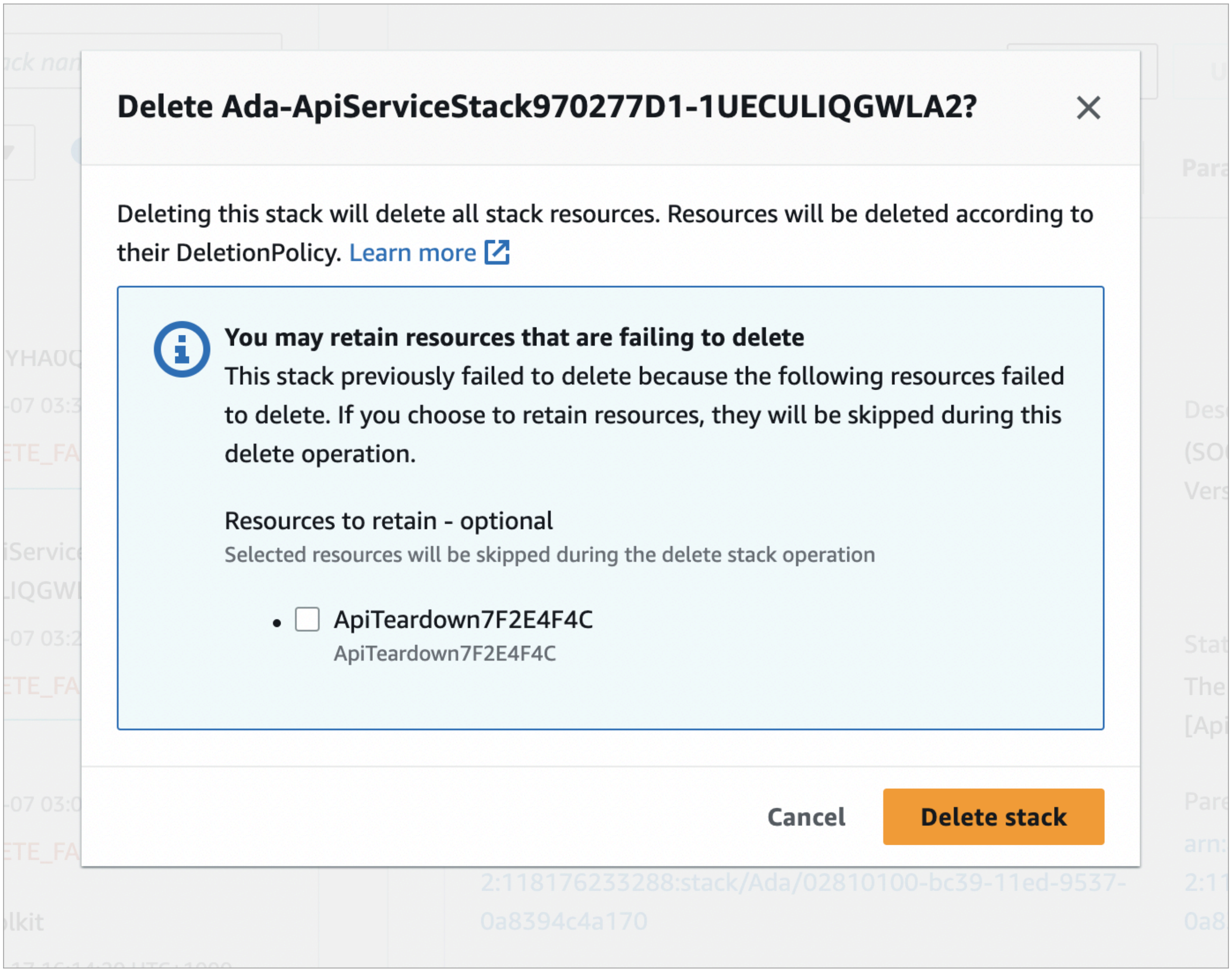
CDKToolkit

Nested stacks fail

Issue: This issue occurs due to the APIGateway REST service level requests limit. For more information, refer to this Github issue.

Resolution: To resolve this:

1. Select the failed stack and delete it from the CloudFormation UI. The following message is displayed.



Delete Ada-ApiServiceStack970277D1-1UECULIQGWLA2?

Deleting this stack will delete all stack resources. Resources will be deleted according to their DeletionPolicy. Learn more

You may retain resources that are failing to delete
This stack previously failed to delete because the following resources failed to delete. If you choose to retain resources, they will be skipped during this delete operation.

Resources to retain - optional
Selected resources will be skipped during the delete stack operation

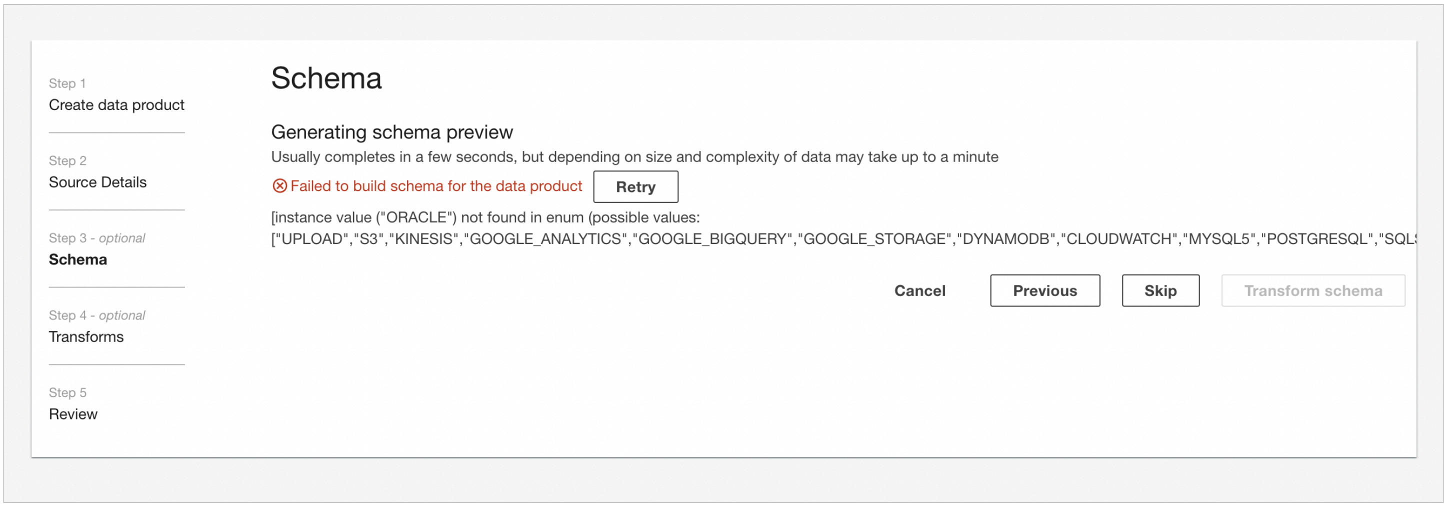
- □ ApiTeardown7F2E4F4C
  ApiTeardown7F2E4F4C

Delete stack from CloudFormation UI

2. Select **Delete stack** to confirm deletion.

Schema preview error

When you update the Automated Data Analytics on AWS solution, where new data products are introduced, you may encounter an error during the Schema preview, when you are adding a new data product.



Schema

Generating schema preview

Usually completes in a few seconds, but depending on size and complexity of data may take up to a minute

⊗ Failed to build schema for the data product

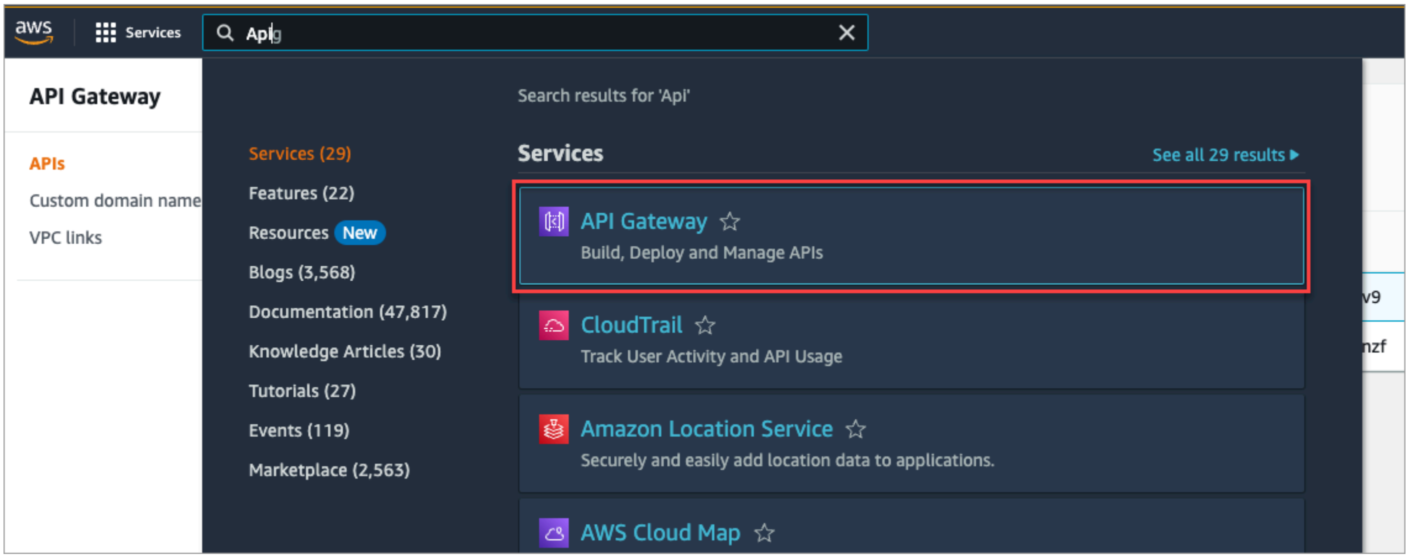
[instance value ("ORACLE") not found in enum (possible values: ["UPLOAD","S3","KINESIS","GOOGLE_ANALYTICS","GOOGLE_BIGQUERY","GOOGLE_STORAGE","DYNAMODB","CLOUDWATCH","MYSQL5","POSTGRESQL","SQL

Schema preview error

Issue: This issue is caused by FederatedRestAPI that services the data products.

Resolution: To resolve this:

1. Navigate to your AWS Management Console > API Gateway. Make sure you select the region where this solution has been deployed.



API Gateway

2. On the APIs page, click to select **FederatedRestApi**. The API details are displayed.
3. From the Actions dropdown, select **Deploy API**. The Deploy API dialog displays.

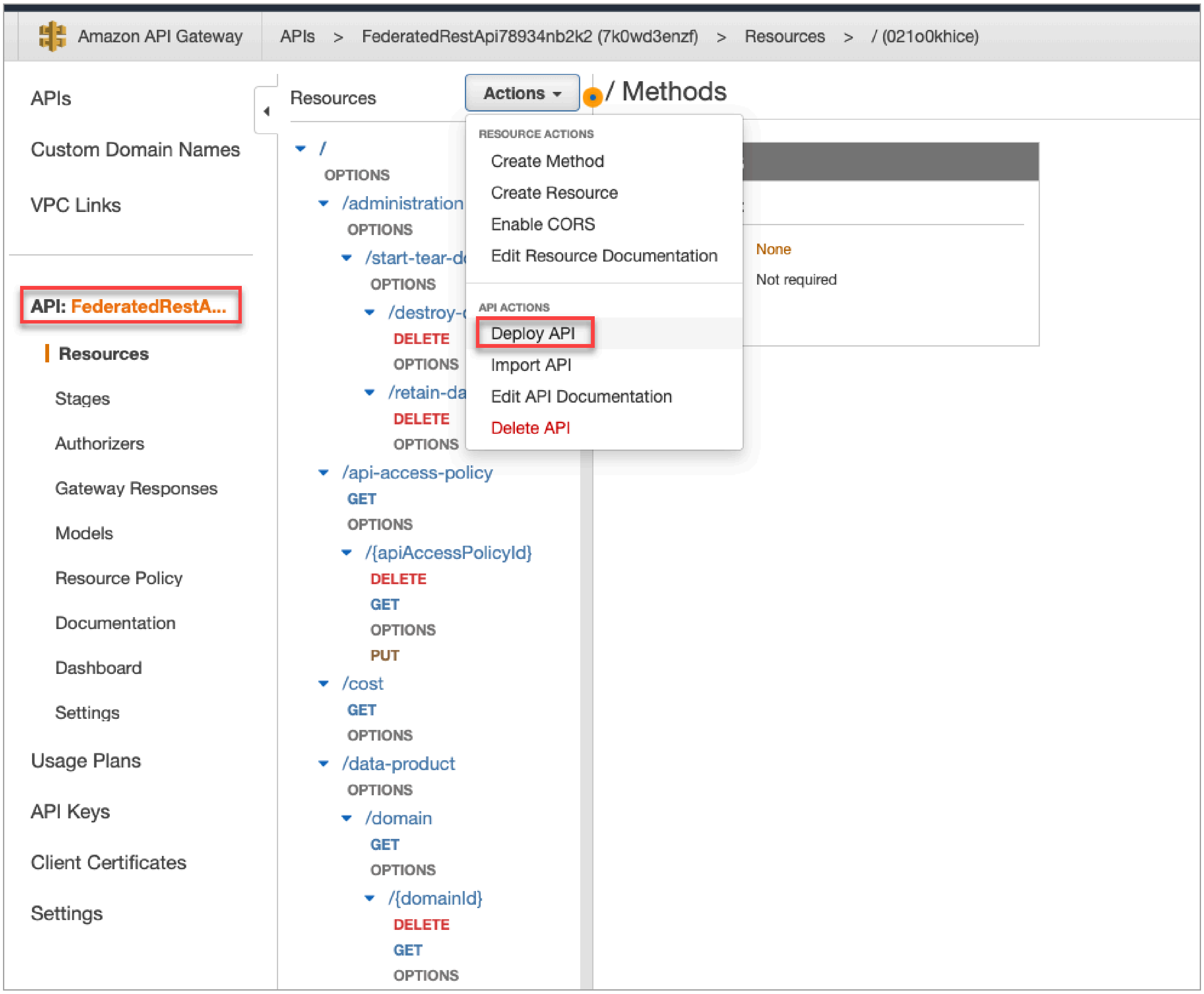
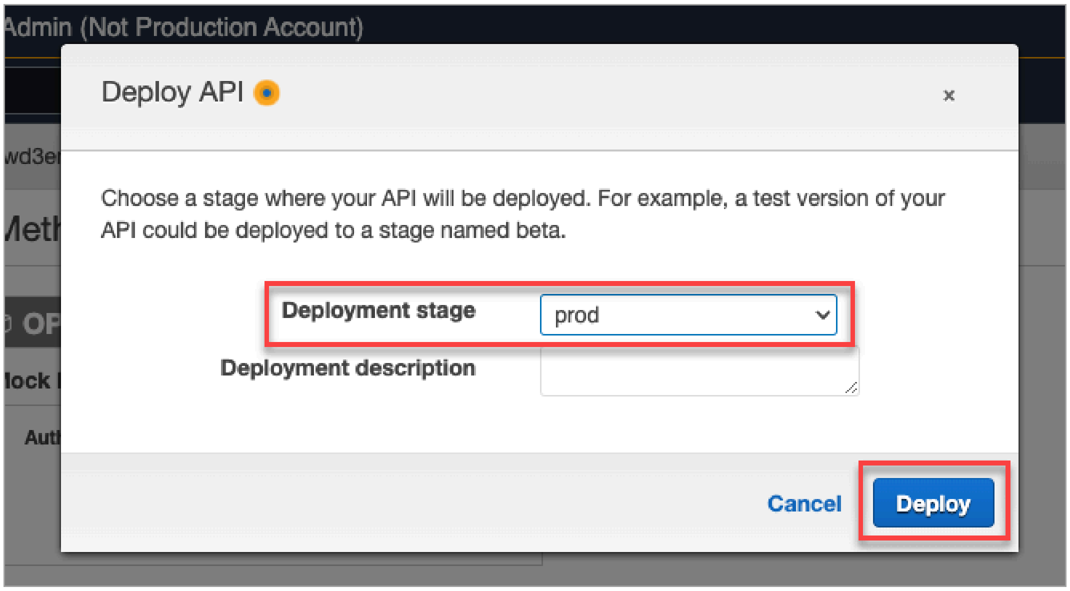


Figure 18: Deploy action for API

4. On the Deploy API dialog, from the deployment stage dropdown, select **prod**, and select **Deploy**. The FederatedRestApi is deployed



Deploy API

Choose a stage where your API will be deployed. For example, a test version of your API could be deployed to a stage named beta.

Deployment stage: prod

Deployment description:

Confirm deploy action

5. To test this has worked, create a new data product and verify the Schema preview is generated for the data source.

Uninstall the solution

You can uninstall the Automated Data Analytics on AWS solution from the deployed web UI with admin permissions (recommended), from the AWS Management Console, or by using the AWS Command Line Interface.

If you uninstall from the AWS Management Console or AWS Command Line Interface, you must manually delete resources retained by the solution, such as Amazon S3 buckets and associated AWS KMS Keys. Hence, we recommend uninstalling the solution from the deployed web UI. For a complete list of resources that will be retained after solution is deleted, from the Ada stack Outputs tab, confirm the *RetainedResources* value before you delete the stack.

This solution provisions additional resources for each existing data product. These dynamically provisioned resources are organised as additional CloudFormation stacks and can be seen in the AWS CloudFormation console, with a prefix "ada-dp-".

If you need to delete or tear down this solution from the AWS CloudFormation console, make sure you delete these data product stacks prior to deleting the Ada main stack. When tearing down the solution from the web UI, these data product stacks will be deleted automatically.

Note

It is recommended to use the web UI teardown feature to uninstall the solution. While you can delete the solution via the AWS Management Console and AWS Command Line Interface, you will need to manually delete all *data product* stacks and retained resources.

Note

Troubleshooting: The first attempt to delete the main Ada stack may fail due to service limits of managing Amazon API Gateway resources. If this occurs, retry deleting the stack from the AWS CloudFormation console. When this occurs, the solution ensures that all resources defined by the stack(s) are removed during the first delete operation, and only the stack(s) themselves remain.

Teardown using the web UI

1. Sign in to the deployed web UI using the *root admin* email specified in deployment parameters.
2. From the navigation, select **Teardown**.
3. From the **Teardown** page, choose one of these following actions.
   a. ***Delete solution*** – This will uninstall all the resources associated with the solution but retain the imported data.
   b. ***Delete solution and data*** – This will uninstall all the resources associated with the solution and destroy the imported data.
4. After confirming one of the actions, you will be presented with details on what resources will be *retained*, if any, and links to the AWS CloudFormation stacks being deleted.
5. Open the links provided in the web UI to follow the status of the deletion via the AWS Management Console.

> **Note**
>
> Make sure to take note of the details provided in the Web UI after deletion starts, as the web UI will become unavailable shortly as the solution is uninstalled.

Using the AWS Management Console

1. Sign in to the AWS CloudFormation console.
2. On the **Stacks** page, filter by stack name “*ada-dp-*“ to get a list of *dynamic infrastructure* created for each *Data Product.*
3. For each of the *dynamic infrastructure* stacks above, select the stack and choose **Delete**. You can delete multiple data product stacks at the same time. You do not need to wait for each to complete before deleting the others. Each stack should take approximately 5 minutes to delete.
4. After you have deleted all *data product* stacks, on the **Stacks** page, choose the Ada stack. If you have any other Ada deployed stacks, you can delete them at the same time.
5. Take note of the *RetainedResourcesExport* output value for the stack as these resources will need to be manually deleted.
6. Choose **Delete**.

7. (Optional) Manually delete the retained resources exported by the *RetainedResourcesExport* output. For more information, refer to the Deleting the retained resources section.

Using AWS Command Line Interface

Determine whether the AWS Command Line Interface (AWS CLI) is available in your environment. For installation instructions, refer to What Is the AWS Command Line Interface in the *AWS CLI User Guide*.

After confirming that the AWS CLI is available, follow the steps above for Using the AWS Management Console and replace each console step with respective AWS CLI command.

Deleting the retained resources

This solution is configured to retain the solution-created Amazon S3 buckets and associated AWS KMS Keys if you decide to delete the AWS CloudFormation stack to prevent accidental data loss. After uninstalling the solution, you can manually delete these resources using the AWS Management Console.

> **Note**
>
> For a complete list of the retained resourced the solution created, open the AWS CloudFormation console and choose the Ada stack, then locate the *RetainedResourcesExport* output value for the stack. If the stack has already been deleted, you can locate the stack by changing filter to **Deleted** in stack list.

1. Sign in to the Amazon S3 console.
2. Delete the KMS keys listed in the *RetainedResourcesExport* section.
3. To delete the S3 bucket using AWS CLI, run the following command:

```
$ aws s3 rb s3://<bucket-name> --force
```

Using the ADA solution

You can access the Automated Data Analytics on AWS solution through its web user interface to ingest, transform, query, and manage datasets. This section provides details about the features of the Automated Data Analytics on AWS and how to use them for working with datasets.

The side navigation pane displays the following options.

Option	Description
Data Products	Import, transform, and manage datasets.
Query Workbench	Query available data products.
Governance	Apply governance controls for accessing and viewing datasets.
Groups	View groups, request access and create new groups.
Users	List of all users.
Admin	*The following options are only available for users with admin access.*
Identity Providers	Allow users to set up and manage external identity providers to access the ADA solution.
Cost Explorer	View service and account level cost breakdowns.
Teardown	Permanently remove the solution from your account. ⓘ **Note** The Teardown option is only available to root_admin users.

Setting up Automated Data Analytics on AWS

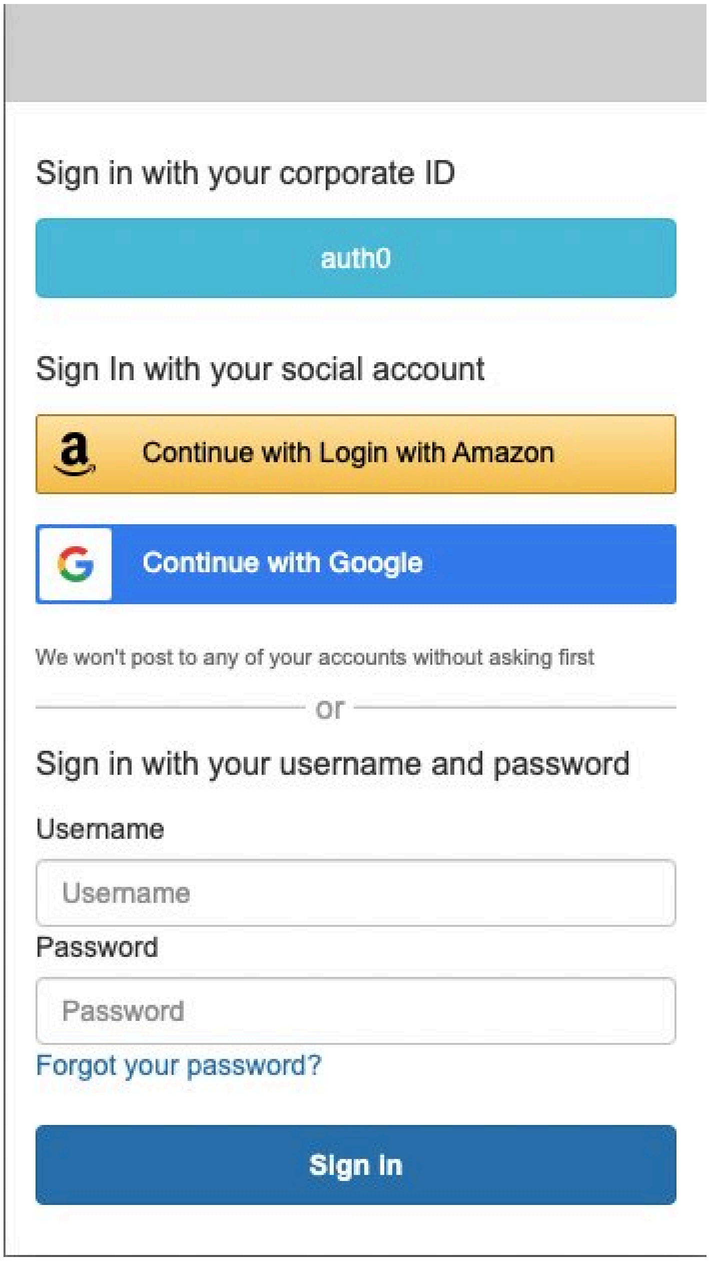
After you have deployed Automated Data Analytics on AWS into your AWS account, you can grant access and permissions to your users. This section describes how an administrator (*admin*) can set up the solution for other users.

- An *admin* is a single root user with full access rights to Automated Data Analytics on AWS. They are responsible for setting up the solution and granting access control policies.
- To enable additional user access to the solution, the admin user must set up ADA to integrate with an external Identity Provider that provides authentication and user directory for other users. They can do this using the web UI. An *Identity Provider or IdP* is a service that stores and manages digital identities.

> **Note**
>
> Only the Admin can set up the federated access with an IDP.

Sign in

1. After you have deployed Automated Data Analytics on AWS into your AWS Account, open the AWS CloudFormation console, and navigate to the **ADA > Outputs** section. Click to open the **WebsiteUrl** shown in the CloudFormation output.
2. The initial username is the adminEmail specified during the deployment. You will receive a temporary password from <`no-reply@verificationemail.com`>.
3. Enter the **Username** and **Password** on the Automated Data Analytics on AWS login screen and select **Sign In**.



Sign in with your corporate ID

auth0

Sign In with your social account

Continue with Login with Amazon

Continue with Google

We won't post to any of your accounts without asking first

or

Sign in with your username and password

Username

Username

Password

Password

Forgot your password?

Sign In

Figure 20: Sign in

Sign in with your Corporate ID

To enable federated sign in from an external Identity Provider such as the company user directory, an admin user must set up the Identity Provider integration using the **Identity Provider** page.

After it is set up, users can see identity provider buttons on the login screen. When users click the identity provider button, it will take them to the identity provider page for authentication and then redirect them back to the solution web UI after logging in successfully.

Sign in with your Social Account

Automated Data Analytics on AWS allows you to natively authenticate at login via Amazon or Google authentication.

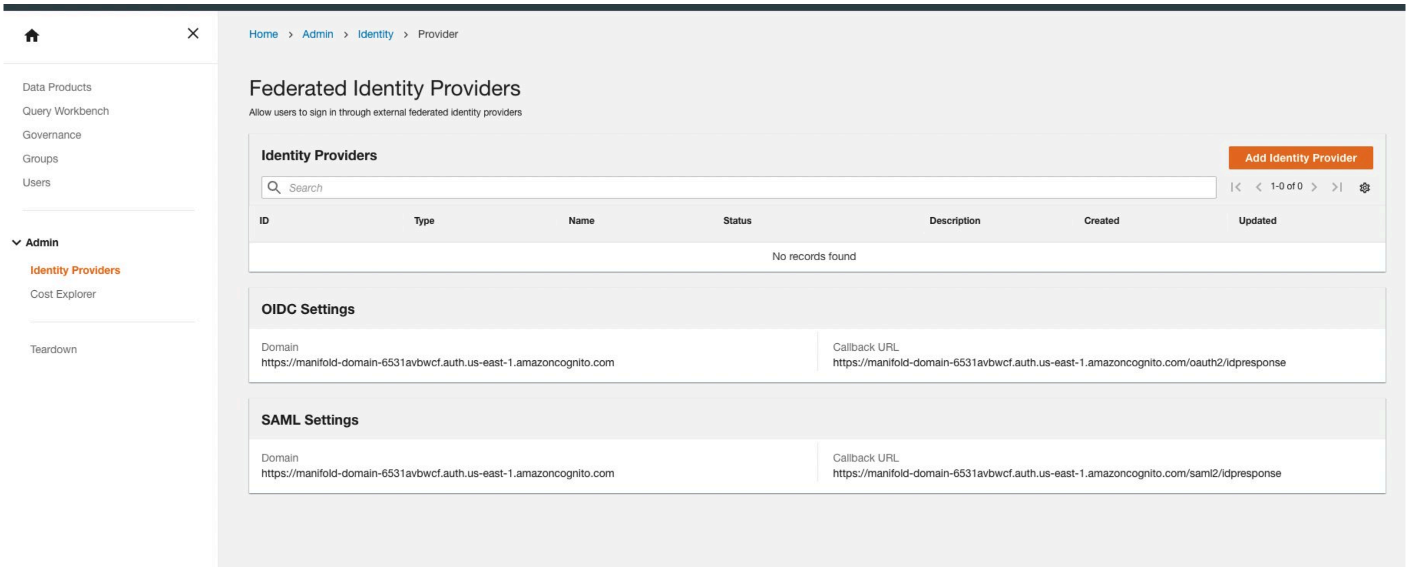
To enable this feature, an admin user must set up Amazon or Google as an external Identity Provider from the **Identity Providers** page, before the user can see and use these login options.

Integrate your Identity Provider (IdP)

This section describes how to enable federate access to Automated Data Analytics on AWS by using your existing Identity Provider.

> **Note**
>
> Automated Data Analytics on AWS does not provide any user management. If you want separate user identities, you must authenticate with an existing IdP.

1. From the Automated Data Analytics on AWS home screen, under **Admin**, select **Identity Providers** from the left side navigation pane.



Federated Identity Providers

Allow users to sign in through external federated identity providers

Identity Providers

ID	Type	Name	Status	Description	Created	Updated
No records found						

OIDC Settings

Domain
https://manifold-domain-6531avbwcf.auth.us-east-1.amazoncognito.com

Callback URL
https://manifold-domain-6531avbwcf.auth.us-east-1.amazoncognito.com/oauth2/idpresponse

SAML Settings

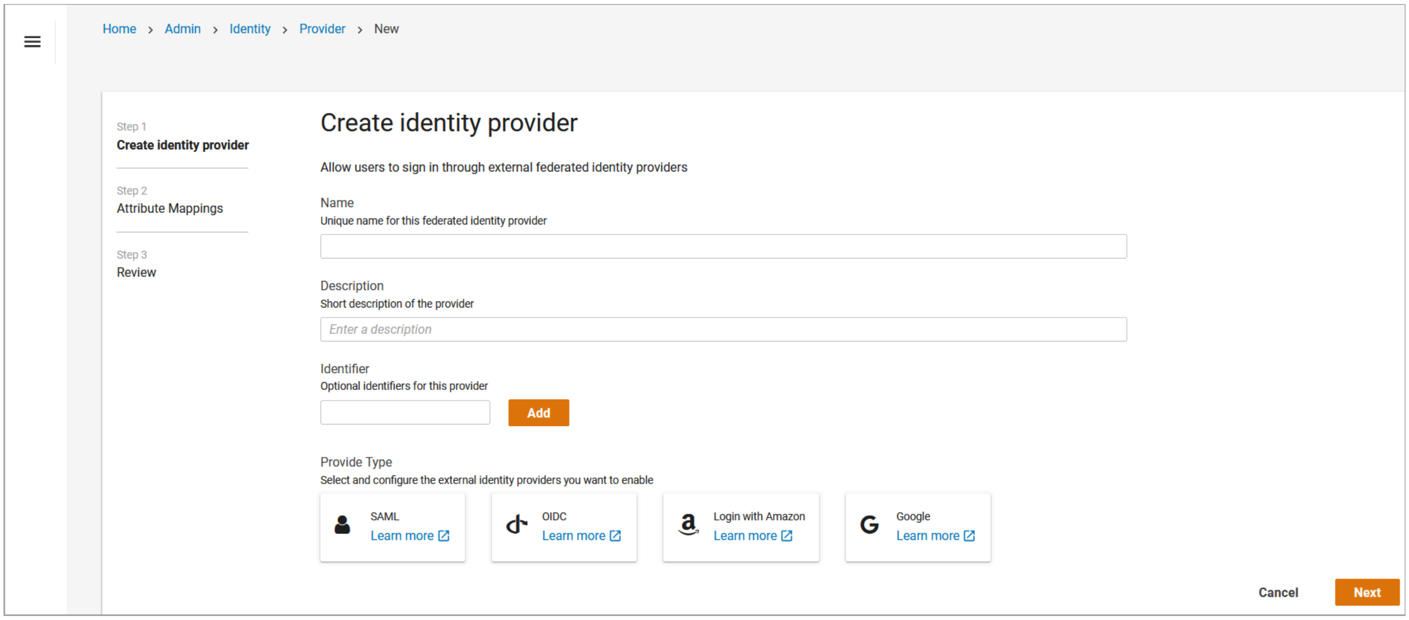
Domain
https://manifold-domain-6531avbwcf.auth.us-east-1.amazoncognito.com

Callback URL
https://manifold-domain-6531avbwcf.auth.us-east-1.amazoncognito.com/saml2/idpresponse

List of identity providers

2. On the Federated Identity Providers page, select **Add Identity Provider**.

3. Enter a name and description for the IdP along with any additional identifiers. Automated Data Analytics on AWS offers four options of authentication providers: SAML, OIDC, Amazon, and Google.



Create identity provider

Allow users to sign in through external federated identity providers

Name
Unique name for this federated identity provider

Description
Short description of the provider
Enter a description

Identifier
Optional identifiers for this provider

Provide Type
Select and configure the external identity providers you want to enable

Figure 22: Add a new identity provider

4. Fill in the requested information and select **Next**. For more information, refer to these documentation resources.

   - SAML (SAML 2.0)
   - OIDC
   - Amazon
   - Google

5. After you have configured your IdP, you need to map user attributes with the corresponding Cognito User Pools. To do this, specify a human-readable claim to identify different users, along with any additional attributes such as email, phone number, address, etc.

6. Fill in descriptions and select **Next.**

7. Review the completed information and select **Submit.**

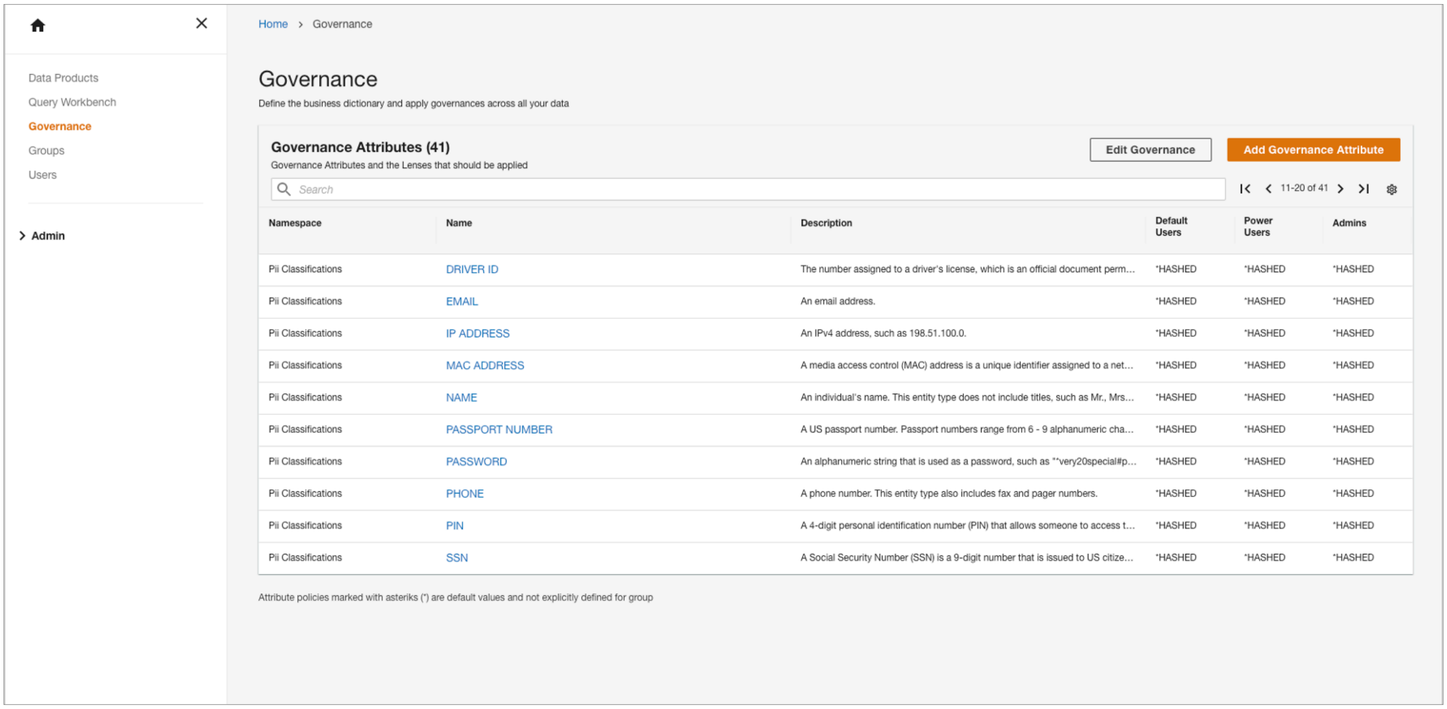
8. Automated Data Analytics on AWS will connect to your IdP to provide federated access to authorized users.

Setting up governance

The Automated Data Analytics on AWS solution provides granular row/column level controls through the Governance feature. For more information, refer to the Governance service section.

> **Note**
>
> Only Admins and users with specified privileges can modify Governance controls.

To access the Governance feature, from the Automated Data Analytics on AWS home screen, on the left menu, select **Governance**.



Governance

Define the business dictionary and apply governances across all your data

Governance Attributes (41)

Governance Attributes and the Lenses that should be applied

Namespace	Name	Description	Default Users	Power Users	Admins
Pii Classifications	DRIVER ID	The number assigned to a driver's license, which is an official document perm...	*HASHED	*HASHED	*HASHED
Pii Classifications	EMAIL	An email address.	*HASHED	*HASHED	*HASHED
Pii Classifications	IP ADDRESS	An IPv4 address, such as 198.51.100.0.	*HASHED	*HASHED	*HASHED
Pii Classifications	MAC ADDRESS	A media access control (MAC) address is a unique identifier assigned to a net...	*HASHED	*HASHED	*HASHED
Pii Classifications	NAME	An individual's name. This entity type does not include titles, such as Mr., Mrs...	*HASHED	*HASHED	*HASHED
Pii Classifications	PASSPORT NUMBER	A US passport number. Passport numbers range from 6 - 9 alphanumeric cha...	*HASHED	*HASHED	*HASHED
Pii Classifications	PASSWORD	An alphanumeric string that is used as a password, such as "*very20special#p...	*HASHED	*HASHED	*HASHED
Pii Classifications	PHONE	A phone number. This entity type also includes fax and pager numbers.	*HASHED	*HASHED	*HASHED
Pii Classifications	PIN	A 4-digit personal identification number (PIN) that allows someone to access t...	*HASHED	*HASHED	*HASHED
Pii Classifications	SSN	A Social Security Number (SSN) is a 9-digit number that is issued to US citize...	*HASHED	*HASHED	*HASHED

Attribute policies marked with asteriks (*) are default values and not explicitly defined for group

Governance page

Governance attributes

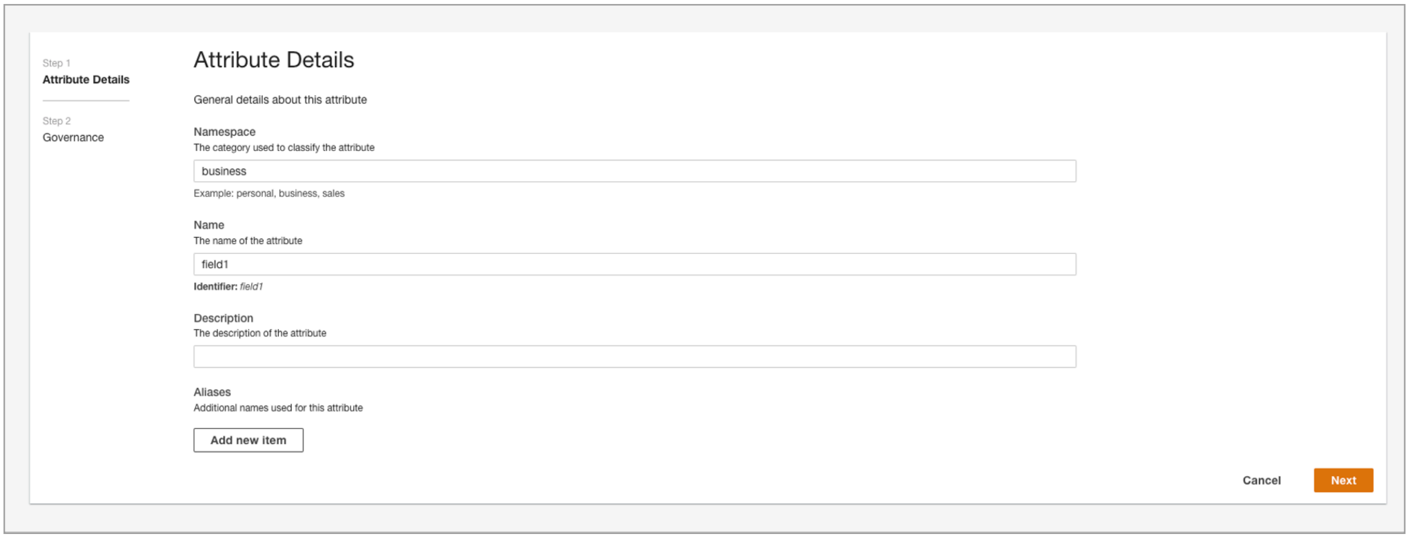
Governance attributes are business definitions for data with assigned group level controls. You can access, edit, and create new attributes through the **Governance** page.

By default, Automated Data Analytics on AWS provides a list of standard governance attributes for PII such as Name, Email, and Phone Number. Users can find these governance attributes using the Search bar on the **Governance Attributes** page.

Create custom governance attributes

To add a new governance attribute:

1. On the Governance page, select **Add Governance Attribute**.



Step 1
Attribute Details

Step 2
Governance

Attribute Details

General details about this attribute

Namespace
The category used to classify the attribute
business
Example: personal, business, sales

Name
The name of the attribute
field1
Identifier: *field1*

Description
The description of the attribute

Aliases
Additional names used for this attribute

2. On the details page, enter a namespace, name, description, and additional aliases (if required).

> **Note**
>
> You can apply a governance attribute to multiple data products. When naming an attribute, we recommend keeping it generic, like location or email without relating it to a data product.

3. Confirm the attribute details, and select **Next**.
4. On the Governance page, select the appropriate governance level for the Default Lens:
   - **Clear** - See the data as is.
   - **Hidden** - Hide the data completely. Users will not see data in search results.
   - **Hashed** - Create a consistent, tokenized hash for data. Users will see the same value and can create relationships with other datasets without exposing the underlying data.

   > **Note**
   >
   > When setting up **Default Lens**, the least permissive governance level will be applied for the query executions when multiple policies are applicable. For example, for the default lens, choose the most permissive level.

5. In the search box, search user group by group name and choose **Add Governance Settings** to view and set up governance policy for the user group. You can set both Column level policy and Row level policy for your groups.

6. To proceed, select **Submit**.

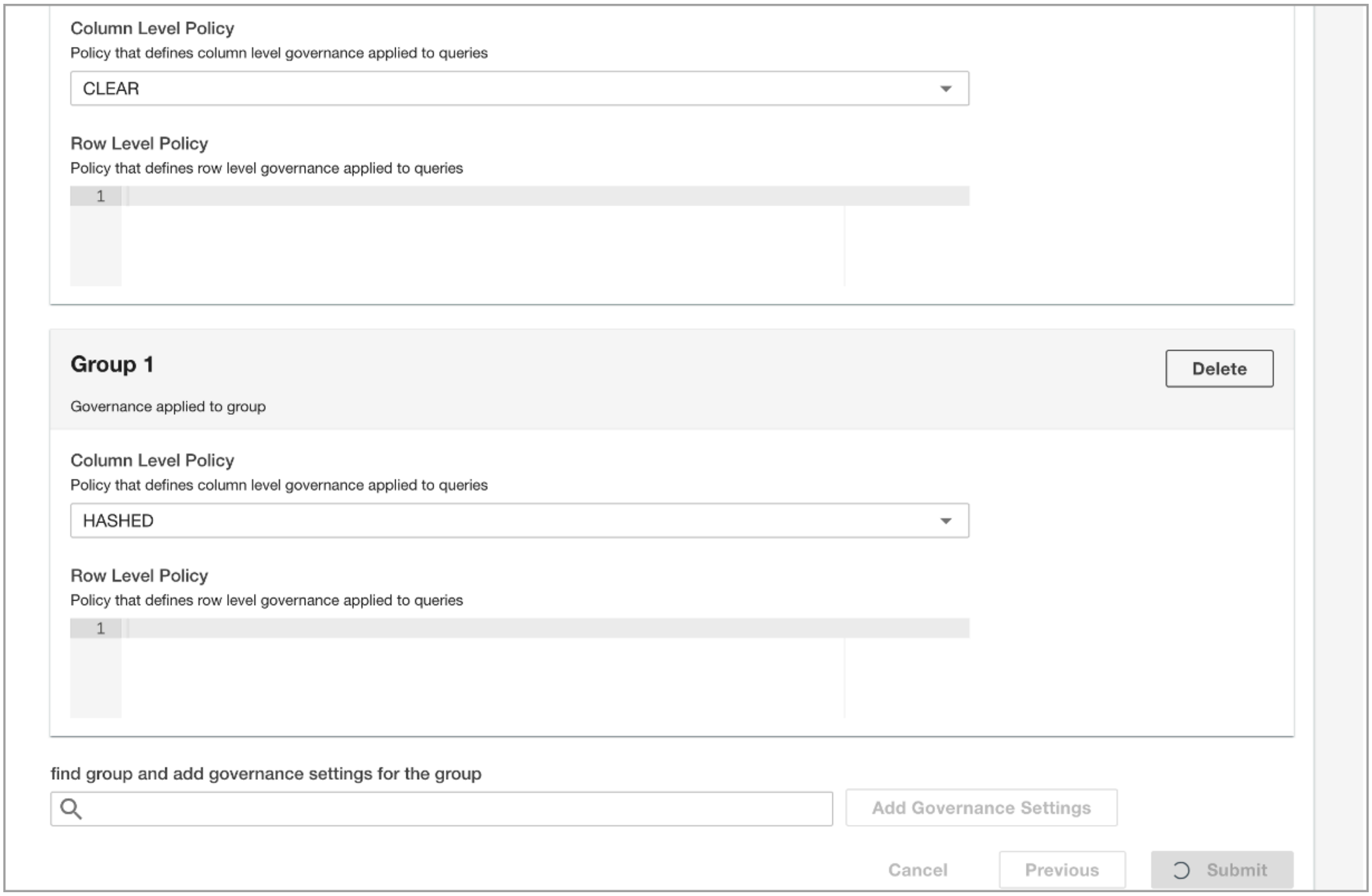
Column level policy

To set up a column-level policy, select a value from **Column Level Policy** dropdown:

- **Clear** - See the data as is.
- **Hidden** - Hide the data completely. Users will not see data in search results.
- **Hashed** - Create a consistent, tokenized hash for data. Users will see the same value and can create relationships with other datasets without exposing the underlying data.

> **Note**
>
> When setting up **Default Lens**, the least permissive governance level will be applied for the query executions when multiple policies are applicable. For example, for the default lens, choose the most permissive level.

A user may belong to multiple user groups. In this case, the least permissive policy will be used when running a query. For example, if a user belongs to Group 1 and Group 2, and group 1 has **Hashed** column level policy, while Group 2 has **Clear** column level policy, and the Default Lens has a **Hidden** column level policy, then the user will have **Clear** column level policy applied while running the query, as the order of policies is **Clear > Hashed > Hidden**.



Column Level Policy
Policy that defines column level governance applied to queries
CLEAR

Row Level Policy
Policy that defines row level governance applied to queries

Group 1

Governance applied to group

Column Level Policy
Policy that defines column level governance applied to queries
HASHED

Row Level Policy
Policy that defines row level governance applied to queries

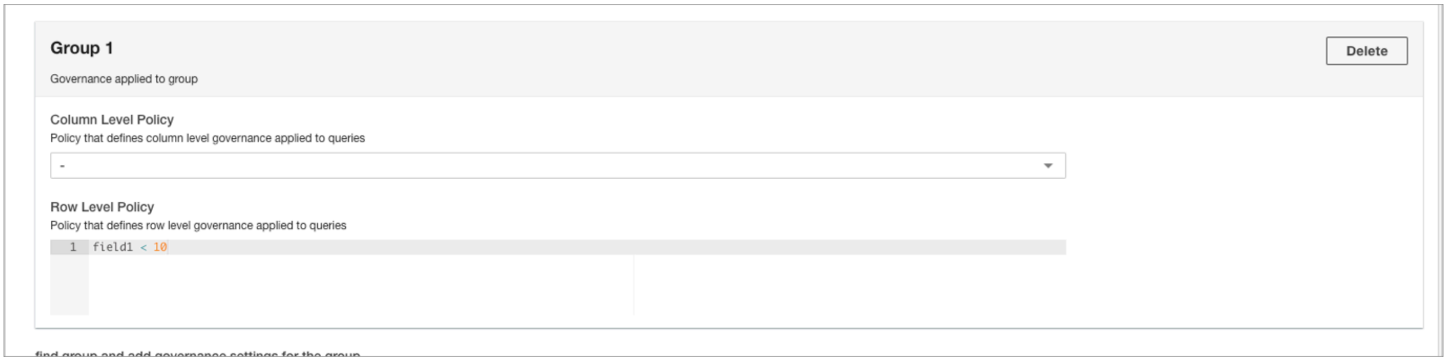
find group and add governance settings for the group

Column Level Policy options

Row level policy

To set up a row level policy, you must define a filtering statement. The filtering statement is used in the **Where** clause of the SQL query statement to filter out the matching rows

For example, to allow all users in group 1 to only see rows with referenced column value less than 10, enter `attribute_name < 10` in the Row level policy under the group 1 panel. The `attribute_name` should match the attribute name, and will be replaced by the actual mapping column name when the query is run. The grammar of the filtering statement should follow the **Where** clause of a SQL statement and you can only use the attribute name to refer to a column.



Group 1

Governance applied to group

Column Level Policy

Policy that defines column level governance applied to queries

-

Row Level Policy

Policy that defines row level governance applied to queries

```
field1 < 10
```

Row Level Policy options

A user can belong to multiple user groups. In this case, the union of filtering statements is used in running the query. For example, if a user belongs to Group 1 and Group 2, and Group 1 has `attribute_name = ‘UK’` row level policy attached and group 2 has `attribute_name = ‘US’` row level policy attached. When the query is run, the user will see rows with attribute value equal to UK or US, as the SQL statement equivalent will be `attribute_name = ‘UK’` or `attribute_name = ‘US’`.

Apply governance attributes

> **Note**
>
> We recommend setting up column level and row level permissions before granting entity level permission to avoid data leakage. For more information, refer to the Permissions section.

For more information on how to apply the governance attributes to an entity column, refer to the Schema section.

Creating data products

With Automated Data Analytics on AWS, you can join datasets from different source locations and file types by creating data products.

- A *dataset* is a singular collection of data, such as a database table.
- A *Data Product* is a dataset that has successfully been imported into Automated Data Analytics on AWS and is ready to be queried.

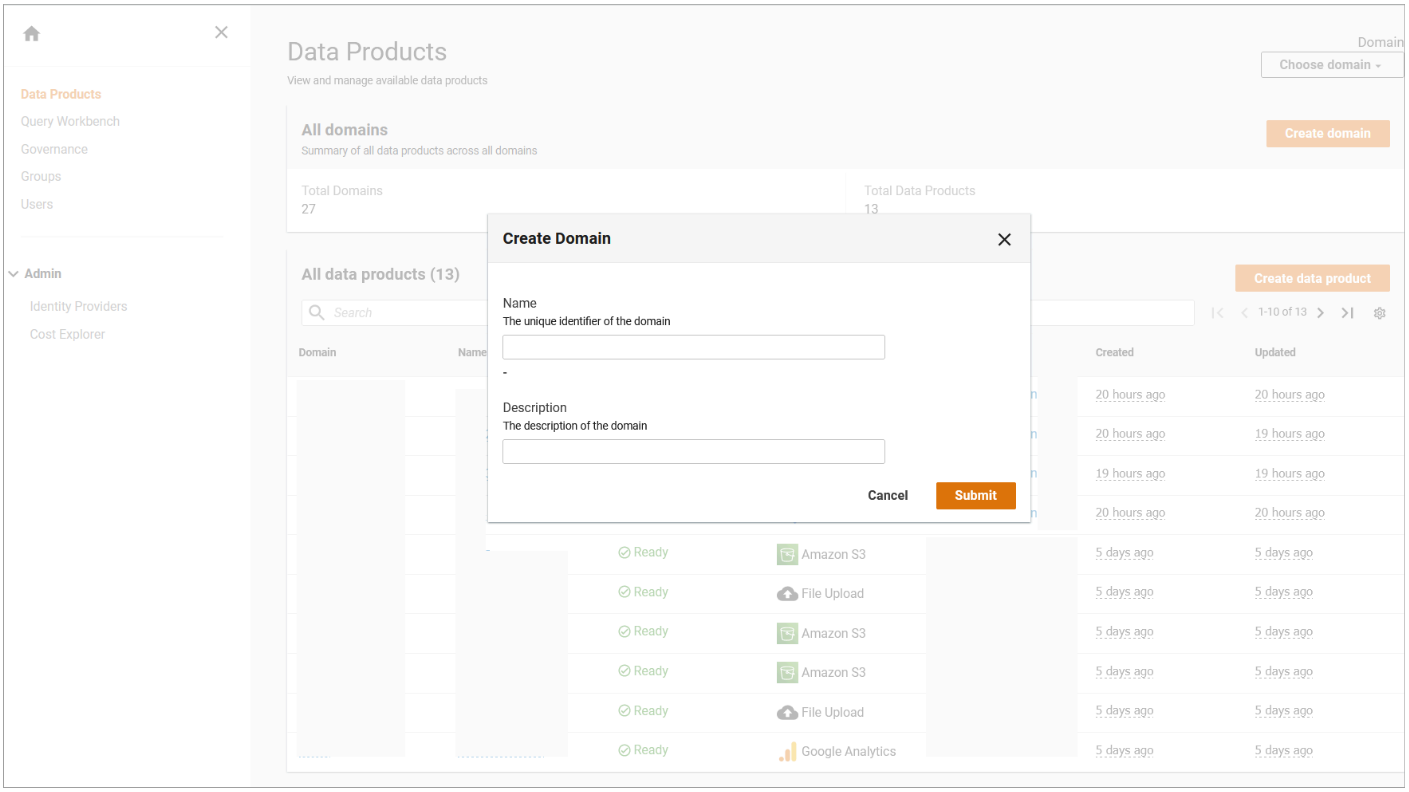
In Automated Data Analytics on AWS, you must create a Data Product in order to query a dataset.

Step 1: Create a domain

In Automated Data Analytics on AWS, a *domain* is a user defined group of data products. For example, this might be a team or a project. Domains are used as a structured way for users to access data products. Before you create a Data Product, you must first create a Domain.

The following steps describe the process of creating a domain.

1. On the **Data Products** screen, select **Create Domain.** The **Create Domain** dialog is displayed.



Create a new domain

2. On the **Create Domain** dialog, enter a Domain Name and Description, and select **Submit**. You can also add tags to the Domain and use them to give context to the domain.

   - A Domain Name must have at least 2 characters and cannot exceed 2,048 characters. Additionally, the Domain Name must start with a letter and cannot contain and special such as colons, asterisks, or exclamation marks.
   - A *tag* is a user specified piece of data that makes it easier to identify the resources.Tags used throughout the Automated Data Analytics on AWS solution to enhance the search functionality. For more information, refer to the Search section.

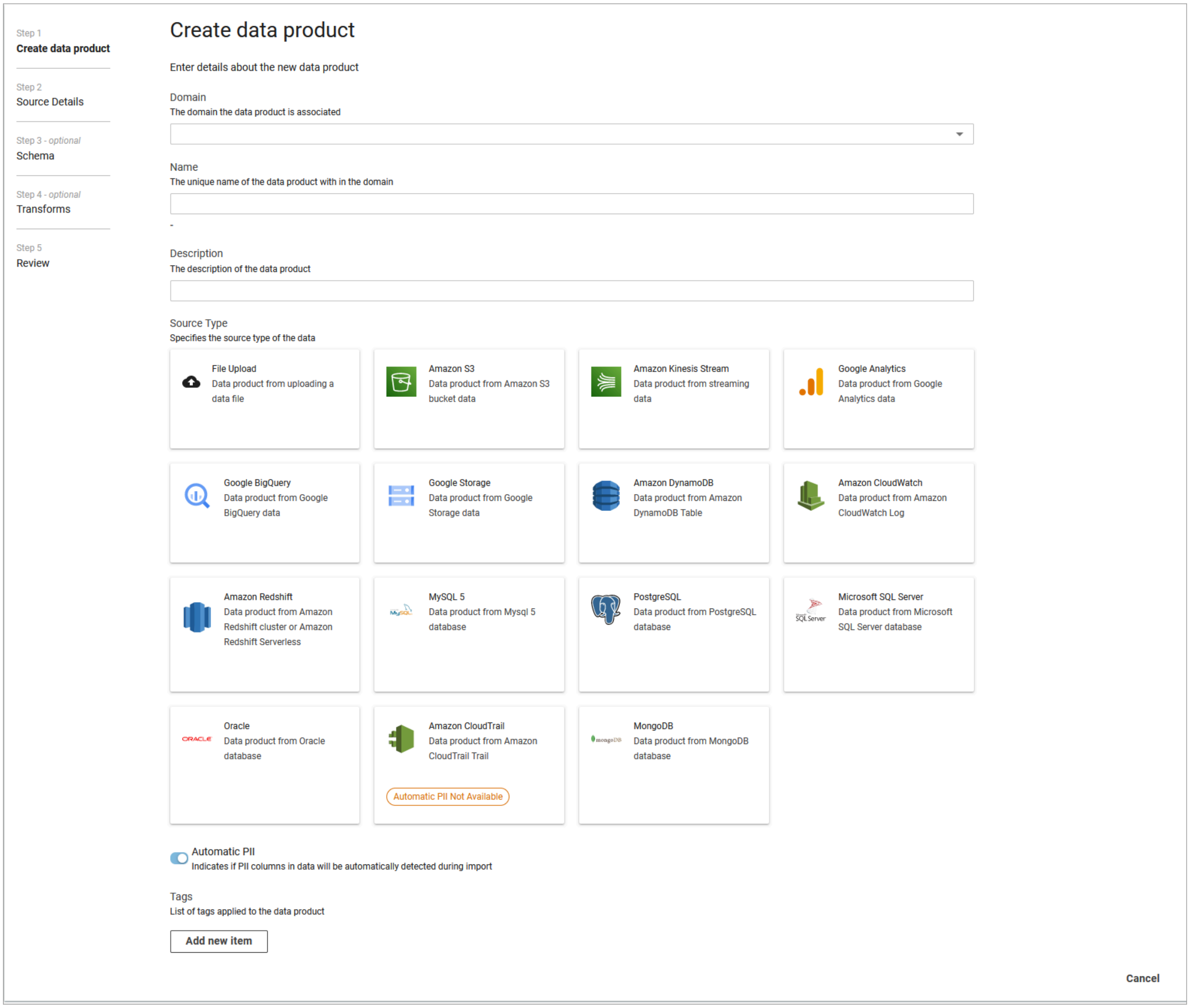
Step 2: Create a new data product

Before you create a data product, ensure you have created a domain. The following steps describe how to create a data product.

1. Using the left side navigation pane, select **Data Products.**
2. On the Data Products page, under the Data Products section, select **Create data product**.
3. On the **Create data product** page, choose the domain you have previously created.
4. Add a name, description, and any tags.
5. Select **Source.** For detailed instructions on data connectors, refer to the Data connectors guide.
    - A *source* is the origin of a dataset. You can import data from Amazon S3, Amazon Kinesis Stream, Amazon CloudWatch, File Upload, Google Cloud Storage, Google Analytics, Google BigQuery, MySQL5, PostgreSQL, Microsoft SQL Server, DynamoDB, MongoDB, or CloudTrail.

    > **Note**
    >
    > Automated Data Analytics on AWS will not grant itself, or its users', permissions to access resources, maintaining the defined access policies of the source resource. Access to source data must be granted by the maintainer of the source resource outside of the solution to allow the solution to read the source for purposes of creating data products from source and federating queries including source within Automated Data Analytics on AWS. Whenever the solution accesses an AWS-based source, it applies principal tags which specify the Automated Data Analytics on AWS service, groups, and user performing the action.

6. Many businesses interact with datasets that contain sensitive customer information like emails, credit card numbers, or passport numbers. This is often referred to as Personally Identifiable Information (PII) and must be only be viewed by privileged users. To automatically detect PII, select **Automatic PII Detection** on the upper right hand side of the screen. This scans your dataset for any potentially PII information and applies the user defined Governance controls. For more information on these controls, refer to the Governance section.



Step 1
Create data product

Step 2
Source Details

Step 3 - optional
Schema

Step 4 - optional
Transforms

Step 5
Review

Create data product

Enter details about the new data product

Domain
The domain the data product is associated

Name
The unique name of the data product with in the domain

Description
The description of the data product

Source Type
Specifies the source type of the data

File Upload
Data product from uploading a data file

Amazon S3
Data product from Amazon S3 bucket data

Amazon Kinesis Stream
Data product from streaming data

Google Analytics
Data product from Google Analytics data

Google BigQuery
Data product from Google BigQuery data

Google Storage
Data product from Google Storage data

Amazon DynamoDB
Data product from Amazon DynamoDB Table

Amazon CloudWatch
Data product from Amazon CloudWatch Log

Amazon Redshift
Data product from Amazon Redshift cluster or Amazon Redshift Serverless

MySQL 5
Data product from Mysql 5 database

PostgreSQL
Data product from PostgreSQL database

Microsoft SQL Server
Data product from Microsoft SQL Server database

Oracle
Data product from Oracle database

Amazon CloudTrail
Data product from Amazon CloudTrail Trail
Automatic PII Not Available

MongoDB
Data product from MongoDB database

Automatic PII
Indicates if PII columns in data will be automatically detected during import

Tags
List of tags applied to the data product

Create a new data product

7. Select **Next**. The **Source Details** page is displayed.

Step 3: Provide source details

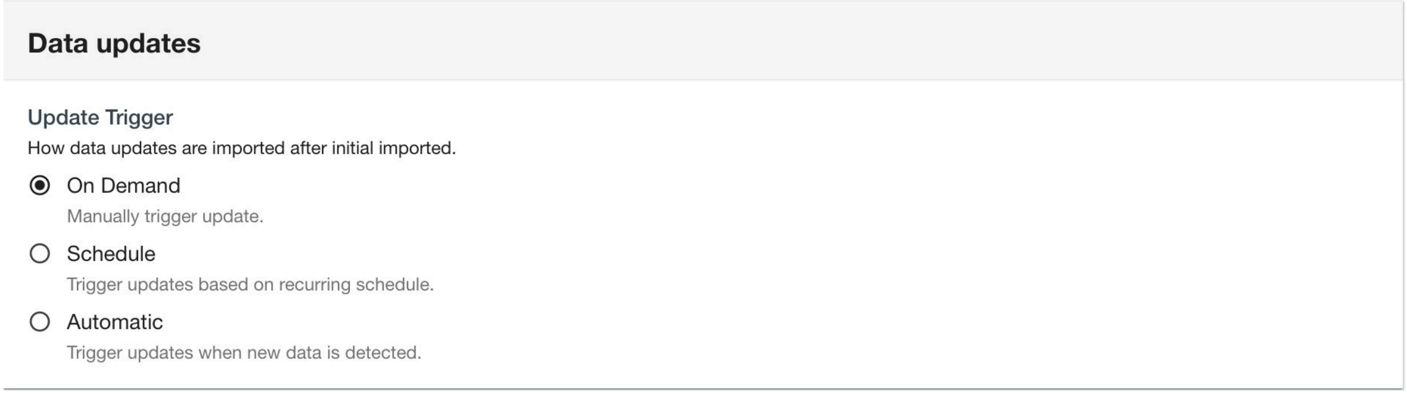
For importing a dataset, you will need to provide access to the data source and specify the update frequency.

1. On the **Source Details** page, enter the source details.

Note

The source details will vary depending on the type of source selected in the previous step. For detailed instructions on data connectors, refer to the Data connectors guide.

2. Under Data updates, select the **Update Trigger** for your source.



Data updates

Update Trigger

How data updates are imported after initial imported.

● On Demand

Manually trigger update.

○ Schedule

Trigger updates based on recurring schedule.

○ Automatic

Trigger updates when new data is detected.

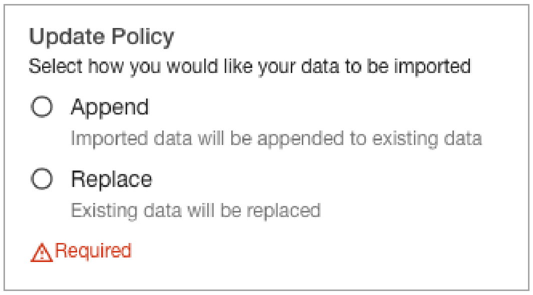
Update data trigger

- Automatic
  - The Automatic option will refresh data continuously.
- On Demand
  - The On Demand option will only import the data at the time of creating the new data pproduct. Once the data product has been created, you can refresh the data by choosing **Start Data Update** in the dataproduct table or making a call via the Automated Data Analytics on AWS API.
  - For Google Analytics, when user selects the On Demand trigger type, they need to configure the start and end date for the import process.
- Schedule
  - Use the Schedule option to choose the interval you would like data to be refreshed. This includes hourly, daily, weekly, monthly, or a custom window.
  - For the Google Analytics connector, Automated Data Analytics on AWS only supports daily, weekly and monthly scheduled frequency. Users can choose append or replace as the update policy as part of the scheduling.

Note

Depending on the trigger selected, your AWS costs for operating the Automated Data Analytics on AWS solution may vary.

3. Select the Update Policy for your data source.



Update Policy
Select how you would like your data to be imported

○ Append
Imported data will be appended to existing data

○ Replace
Existing data will be replaced

⚠Required

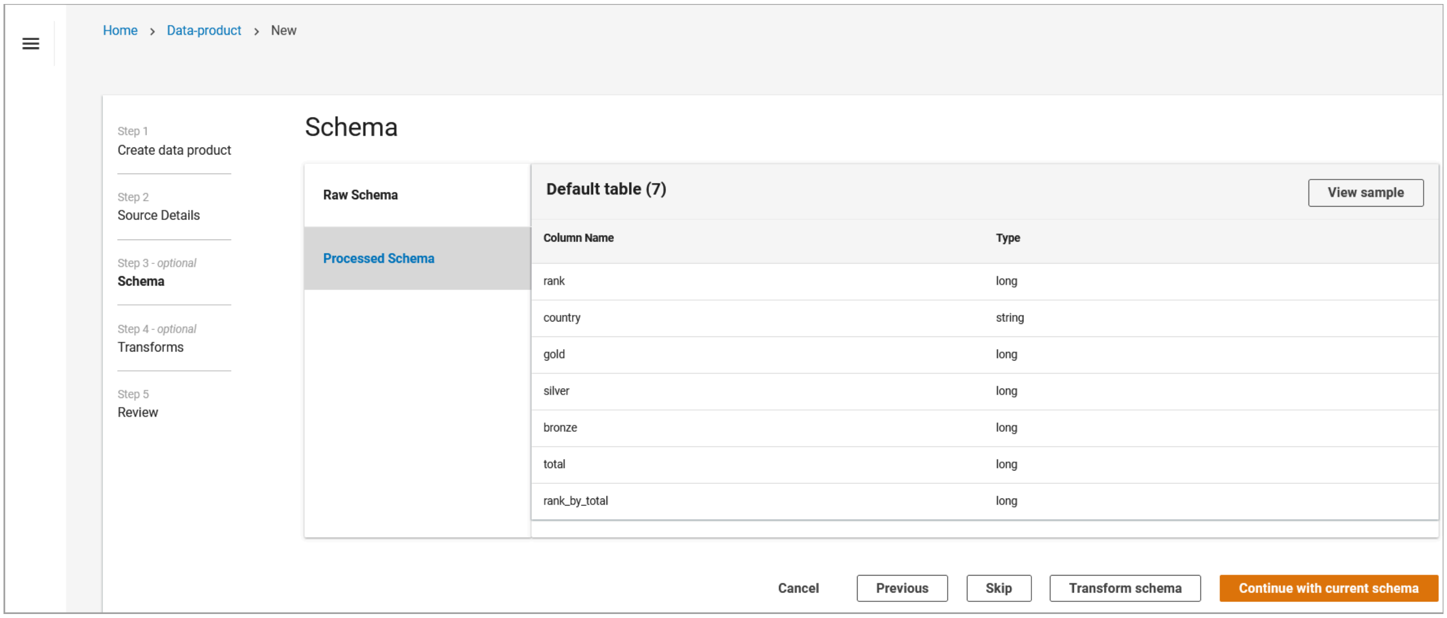
Update policy

- **Append**: If you select the **Append** update policy for your data product, every time new data is created in the data source (e.g. new data files are dropped into S3 bucket), ADA will only process the incoming data and append it to the existing dataset. Use this option when the data source has new data added to the data source regularly.
- **Replace**: If you select the **Replace** update policy, every time ADA updates a data product from the data source, it will re-import all the data from the data source. Use this option when you want the source data to be regularly updated in an existing dataset.

4. Select **Next.** The **Schema** page is displayed.

Step 4: Review data schema

After entering source details, Automated Data Analytics on AWS will scan your data to generate a preview of its schema. This is displayed as **Transformed Schema.**



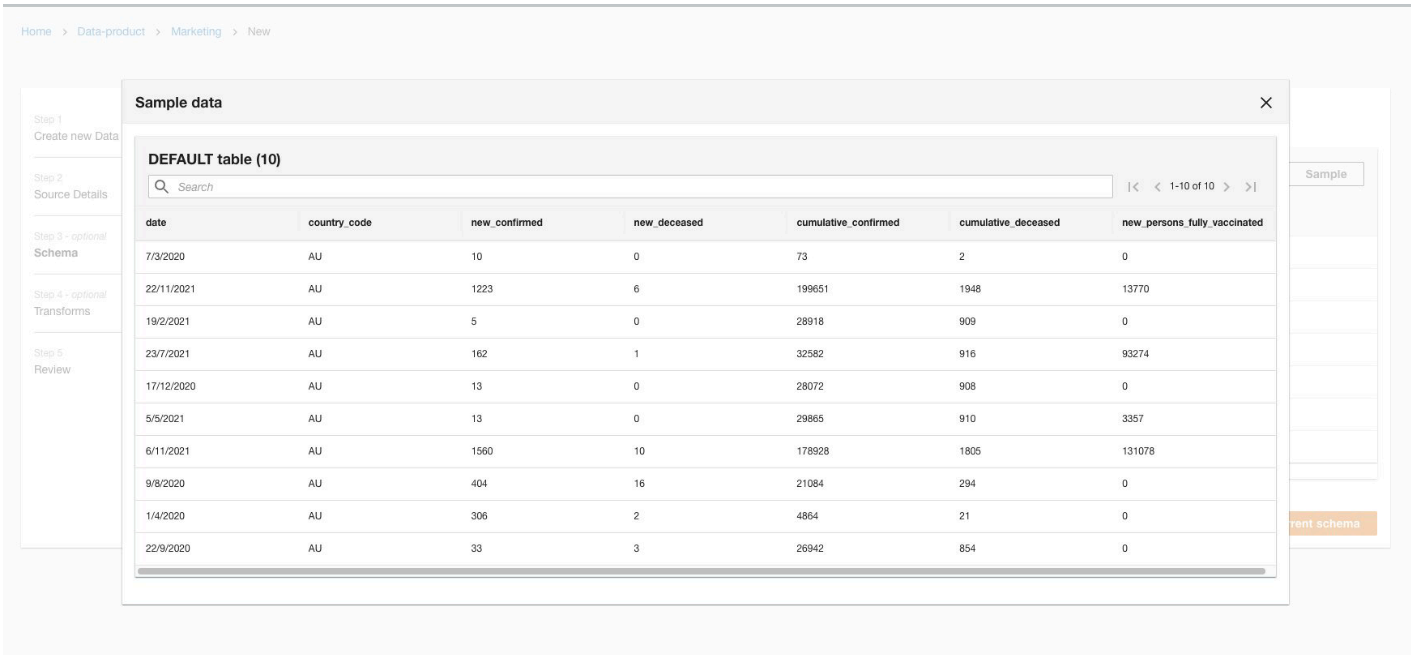
Schema

Default table (7)

Column Name	Type
rank	long
country	string
gold	long
silver	long
bronze	long
total	long
rank_by_total	long

Transformed schema page

- You can use the schema preview to inspect the sample data. To do this, select **Sample**.
- You might receive a notification stating there may be PII data in the dataset. Users must confirm that the dataset does not contain PII or they have privileged access rights. To confirm, select **Agree.**
- The sample data displays a table containing the first 10 rows of the dataset.



Sample data

DEFAULT table (10)

date	country_code	new_confirmed	new_deceased	cumulative_confirmed	cumulative_deceased	new_persons_fully_vaccinated
7/3/2020	AU	10	0	73	2	0
22/11/2021	AU	1223	6	199651	1948	13770
19/2/2021	AU	5	0	28918	909	0
23/7/2021	AU	162	1	32582	916	93274
17/12/2020	AU	13	0	28072	908	0
5/5/2021	AU	13	0	29865	910	3357
6/11/2021	AU	1560	10	178928	1805	131078
9/8/2020	AU	404	16	21084	294	0
1/4/2020	AU	306	2	4864	21	0
22/9/2020	AU	33	3	26942	854	0

Display data

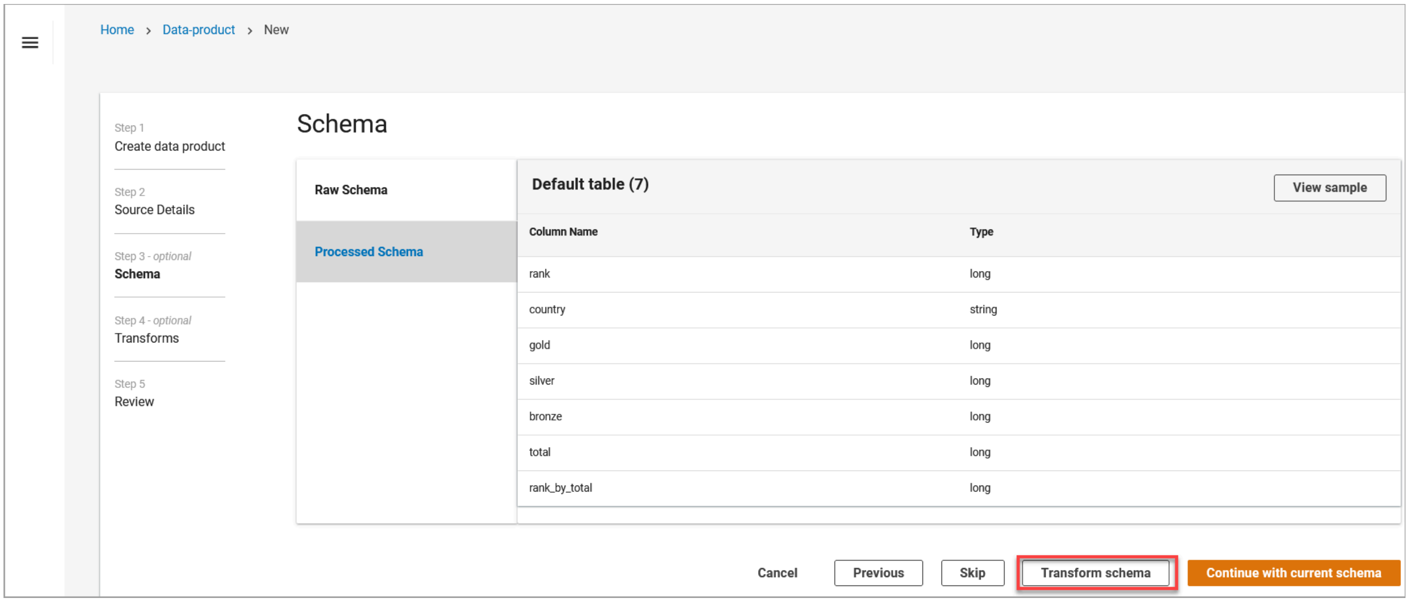
- Once you review the data, you can either use the inferred schema or transform the schema.
  - If the schema looks correct, select **Continue with Current Schema**, and **Submit.** This will start the workflow to import the data, making it available to be queried.
  - If you want to modify the data using Transforms, refer to the section.

Step 5: Transform your schema

During the dataset import process, you can modify your dataset through transforms.

A *transform* is the process of converting data from one format to another. In Automated Data Analytics on AWS, this helps to clean the data into a useable format to run queries.

1. To use Transform, on the Schema Preview page, select **Transform Schema**.



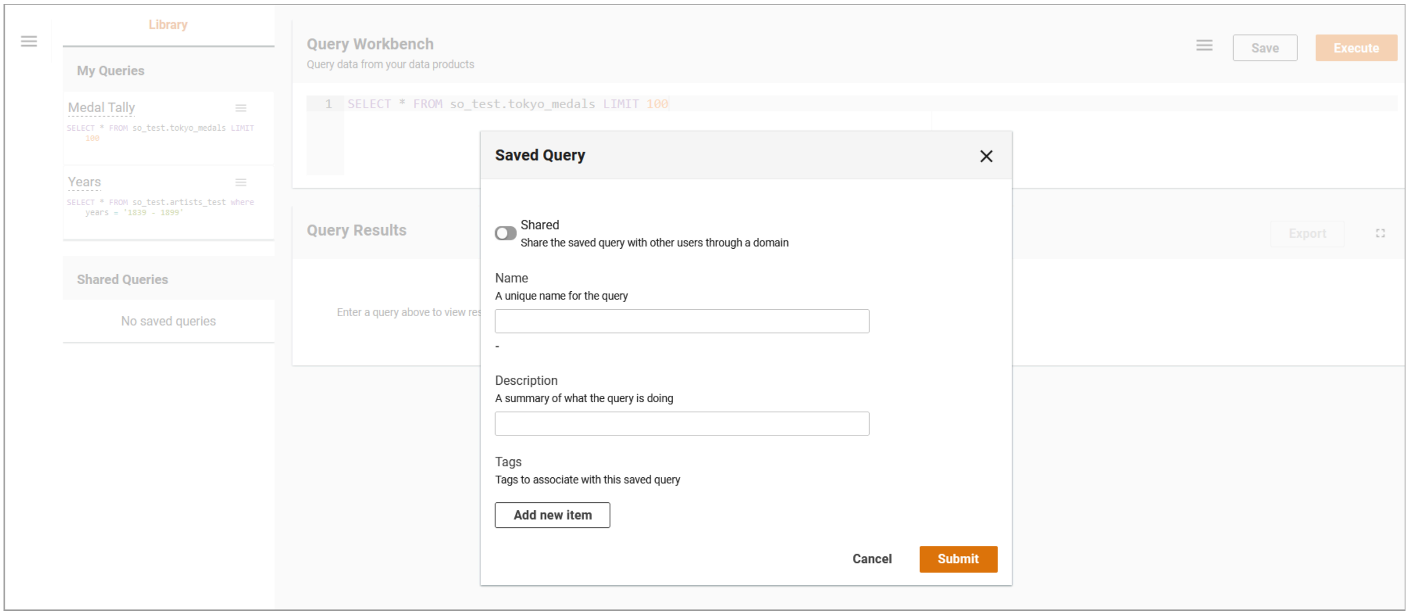
Default table (7)

Column Name	Type
rank	long
country	string
gold	long
silver	long
bronze	long
total	long
rank_by_total	long

Transform schema

Automated Data Analytics on AWS has 4 default transforms: **Apply Mapping, Drop Fields, JSON Relationalize, Select Fields**. You can use these transforms through the Transform plan screen. Transformations can be applied individually, or multiple transformations can be layered in a sequence.

2. Drag the desired transform from the left hand panel into the center panel.



```
SELECT * FROM so_test.tokyo_medals LIMIT 100
```

```
SELECT * FROM so_test.artists_test where years = '1839 - 1899'
```

```
1  SELECT * FROM so_test.tokyo_medals LIMIT 100
```

Saved Query

Shared
Share the saved query with other users through a domain

Name
A unique name for the query

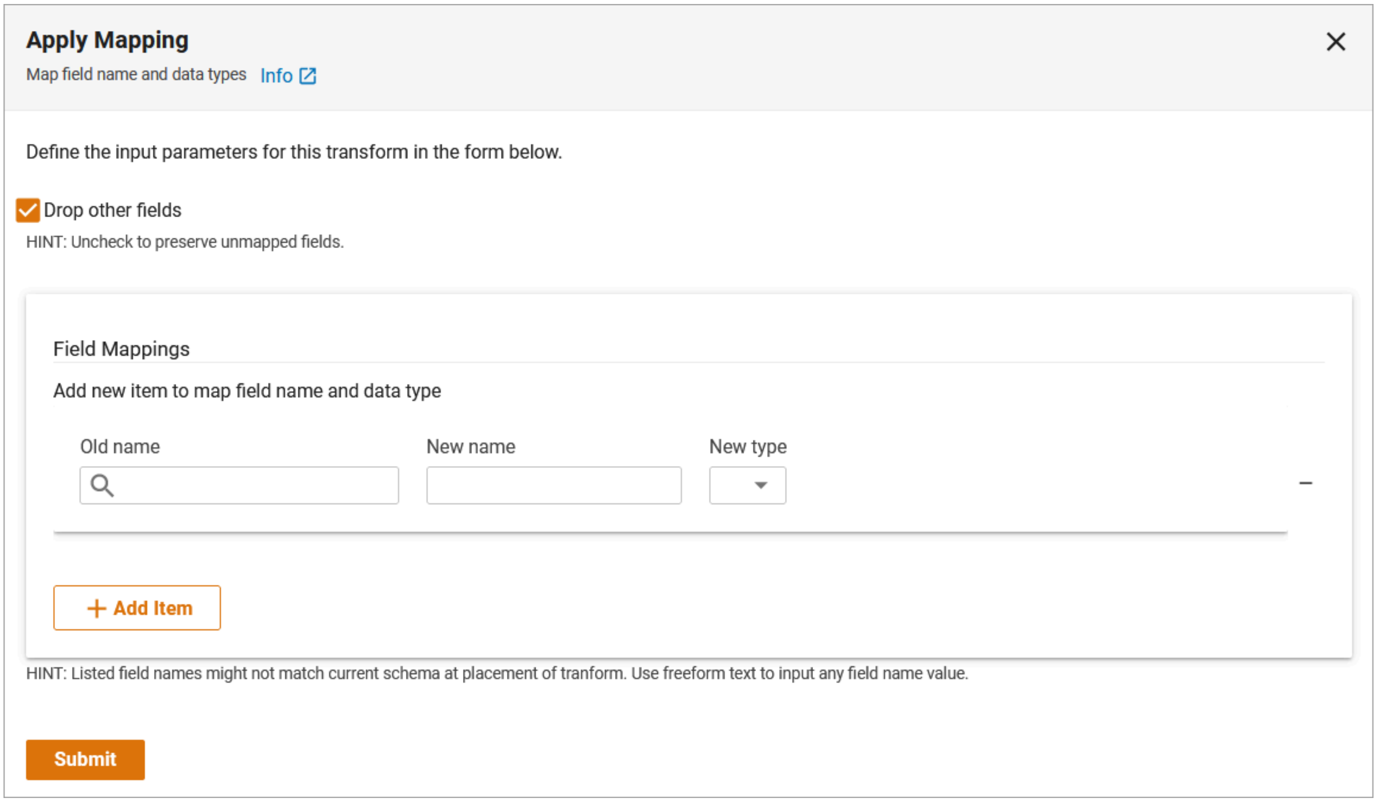
-

Description
A summary of what the query is doing

Tags
Tags to associate with this saved query

Saved query dialog

- You can use **Apply Mapping** to modify the name and data type for columns i.e. string, date, integer, etc. To do this, select **Add new item.** The Apply Mapping Input Parameters dialog box is displayed.



Apply Mapping

Map field name and data types

Define the input parameters for this transform in the form below.

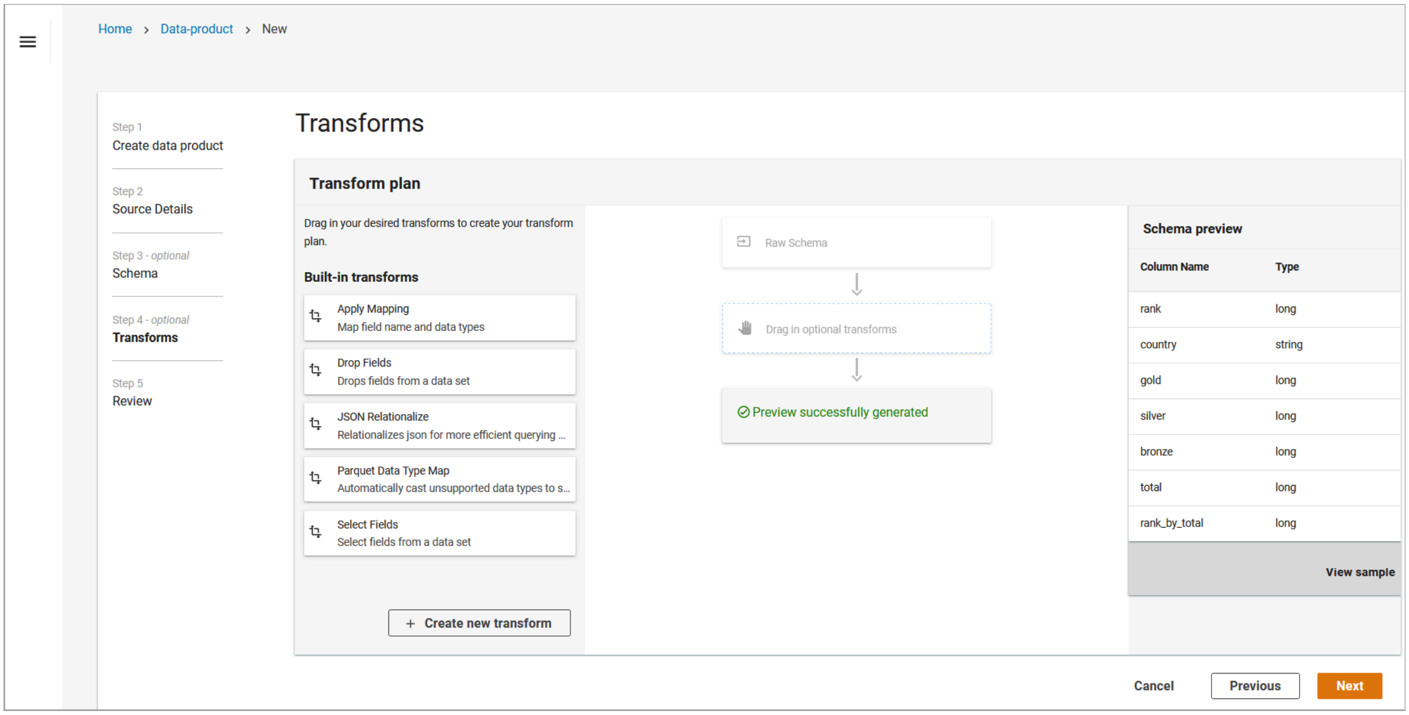
☑ Drop other fields
HINT: Uncheck to preserve unmapped fields.

Field Mappings
Add new item to map field name and data type

HINT: Listed field names might not match current schema at placement of tranform. Use freeform text to input any field name value.

Apply Mapping

3. Select the desired source name (column) and choose from changing the Target Name (column name) or Target Type (data type). Note: You can perform simultaneously or independent of one another. For more information on the Apply Mapping transform, refer to the [ApplyMapping Class - AWS Glue](#) documentation.



Drag in your desired transforms to create your transform plan.

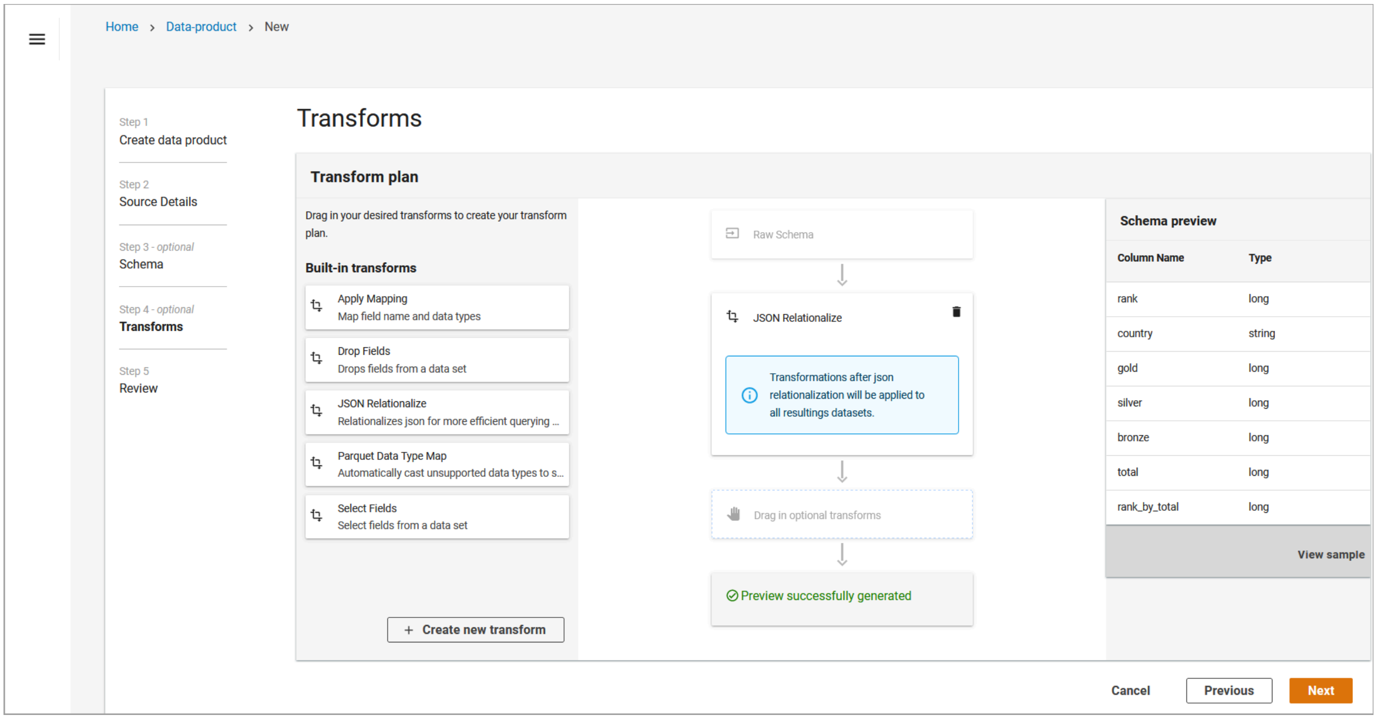
Preview successfully generated

Schema preview

Column Name	Type
rank	long
country	string
gold	long
silver	long
bronze	long
total	long
rank_by_total	long

Apply transforms

- **Drop Fields** allows you to drop a field in the dataset, such as top-level or nested field. You can do this by typing or selecting from the list of fields to drop. Select **Submit** to submit your changes. For more information on the Drop Fields transform, refer to the [SelectFields Class - AWS Glue](#) documentation.
- **JSON Relationalize** allows you convert nested JSON into columns for more efficient queries. To do this, drag the JSON Relationalize transform into the Transform Planner. For more information on the JSON Relationalize transform, refer to [SelectFields Class - AWS Glue](#) documentation.



Drag in your desired transforms to create your transform plan.

Apply Mapping — Map field name and data types

Drop Fields — Drops fields from a data set

JSON Relationalize — Relationalizes json for more efficient querying ...

Parquet Data Type Map — Automatically cast unsupported data types to s...

Select Fields — Select fields from a data set

Transformations after json relationalization will be applied to all resultings datasets.

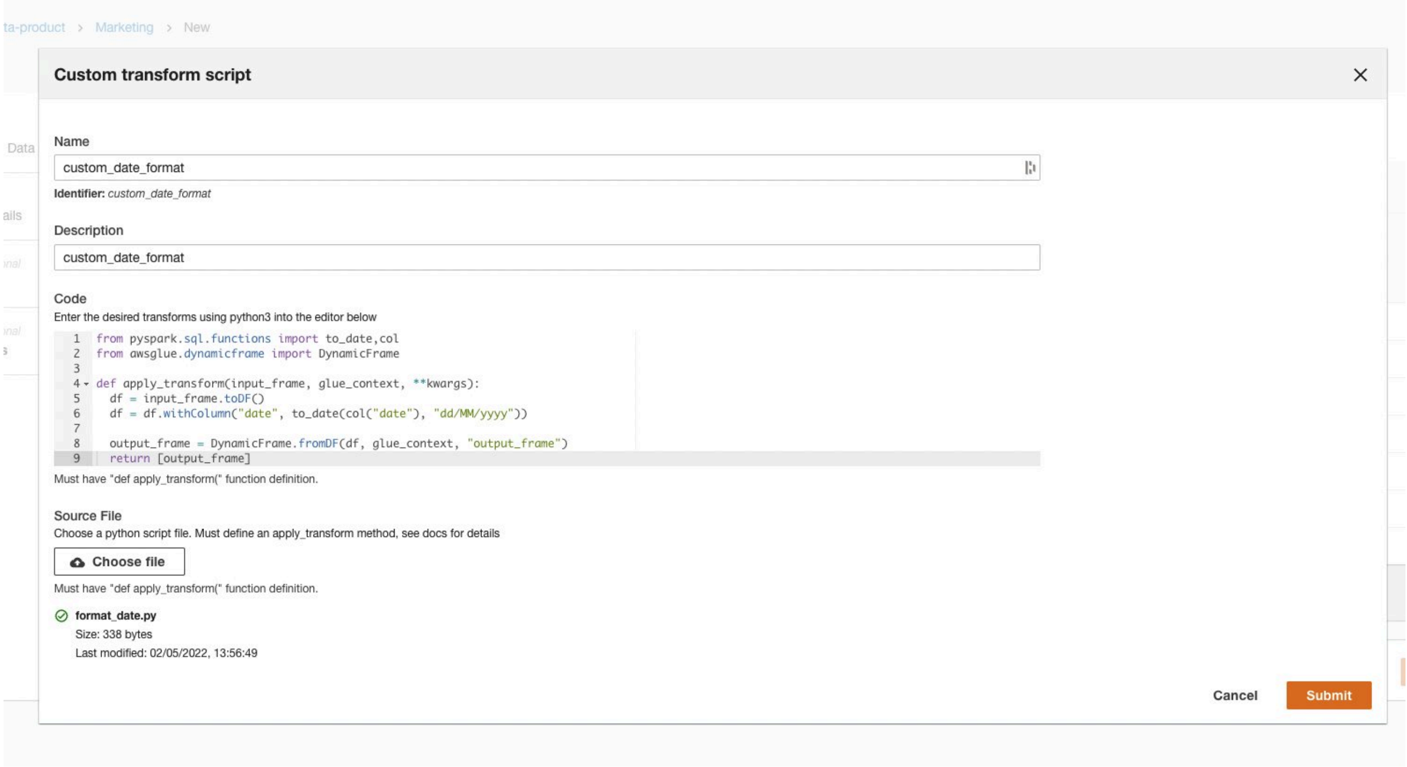
Preview successfully generated

Column Name	Type
rank	long
country	string
gold	long
silver	long
bronze	long
total	long
rank_by_total	long

JSON Relationalize

- **Select Fields** allows you to select fields in the dataset. To do this, type or select from the list of fields. For more information on the JSON Relationalize transform, refer to the SelectFields Class - AWS Glue documentation.

4. You also have the ability to add your own custom transforms through a Python script. To do so, select **Add Custom Transform** at the bottom left hand side of the screen. You will need to add a Name, Identifier and Description of the transform, along with the script added manually or by file upload. If you are using a file upload, select **Choose File** and **Submit**



Custom transform script

Name

custom_date_format

Identifier: *custom_date_format*

Description

custom_date_format

Code

Enter the desired transforms using python3 into the editor below

```python
from pyspark.sql.functions import to_date,col
from awsglue.dynamicframe import DynamicFrame

def apply_transform(input_frame, glue_context, **kwargs):
  df = input_frame.toDF()
  df = df.withColumn("date", to_date(col("date"), "dd/MM/yyyy"))

  output_frame = DynamicFrame.fromDF(df, glue_context, "output_frame")
  return [output_frame]
```

Must have "def apply_transform(" function definition.

Source File

Choose a python script file. Must define an apply_transform method, see docs for details

Must have "def apply_transform(" function definition.

format_date.py
Size: 338 bytes
Last modified: 02/05/2022, 13:56:49

Custom transform script

5. Once you have applied all of the required transforms, select **Next**.
6. Review the transformed schema and select **Submit.**

Viewing the dataset

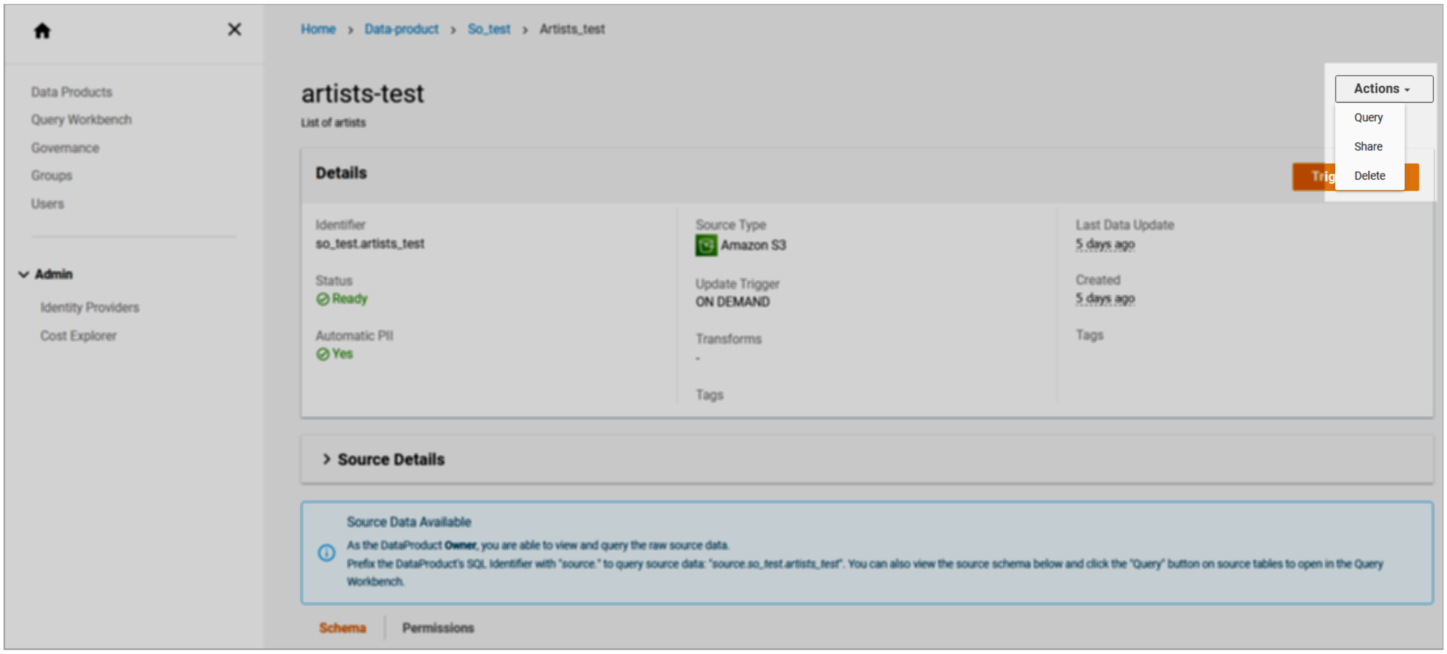
After you have added a data product, you can access it through the **Data Product** option on the left hand navigation pane. This will display all the data products available to you in the **All Data Products** table.

Data products are listed by the domain, name and description that was specified in the Create New Data Product section, along with the metadata collected when it was created and updated. This screen also displays the status of the data product - **Ready, Importing**, or **Import Failed**.

> **Note**
>
> You can only query **Ready** data products.

To access individual data products, select the **Name**.

You can interact with the data products in many ways. You can choose one of the 3 options, Share, Delete, and Query from the **Actions** drop down.



artists-test

List of artists

Details

Identifier: so_test.artists_test

Status: Ready

Automatic PII: Yes

Source Type: Amazon S3

Update Trigger: ON DEMAND

Transforms: -

Tags

Last Data Update: 5 days ago

Created: 5 days ago

Tags

Source Data Available

As the DataProduct **Owner**, you are able to view and query the raw source data.
Prefix the DataProduct's SQL Identifier with "source." to query source data: "source.so_test.artists_test". You can also view the source schema below and click the "Query" button on source tables to open in the Query Workbench.

Query actions

- **Query** allows you to analyze the data product through the Query Workbench. You can query a data product as soon as you complete the **Create a New Data Product** process, while the data product is being imported. To learn more about how to use the Query Workbench, refer to the Query Workbench section.
- **Share** allows you to copy the URL to be sent to other Automated Data Analytics on AWS users.

> **Note**
>
> The user must have privileged access to the data product in order to view its contents.

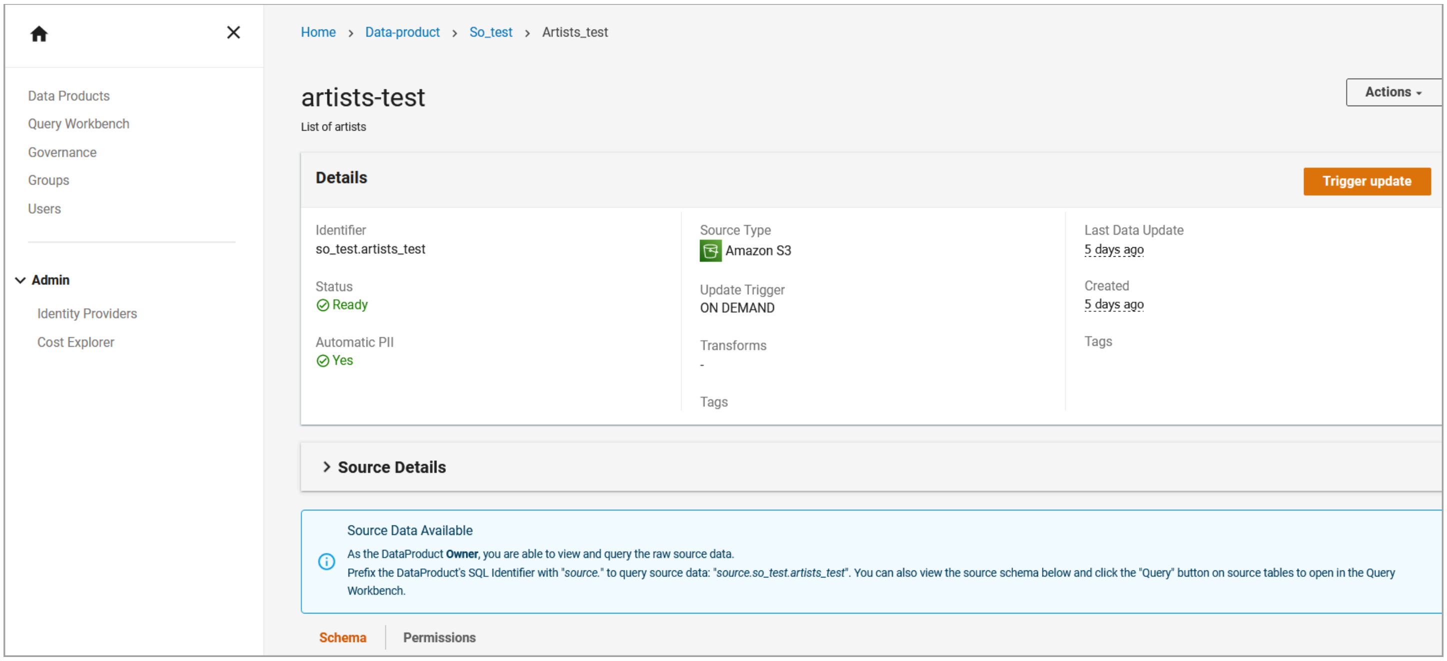
- **Delete** will permanently delete the Data Product.

- Only owners, admins, and users with FULL access rights can delete a data product.
- If there are existing entity dependencies on this data product, the delete will fail and you will need to delete those entities first. For example, if a query has been saved referencing the data product, you must delete the query before you delete the data product.

- To delete a data product, users must acknowledge by typing the word **delete**, and select **Delete**.

Trigger data update

For data products with an **On Demand** data import trigger, to initiate a data refresh from the data source location, select **Trigger Update** from the upper right hand side of the screen. Data products with **Automatic** and **Scheduled** Data Import Trigger windows are refreshed automatically.



artists-test

List of artists

Details

Identifier
so_test.artists_test

Status
✓ Ready

Automatic PII
✓ Yes

Source Type
Amazon S3

Update Trigger
ON DEMAND

Transforms
-

Tags

Last Data Update
5 days ago

Created
5 days ago

Tags

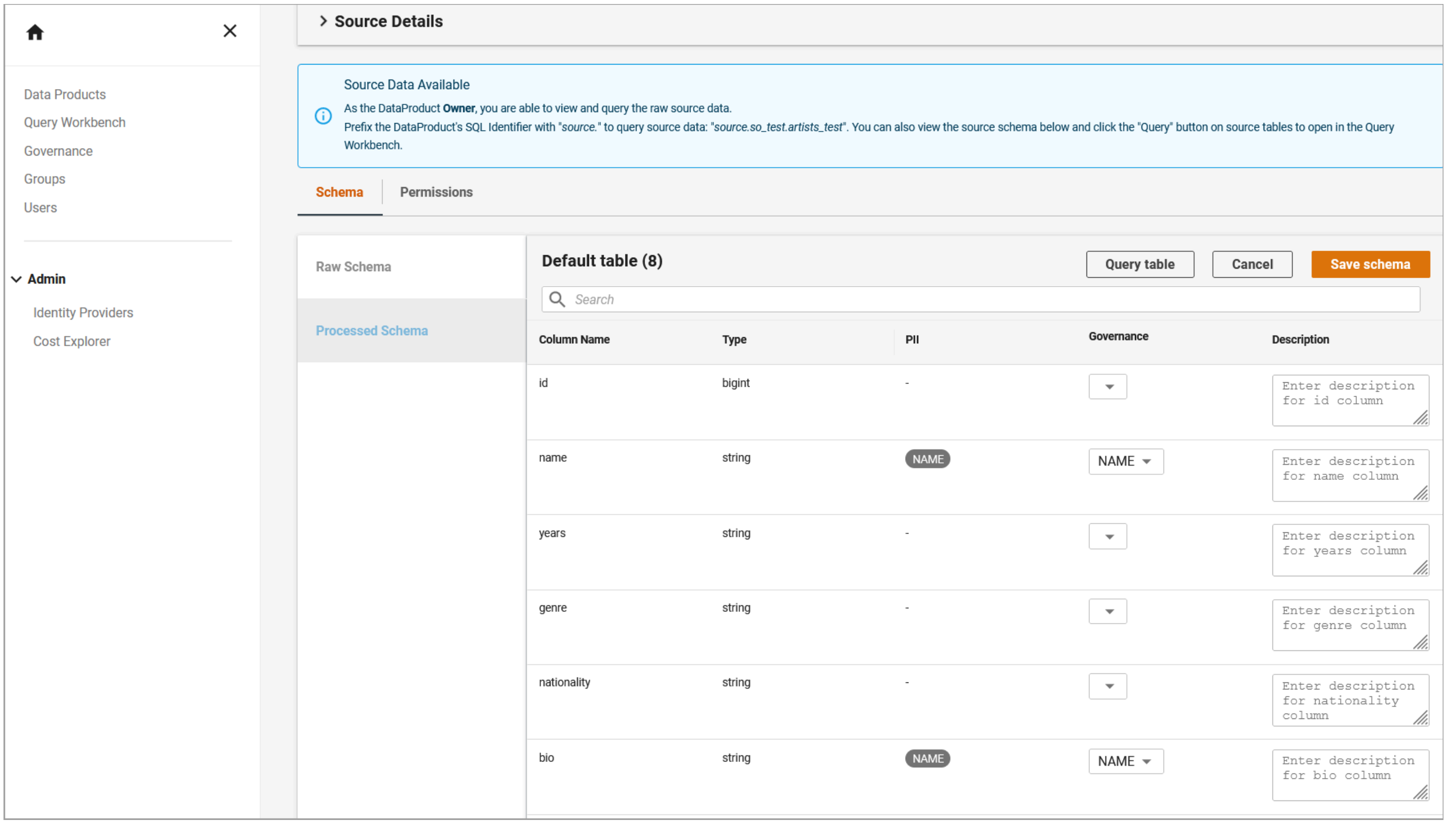
Source Data Available

As the DataProduct **Owner**, you are able to view and query the raw source data.
Prefix the DataProduct's SQL Identifier with "*source.*" to query source data: "*source.so_test.artists_test*". You can also view the source schema below and click the "Query" button on source tables to open in the Query Workbench.

Trigger data update

Schema

To make changes to the Data Product Schema, select the Schema tab in the second table and select **Edit Schema**. To change the governance classifications (such as email, credit card number, or phone number) select a value from the **Governance** field or add a **Description** to the columns. For more information on **Governance** controls, refer to the Governance section.



Source Data Available

As the DataProduct **Owner**, you are able to view and query the raw source data.
Prefix the DataProduct's SQL Identifier with "*source.*" to query source data: "*source.so_test.artists_test*". You can also view the source schema below and click the "Query" button on source tables to open in the Query Workbench.

Default table (8)

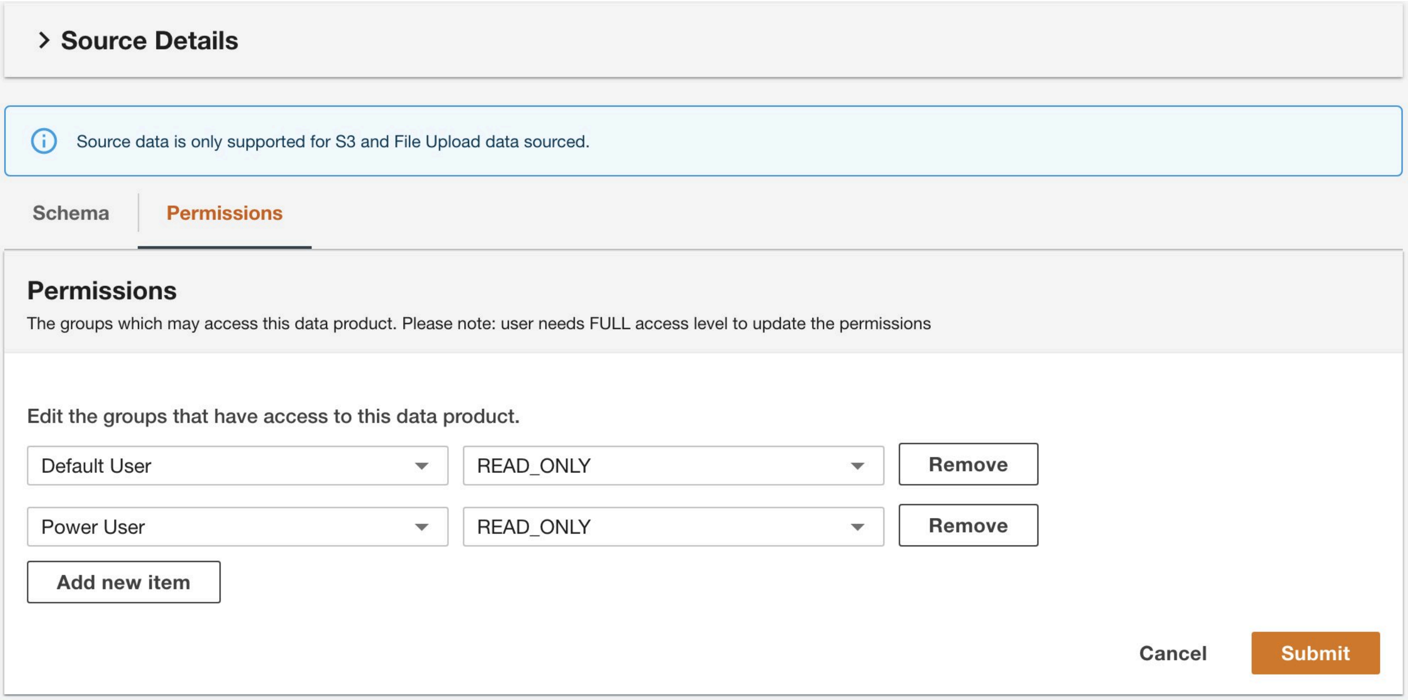
Column Name	Type	PII	Governance	Description
id	bigint	-		Enter description for id column
name	string	NAME	NAME	Enter description for name column
years	string	-		Enter description for years column
genre	string	-		Enter description for genre column
nationality	string	-		Enter description for nationality column
bio	string	NAME	NAME	Enter description for bio column

Update schema

Permissions

To make changes to the group level permissions, select the **Permissions** tab in the second table and select **Edit Permissions**. This allows you to adjust access controls for all the groups you have created in your organization. Permissions include Full, Read Only, and Read Write.

- Only owners, admins, and users with Full rights can modify the data product permissions.
- Source data is only supported for S3 and File Upload data.
- You can search for groups that have been created through the **Search** bar in the Permissions table.



Source data is only supported for S3 and File Upload data sourced.

Permissions

The groups which may access this data product. Please note: user needs FULL access level to update the permissions

Edit the groups that have access to this data product.

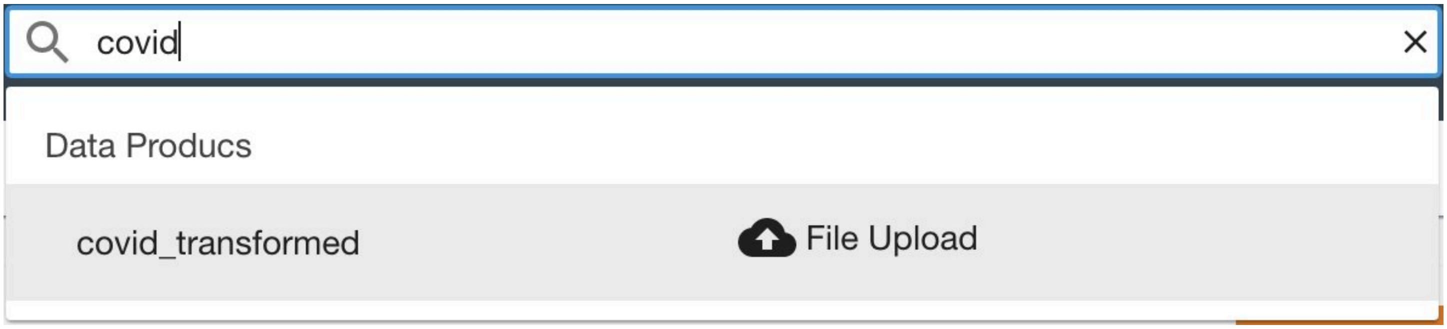
Edit permissions

Source Details

The Source Details drop down in the Details table provides metadata information like Private Key Secret Name, Client ID, Project ID, Client Email, Private Key ID, and Query.

Using the web UI features

Search



ADA Search

From the top of the screen, use the **Search** bar to search for Data Products, Domains, Groups, Saved Queries, and Governance Attributes.

My Profile

To view user information, from the username dropdown, select **Profile**.

Notifications

Users are notified of important information in real time through banners at the top of the screen and the dropdown in upper right hand corner the screen. Notifications include data product creation success, Group join requests and approvals, along with errors.

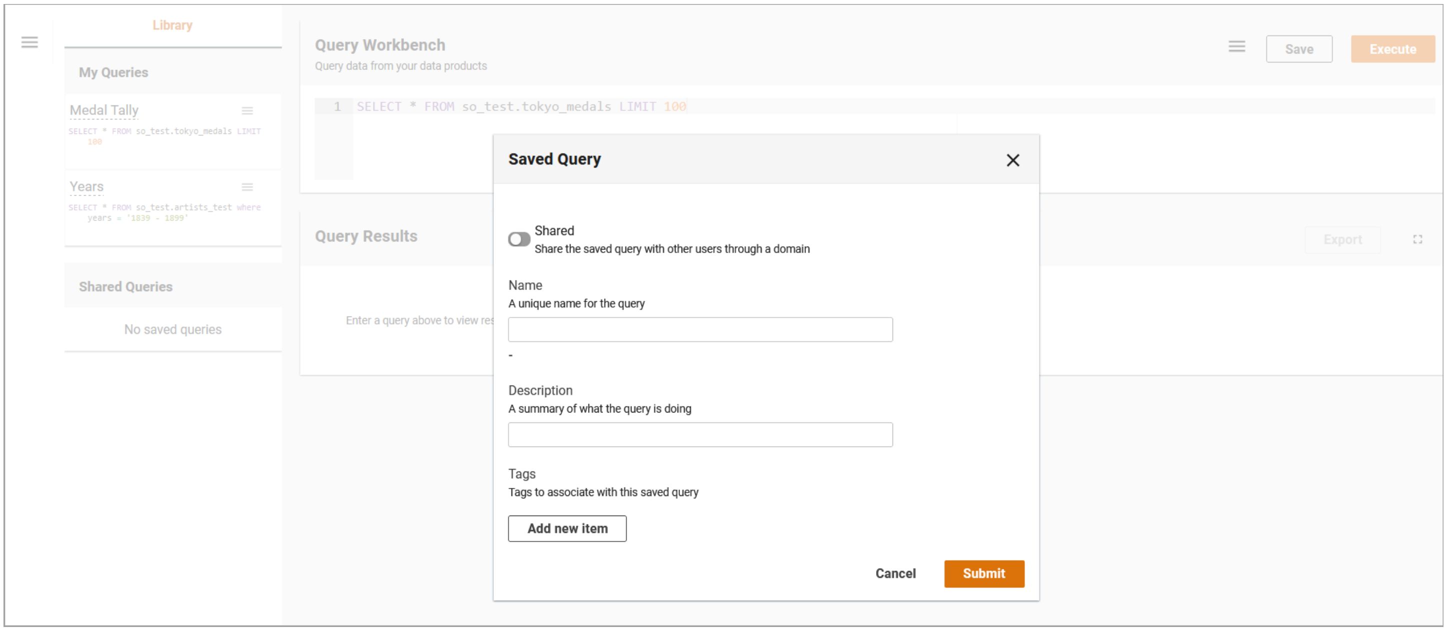
Query Workbench

Using the Query Workbench, you can run SQL-like queries on data products that have been successfully imported. To access the workbench, from the left hand navigation pane, select **Query Workbench** or within a data product, select **Query**.

To write a custom query, use the Query Workbench using SQL Commands. For a full list of the SQL commands, refer to the SELECT - Amazon Athena documentation. To run the SQL Query, select **Execute**.

> **Note**
>
> When you enter the reference data source location, Automated Data Analytics on AWS will automatically fill in the inferred data product.

To save queries that you want to reuse, choose **Save** in the Query Workbench. Add a name, description, and any related tags. If you choose **Shared**, this query will also be made available to others in your domain.



```
SELECT * FROM so_test.tokyo_medals LIMIT 100
```

```
SELECT * FROM so_test.artists_test where
    years = '1839 - 1899'
```

Enter a query above to view res

Saved Query ✕

○ Shared
Share the saved query with other users through a domain

Name
A unique name for the query
-

Description
A summary of what the query is doing

Tags
Tags to associate with this saved query

Add new item

Saved query

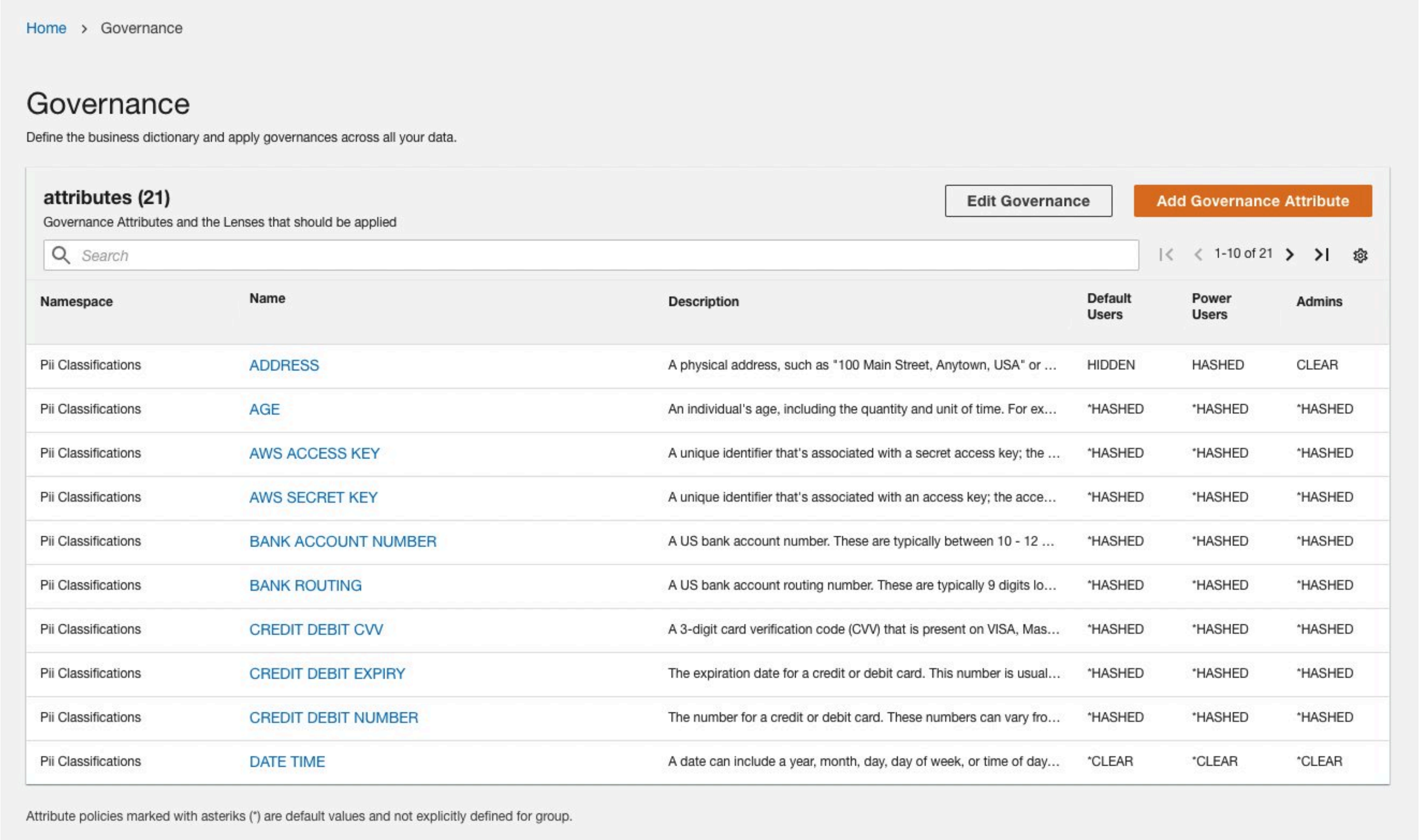
> **Note**
>
> Saved queries that are private will remain private indefinitely.

Query results are generated in the second table on the page. In this table, you can find specific information using the Search bar. You can modify the Page size and Column visibility by selecting the gear icon on the right side of the table or expand to full screen by clicking the square.

To export results to a CSV, select the **Export** button.

Governance

Automated Data Analytics on AWS provides granular row/column level controls through the **Governance** feature. To access this, from the left hand navigation pane, select **Governance.**



Governance

Define the business dictionary and apply governances across all your data.

attributes (21)

Governance Attributes and the Lenses that should be applied

Namespace	Name	Description	Default Users	Power Users	Admins
Pii Classifications	ADDRESS	A physical address, such as "100 Main Street, Anytown, USA" or …	HIDDEN	HASHED	CLEAR
Pii Classifications	AGE	An individual's age, including the quantity and unit of time. For ex…	*HASHED	*HASHED	*HASHED
Pii Classifications	AWS ACCESS KEY	A unique identifier that's associated with a secret access key; the …	*HASHED	*HASHED	*HASHED
Pii Classifications	AWS SECRET KEY	A unique identifier that's associated with an access key; the acce…	*HASHED	*HASHED	*HASHED
Pii Classifications	BANK ACCOUNT NUMBER	A US bank account number. These are typically between 10 - 12 …	*HASHED	*HASHED	*HASHED
Pii Classifications	BANK ROUTING	A US bank account routing number. These are typically 9 digits lo…	*HASHED	*HASHED	*HASHED
Pii Classifications	CREDIT DEBIT CVV	A 3-digit card verification code (CVV) that is present on VISA, Mas…	*HASHED	*HASHED	*HASHED
Pii Classifications	CREDIT DEBIT EXPIRY	The expiration date for a credit or debit card. This number is usual…	*HASHED	*HASHED	*HASHED
Pii Classifications	CREDIT DEBIT NUMBER	The number for a credit or debit card. These numbers can vary fro…	*HASHED	*HASHED	*HASHED
Pii Classifications	DATE TIME	A date can include a year, month, day, day of week, or time of day…	*CLEAR	*CLEAR	*CLEAR

Attribute policies marked with asteriks (*) are default values and not explicitly defined for group.

Governance attributes

For more information on how to set up governance and add governance attributes, refer to the Setting up governance section.

Groups

Groups are Automated Data Analytics on AWS's way of assigning access controls privileges to data products.

To configure them, select the **Groups** tab on the left hand navigation pane.

By default, there are three main groups: Admin, Power User, and Default.

- **Admins** have full access to data products
- **Power Users** have Read/Write access
- **Default** users have read-only access.

> **Note**
>
> Any user can request Group level access by choosing **Join** located in the Status column of the Default Groups table.

To create your own groups:

1. Under Created Groups, select **Create new group**.
2. On the **Create group** page, enter a name, description, select members, and specify Access Policies. The Access Policy options include Default, Administrative Access, Manage Groups, Programmatic Access, Manage Data Products, and Manage Governance.
    - Default - Read only access to the system including domains, data products, query execution, governance rules etc.
    - Administrator Access - Full access to all APIs, including management of domains, data products, groups, identity providers and API access.
    - Manage Groups - Allows creation of groups, assigning users to groups, actioning access request
    - Programmatic Access - Allows creation and management of API keys for programmatic access or integration with external systems.
    - Manage Data Products - Allows creation, editing, and deletion of data products and data product level governance
    - You will also need to add members that will be considered part of the group. You can choose the option to add Bulk users through a list separated by comma, semi-colon, and whitespace (space, tab, return).
3. Once this information is filled in, select **Next.**
4. Review the information and select **Submit.** A new group is created with these details.

Users

The Users page displays a list of users currently federated into the Automated Data Analytics on AWS application.

You can select the user name to view additional user details such as permissions, integration endpoints, API keys, and API access details.

Using the administrator features

Identity Providers

You can set up and allow users to sign in through external federated identity providers.

> **Note**
>
> Only admin users can set up identity providers for other users.

IdP configuration for SAML 2.0

Automated Data Analytics on AWS's identify federation is backed by Cognito, and most of the configuration is mapped to Cognito directly.

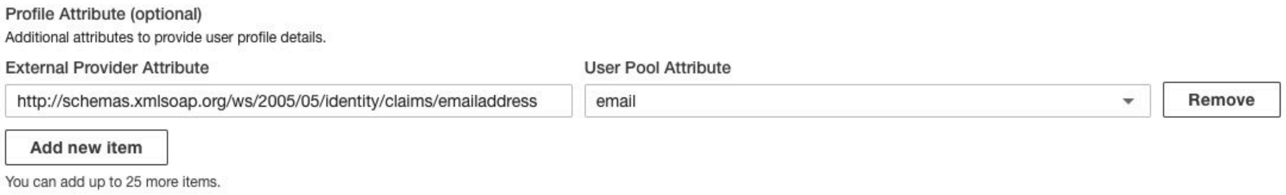
When adding a new identity provider using SAML 2.0, you can find information related to the configuration of the application on the identity provider side (e.g. Okta, Auth0, OneLogin) in the OAuth settings at the bottom of the screen.

- **Domain**: For example, `https://[domainPrefix].auth.ap-southeast-2.amazoncognito.com`
- **Callback / ACS URL / ACS URL Validator:** For example, `https://[domainPrefix].auth.[region].amazoncognito.com/saml2/idpresponse`
- **Audience URI / SP Entity ID:** urn:amazon:cognito:sp:[userPoolId]

> **Note**
>
> - For Audience, replace `userPoolId` with your user pool ID from the OAuth settings.
> - For ACS (Consumer) URL Validator and ACS (Consumer) URL, replace `domainPrefix` and region with the values from the OAuth settings.

1. In the **Preferred Username** field, choose a unique alias that can identify the user. (for example, email, username, or preferredUsername).

2. For **Profile Attribute** under SAML, enter **http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier** as the External Provider Attribute, and use the corresponding attribute in the User Pool Attribute. For example, to map an email address:
   - In the **External Profile Attribute** field, enter **http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress**.
   - In the User Pool Attribute dropdown, select **Email**.


Profile Attribute (optional)
Additional attributes to provide user profile details.

External Provider Attribute: http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress
User Pool Attribute: email

You can add up to 25 more items.

Profile attribute settings.

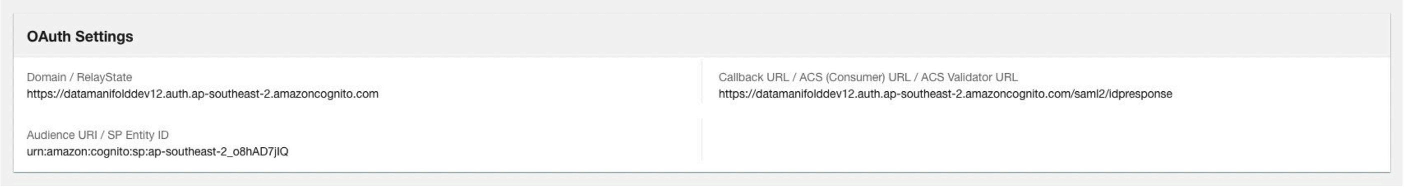
Reference: Set up Auth0 as a SAML identity provider

Create an Auth0 application

1. On the Auth0 website dashboard, choose **Applications**, and then **Create Application**.
2. In the **Create Application** dialog box, enter a name for your application. For example, My App.
3. Under Choose an application type, select **Single Page Web Applications**.
4. Select **Create**.

Configure SAML settings for Automated Data Analytics on AWS

1. From the left navigation hand pane, choose **Applications**.
2. Choose the name of the application you created.
3. On the Addons tab, enable **SAML2 Web App**.
4. In the **Addon: SAML2 Web App** dialog box, on the Settings tab, for Application Callback URL enter `https://[yourDomainPrefix].auth.[region].amazoncognito.com/saml2/idpresponse`. Replace `[yourDomainPrefix]` found in the OAuth settings.


OAuth Settings

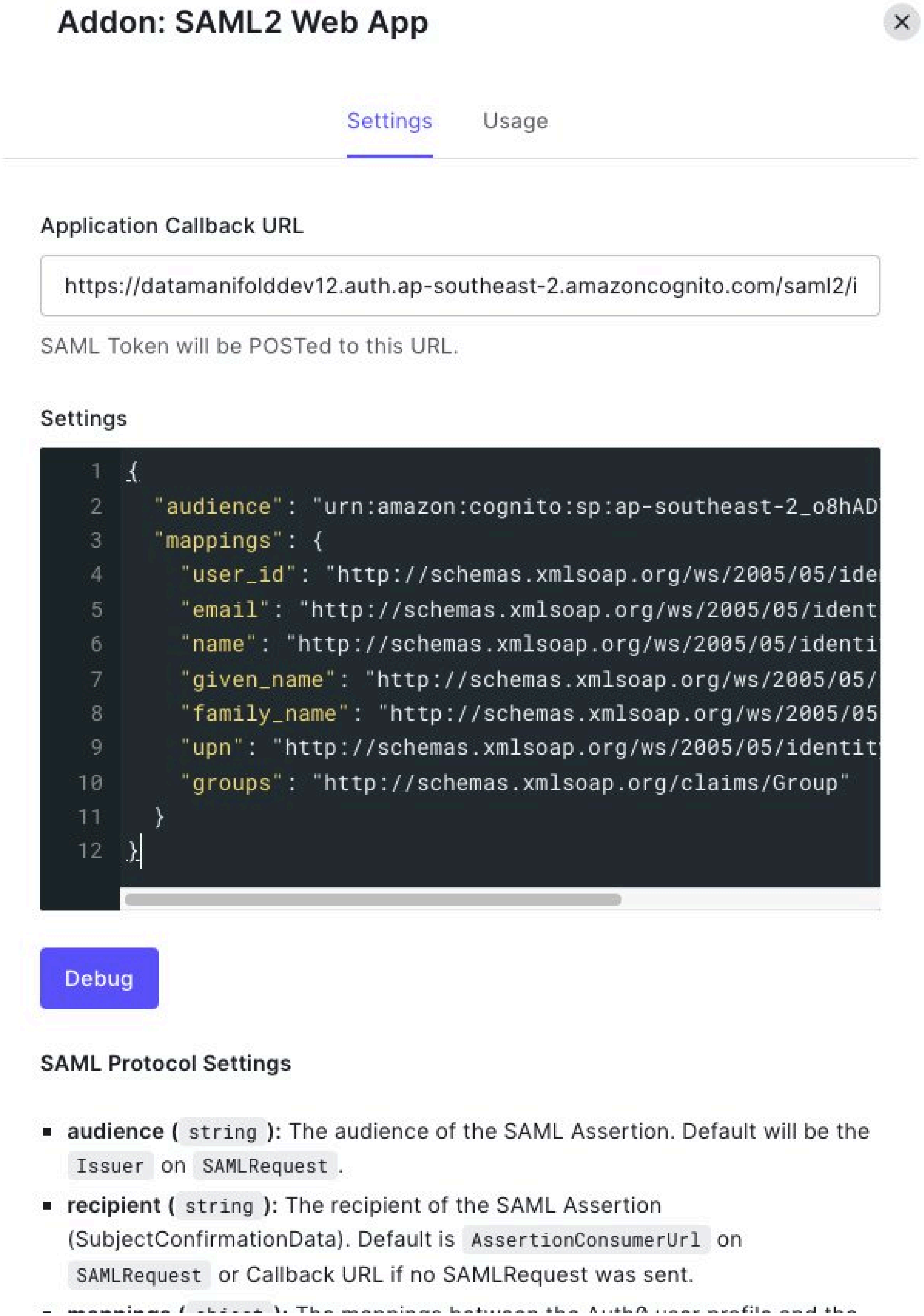
Domain / RelayState
https://datamanifolddev12.auth.ap-southeast-2.amazoncognito.com

Callback URL / ACS (Consumer) URL / ACS Validator URL
https://datamanifolddev12.auth.ap-southeast-2.amazoncognito.com/saml2/idpresponse

Audience URI / SP Entity ID
urn:amazon:cognito:sp:ap-southeast-2_o8hAD7jIQ

OAuth settings

5. Under Settings, complete the following:
    - In the **audience** field, enter **urn:amazon:cognito:sp:yourUserPoolId** from the OAuth settings.
    - In the mappings and email fields, delete the comment delimiters (//), and any other attributes required by your Amazon Cognito user pool. For more information, refer to configuring user pool attributes.
    - In the **nameIdentifierFormat** field, delete the comment delimiters (//). Replace the default value (*urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified*) with *urn:oasis:names:tc:SAML:2.0:nameid-format:persistent*.
6. Select **Enable**, and select **Save**.



Application Callback URL

https://datamanifolddev12.auth.ap-southeast-2.amazoncognito.com/saml2/i

SAML Token will be POSTed to this URL.

Settings

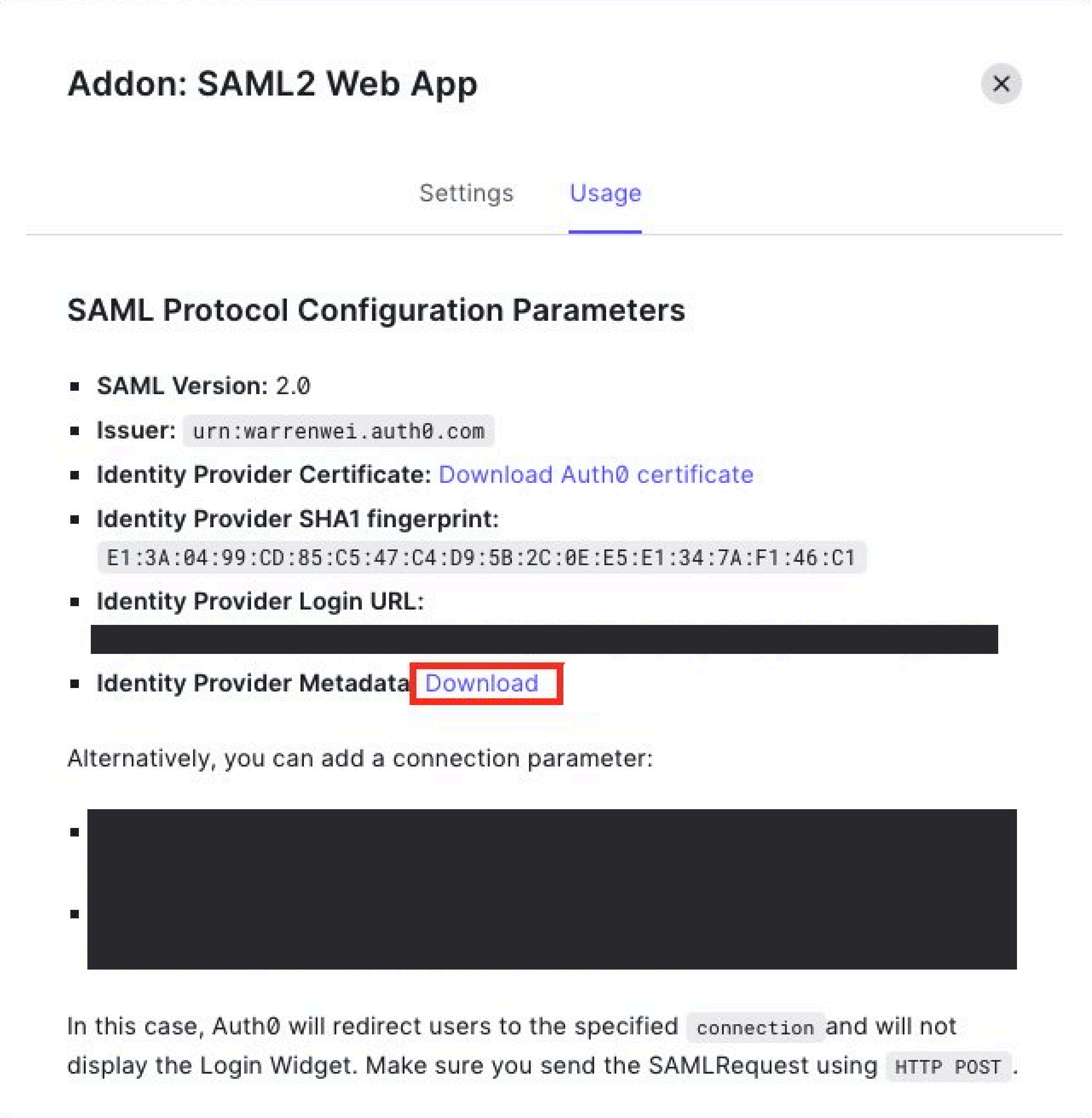
```
{
  "audience": "urn:amazon:cognito:sp:ap-southeast-2_o8hAD
  "mappings": {
    "user_id": "http://schemas.xmlsoap.org/ws/2005/05/ide
    "email": "http://schemas.xmlsoap.org/ws/2005/05/ident
    "name": "http://schemas.xmlsoap.org/ws/2005/05/identi
    "given_name": "http://schemas.xmlsoap.org/ws/2005/05/
    "family_name": "http://schemas.xmlsoap.org/ws/2005/05
    "upn": "http://schemas.xmlsoap.org/ws/2005/05/identit
    "groups": "http://schemas.xmlsoap.org/claims/Group"
  }
}
```

SAML Protocol Settings

- **audience (`string`):** The audience of the SAML Assertion. Default will be the `Issuer` on `SAMLRequest`.
- **recipient (`string`):** The recipient of the SAML Assertion (SubjectConfirmationData). Default is `AssertionConsumerUrl` on `SAMLRequest` or Callback URL if no SAMLRequest was sent.
- **mappings (`object`):** The mappings between the Auth0 user profile and the

SAML settings

7. Select the **Usage** tab, and download the Identity Provider metadata.



Addon: SAML2 Web App

SAML Protocol Configuration Parameters

- **SAML Version:** 2.0
- **Issuer:** `urn:warrenwei.auth0.com`
- **Identity Provider Certificate:** Download Auth0 certificate
- **Identity Provider SHA1 fingerprint:** `E1:3A:04:99:CD:85:C5:47:C4:D9:5B:2C:0E:E5:E1:34:7A:F1:46:C1`
- **Identity Provider Login URL:** [redacted]
- **Identity Provider Metadata** Download

Alternatively, you can add a connection parameter:

- [redacted]

In this case, Auth0 will redirect users to the specified `connection` and will not display the Login Widget. Make sure you send the SAMLRequest using `HTTP POST`.

Download IdP metadata

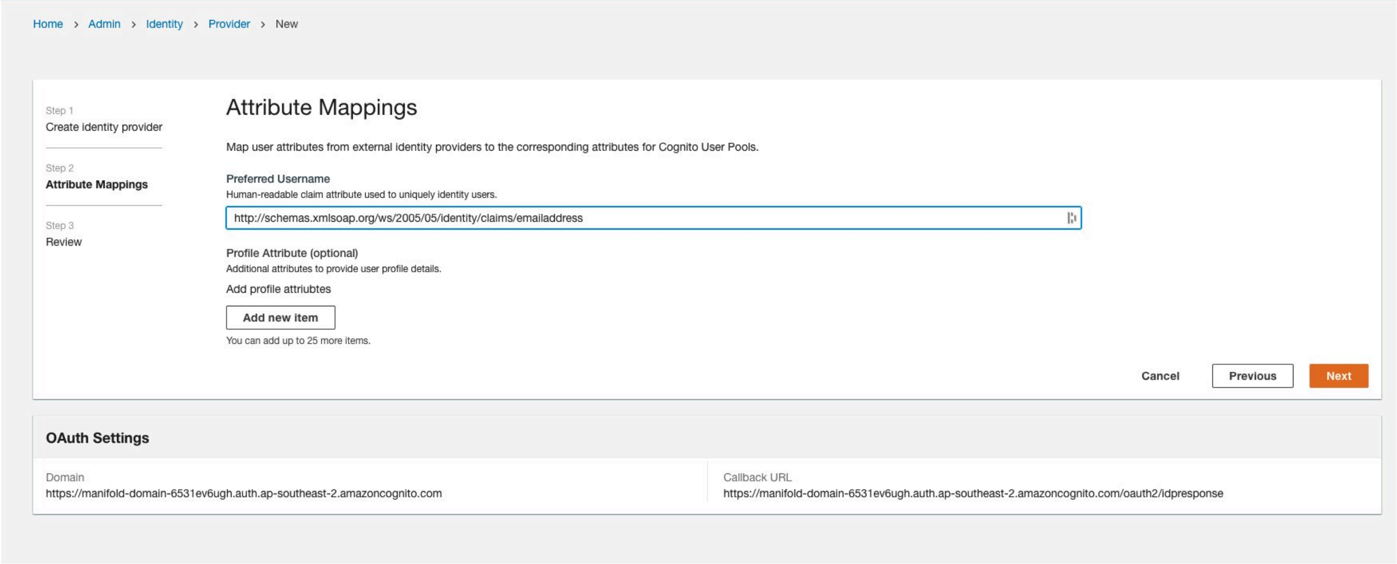
8. On the Automated Data Analytics on AWS web UI, from the left hand, select **Identity Providers**.

9. On the Federated Identity Providers page, select **Add Identity Provider.**

10Enter a name and description, and select SAML.

11. Upload the metadata file downloaded in the previous step, and select **Next**.

12. Enter the preferred name attribute, when adding a SAML attribute, and for the SAML Attribute, enter http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress.



Home > Admin > Identity > Provider > New

Step 1
Create identity provider

Step 2
Attribute Mappings

Step 3
Review

Attribute Mappings

Map user attributes from external identity providers to the corresponding attributes for Cognito User Pools.

Preferred Username
Human-readable claim attribute used to uniquely identity users.
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress

Profile Attribute (optional)
Additional attributes to provide user profile details.
Add profile attriubtes

You can add up to 25 more items.

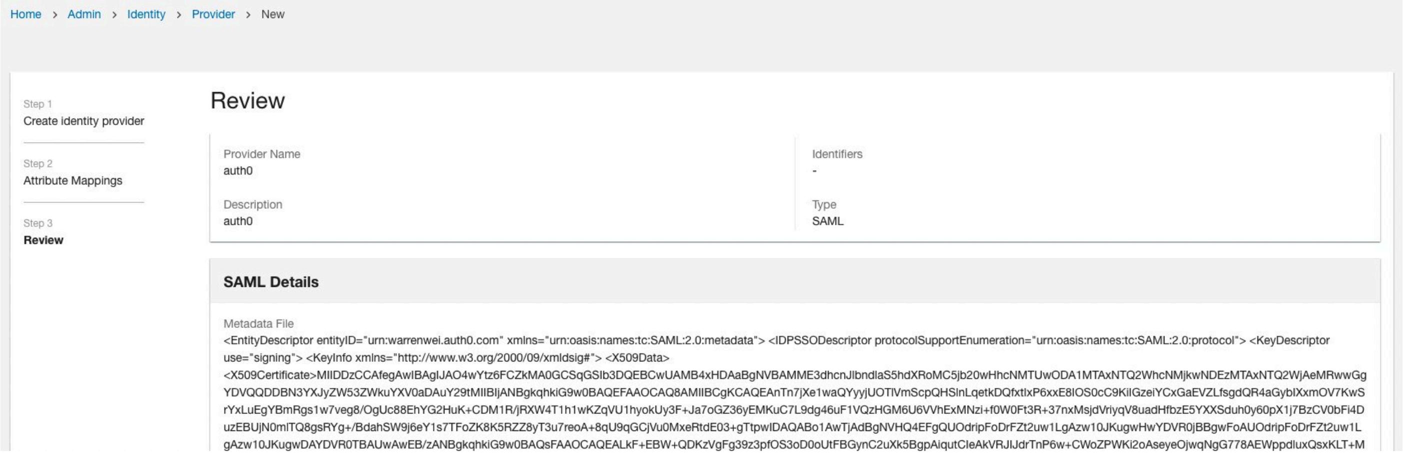
OAuth Settings

Domain
https://manifold-domain-6531ev6ugh.auth.ap-southeast-2.amazoncognito.com

Callback URL
https://manifold-domain-6531ev6ugh.auth.ap-southeast-2.amazoncognito.com/oauth2/idpresponse

Attribute mappings

13. Review your information and select **Submit.** This will set up Auth0 as a SAML Identity Provider for Automated Data Analytics on AWS.



Home > Admin > Identity > Provider > New

Step 1
Create identity provider

Step 2
Attribute Mappings

Step 3
Review

Review

Provider Name
auth0

Identifiers
-

Description
auth0

Type
SAML

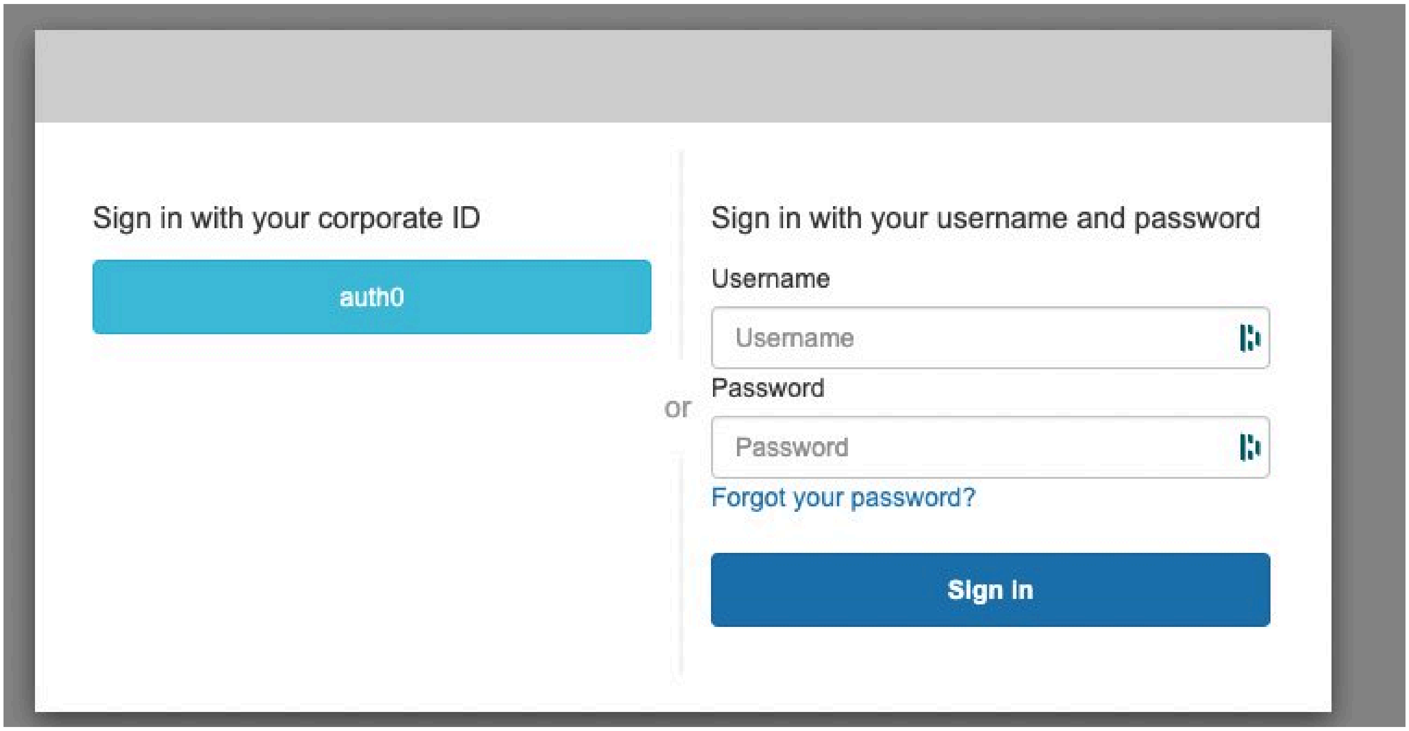
SAML Details

Metadata File

<EntityDescriptor entityID="urn:warrenwei.auth0.com" xmlns="urn:oasis:names:tc:SAML:2.0:metadata"> <IDPSSODescriptor protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol"> <KeyDescriptor
use="signing"> <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#"> <X509Data>
<X509Certificate>MIIDDzCCAfegAwIBAgIJAO4wYtz6FCZkMA0GCSqGSIb3DQEBCwUAMB4xHDAaBgNVBAMME3dhcnJlbndlaS5hdXRoMC5jb20wHhcNMTUwODA1MTAxNTQ2WhcNMjkwNDEzMTAxNTQ2WjAeMRwwGg
YDVQQDDBN3YXJyZW53ZWkuYXV0aDAuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAnTn7jXe1waQYyyjUOTlVmScpQHSlnLqetkDQfxtlxP6xxE8lOS0cC9KilGzeiYCxGaEVZLfsgdQR4aGybIXxmOV7KwS
rYxLuEgYBmRgs1w7veg8/OgUc88EhYG2HuK+CDM1R/jRXW4T1h1wKZqVU1hyokUy3F+Ja7oGZ36yEMKuC7L9dg46uF1VQzHGM6U6VVhExMNzi+f0W0Ft3R+37nxMsjdVriyqV8uadHfbzE5YXXSduh0y60pX1j7BzCV0bFi4D
uzEBUjN0mlTQ8gsRYg+/BdahSW9j6eY1s7TFoZK8K5RZZ8yT3u7reoA+8qU9qGCjVu0MxeRtdE03+gTtpwIDAQABo1AwTjAdBgNVHQ4EFgQUOdripFoDrFZt2uw1LgAzw10JKugwHwYDVR0jBBgwFoAUOdripFoDrFZt2uw1L
gAzw10JKugwDAYDVR0TBAUwAwEB/zANBgkqhkiG9w0BAQsFAAOCAQEALkF+EBW+QDKzVgFg39z3pfOS3oD0oUtFBGynC2uXk5BgpAiqutCIeAkVRJIJdrTnP6w+CWoZPWKi2oAseyeOjwqNgG778AEWppdluxQsxKLT+M

Review attribute mappings

On the sign-in page, you can now log in using auth0.


Sign in with your corporate ID

auth0

or

Sign in with your username and password

Username: Username

Password: Password

Forgot your password?

Sign In

Figure 48: Sign in page

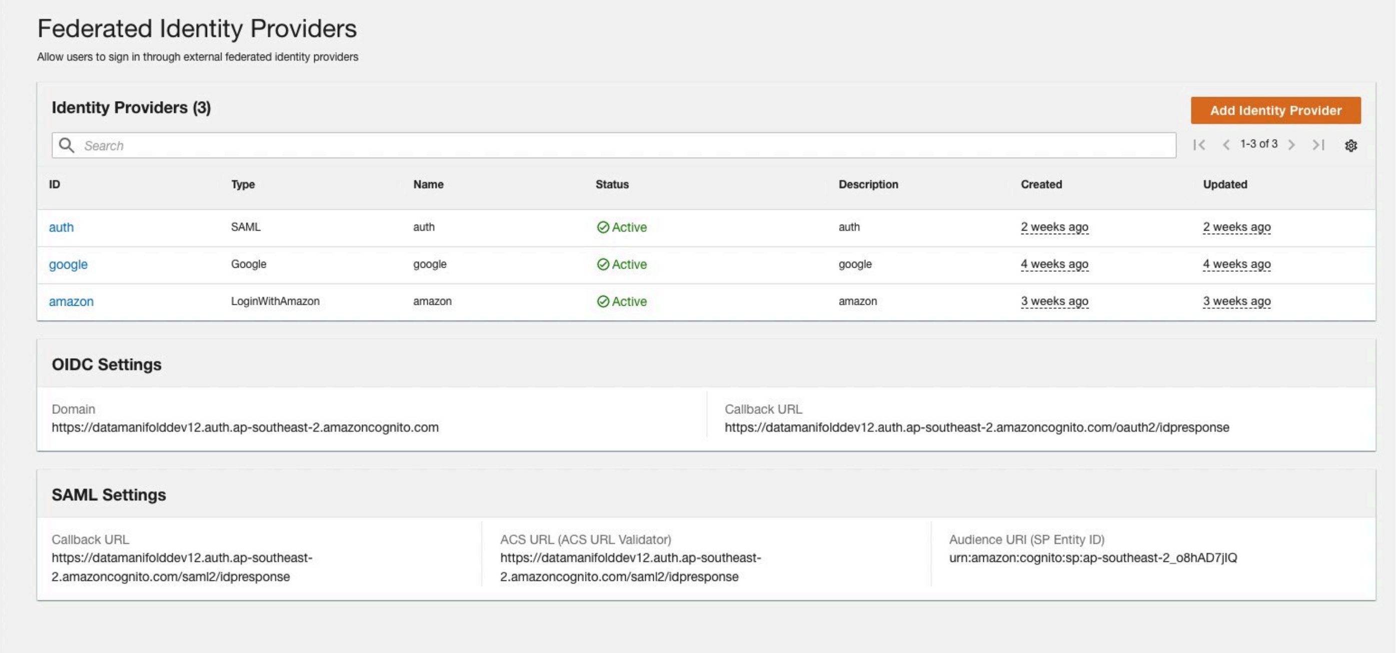
Reference: Set up Auth0 as a SAML identity provider

Create Auth0 application

1. From the Auth0 website, select the Dashboard.
2. In the navigation pane, expand Applications on the left pane, and select **Create Application**.
3. In the dialog box, enter a name for the application. Fox example, *App1*.
4. Under Choose an application type dropdown, select Single Webpage Applications.
5. Select **Create**.

   Note the client ID, client secret, and domain from the application settings tab of the Auth0 application.

6. In the Allowed Callback URLs section, add the Automated Data Analytics on AWS callback domain. You can find the callback URL on the **Automated Data Analytics on AWS Identity providers > OIDC Settings** page.



Federated Identity Providers

Allow users to sign in through external federated identity providers

Identity Providers (3)

ID	Type	Name	Status	Description	Created	Updated
auth	SAML	auth	✓ Active	auth	2 weeks ago	2 weeks ago
google	Google	google	✓ Active	google	4 weeks ago	4 weeks ago
amazon	LoginWithAmazon	amazon	✓ Active	amazon	3 weeks ago	3 weeks ago

OIDC Settings

Domain
https://datamanifolddev12.auth.ap-southeast-2.amazoncognito.com

Callback URL
https://datamanifolddev12.auth.ap-southeast-2.amazoncognito.com/oauth2/idpresponse

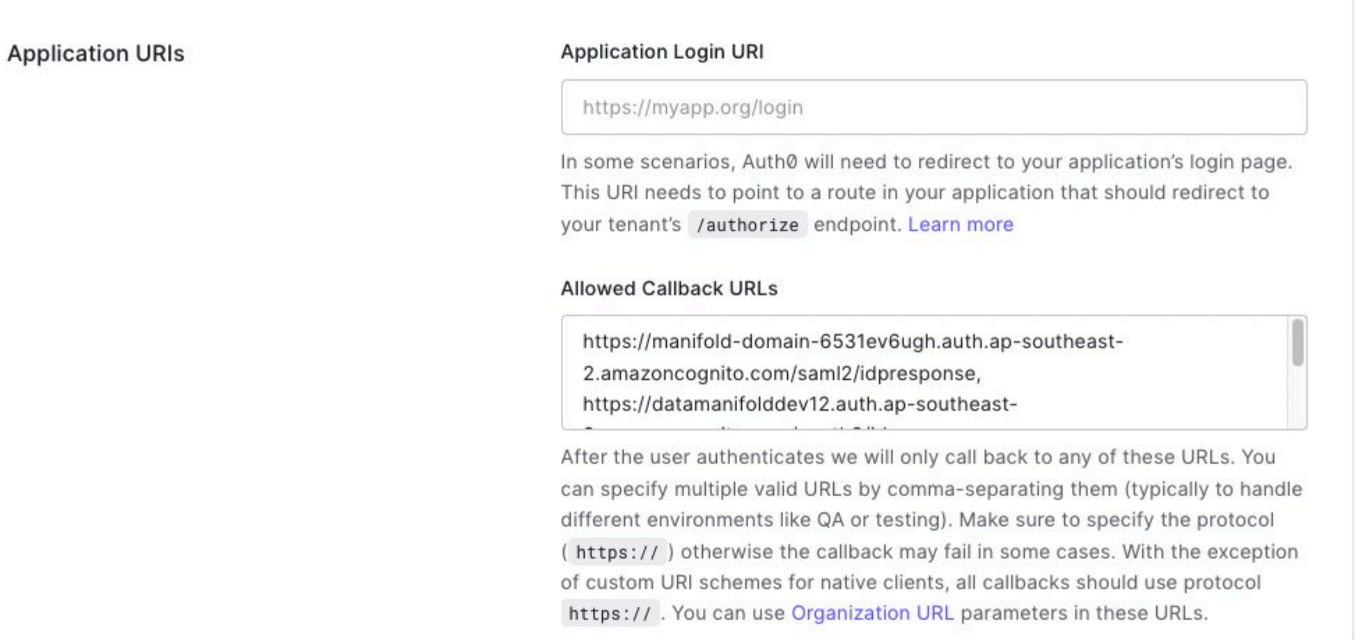
SAML Settings

Callback URL
https://datamanifolddev12.auth.ap-southeast-2.amazoncognito.com/saml2/idpresponse

ACS URL (ACS URL Validator)
https://datamanifolddev12.auth.ap-southeast-2.amazoncognito.com/saml2/idpresponse

Audience URI (SP Entity ID)
urn:amazon:cognito:sp:ap-southeast-2_o8hAD7jIQ

Federated identity providers



Application URIs

Application Login URI

In some scenarios, Auth0 will need to redirect to your application's login page. This URI needs to point to a route in your application that should redirect to your tenant's `/authorize` endpoint. Learn more

Allowed Callback URLs

https://manifold-domain-6531ev6ugh.auth.ap-southeast-2.amazoncognito.com/saml2/idpresponse,
https://datamanifolddev12.auth.ap-southeast-

After the user authenticates we will only call back to any of these URLs. You can specify multiple valid URLs by comma-separating them (typically to handle different environments like QA or testing). Make sure to specify the protocol (`https://`) otherwise the callback may fail in some cases. With the exception of custom URI schemes for native clients, all callbacks should use protocol `https://` . You can use Organization URL parameters in these URLs.

Application URIs

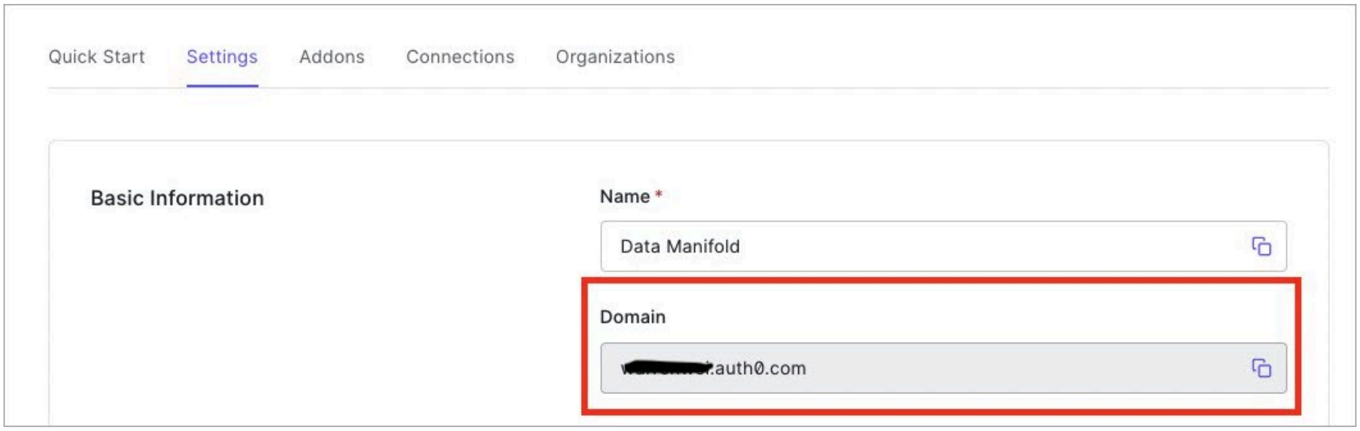
Create Identity Provider in Automated Data Analytics on AWS

1. On the Automated Data Analytics on AWS web UI, from the left hand, select **Identity Providers**.
2. On the Federated Identity Providers page, select **Add Identity Provider**.

3. On the **Create identity provider** page, enter the provider name and description.

4. Choose OIDC.

5. For Issuer, add the domain name from the Auth0 console. For example: https://example.auth0.com.



Basic Information

Name *

Data Manifold

Domain

[redacted].auth0.com

OIDC settings

6. Enter the `Client ID` and `Client secret` from the Auth0 application.

7. Choose GET as Attribute Request Method. You can skip Advanced (Optional) field if all the URLs are following OIDC well-known configuration.

8. Select **Next.**

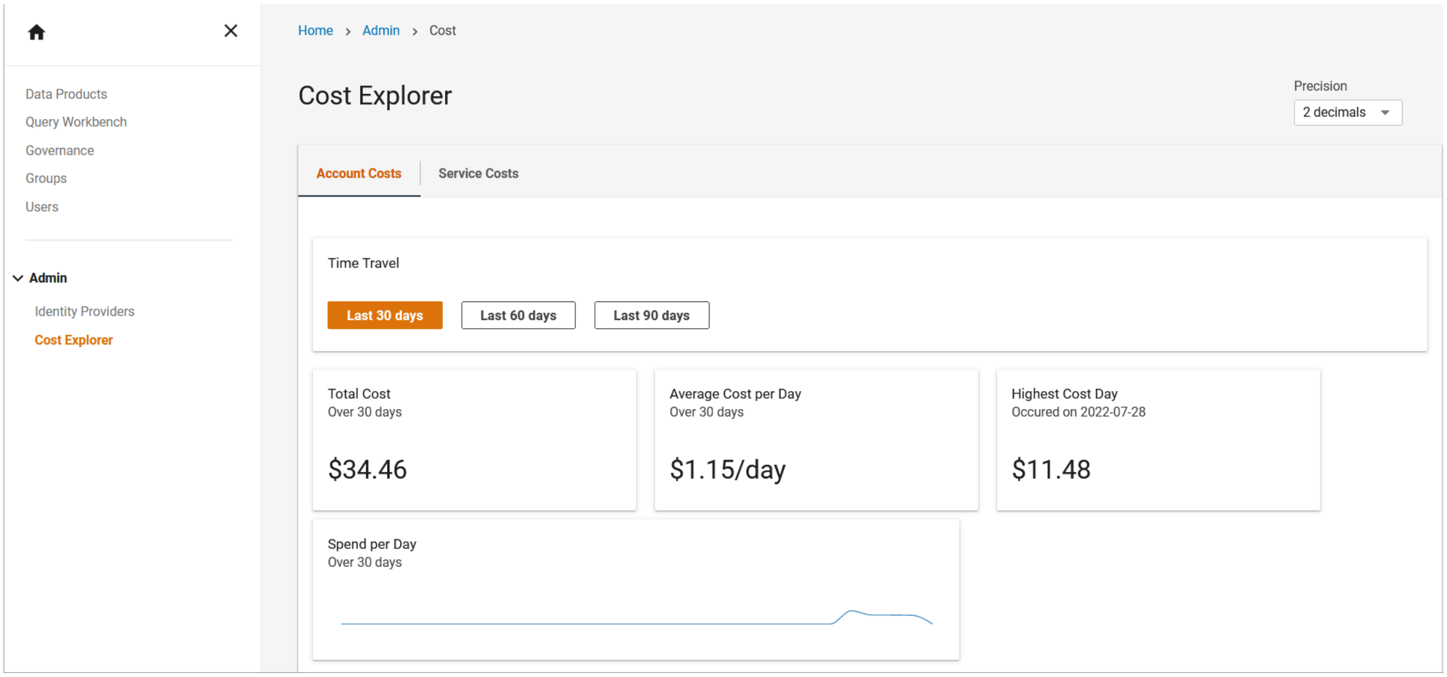
9. For Preferred Username, enter the attribute that maps to a field in the OIDC claim token. (for example, email, username)

10Select **Next** to review, and select **Submit**.

> **Note**
>
> Automated Data Analytics on AWS does not support modifying IdP configuration once saved. You will need to recreate the configuration for changes. Existing users will be not be impacted by this.

Cost Explorer

The solution's web UI provides a **Cost Explorer** feature, available to administrators to view and understand the cost of the solution over time. This feature gets the account and service level cost details from AWS Cost Explorer, and is located under the **Administrator** section of the web UI for administrator group members only.



Cost Explorer

Precision: 2 decimals

Time Travel

Total Cost
Over 30 days
$34.46

Average Cost per Day
Over 30 days
$1.15/day

Highest Cost Day
Occured on 2022-07-28
$11.48

Spend per Day
Over 30 days

Cost explorer option

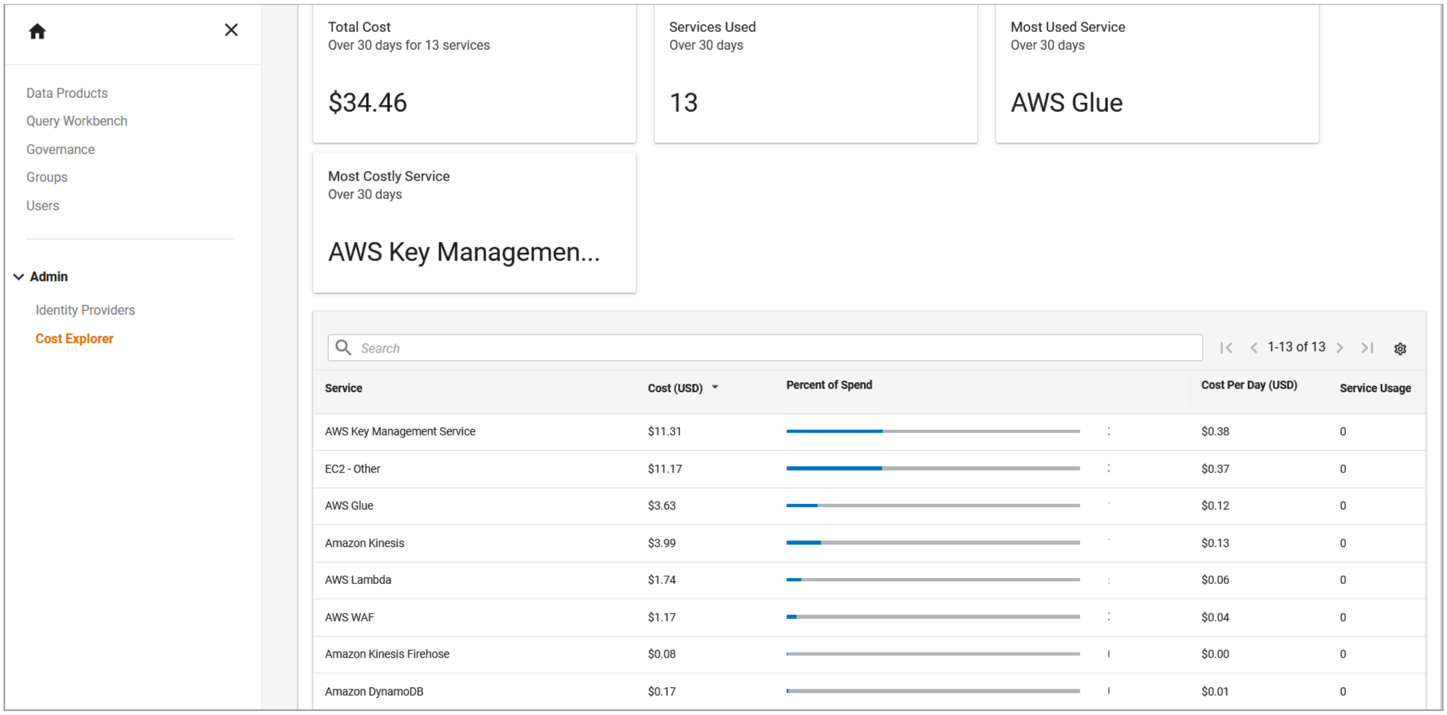
Activating the Cost Explorer feature

To activate this feature in the Web UI, you must enable Cost Explorer from the AWS Management Console and activate the *Application* tag.

1. Sign in to the AWS Management Console.
2. Open the AWS Cost Explorer. Navigating to this page automatically activates the AWS Cost Explorer within 24 hours.
3. After you activate AWS Cost Explorer, open Cost allocation tags.
4. Select the **Application** tag, and choose **Activate.** This activates the Cost Explore feature in the web UI.

Using the Cost Explorer, users can view Account and Service level costs. Account costs are displayed by the last 30, 60, or 90 days. You can refine these figures to different decimals using the **Precision** dropdown in the upper right-hand side of the table.

Users can also view AWS Service costs by toggling to **Service Costs** on the upper left-hand side of the table.



Total Cost
Over 30 days for 13 services
$34.46

Services Used
Over 30 days
13

Most Used Service
Over 30 days
AWS Glue

Most Costly Service
Over 30 days
AWS Key Managemen...

Service	Cost (USD)	Percent of Spend	Cost Per Day (USD)	Service Usage
AWS Key Management Service	$11.31		$0.38	0
EC2 - Other	$11.17		$0.37	0
AWS Glue	$3.63		$0.12	0
Amazon Kinesis	$3.99		$0.13	0
AWS Lambda	$1.74		$0.06	0
AWS WAF	$1.17		$0.04	0
Amazon Kinesis Firehose	$0.08		$0.00	0
Amazon DynamoDB	$0.17		$0.01	0

Service costs

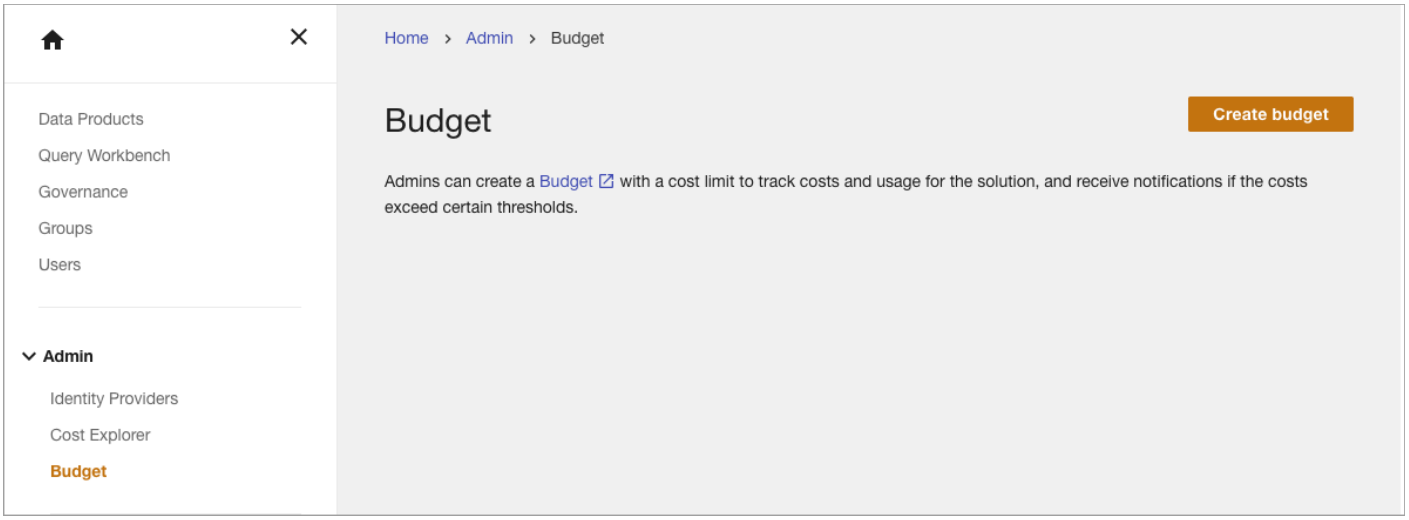
Budget

Use the **Budget** page to create a budget with a cost limit to track costs and usage for the Automated Data Analytics on AWS solution, and receive notifications if the costs exceed certain thresholds.

> **Note**
>
> Only *admin* users can view, create, or modify budgets.

Creating a budget

1. To access the budget feature, from the Automated Data Analytics on AWS home screen, on the left menu, under **Admin**, select **Budget**.

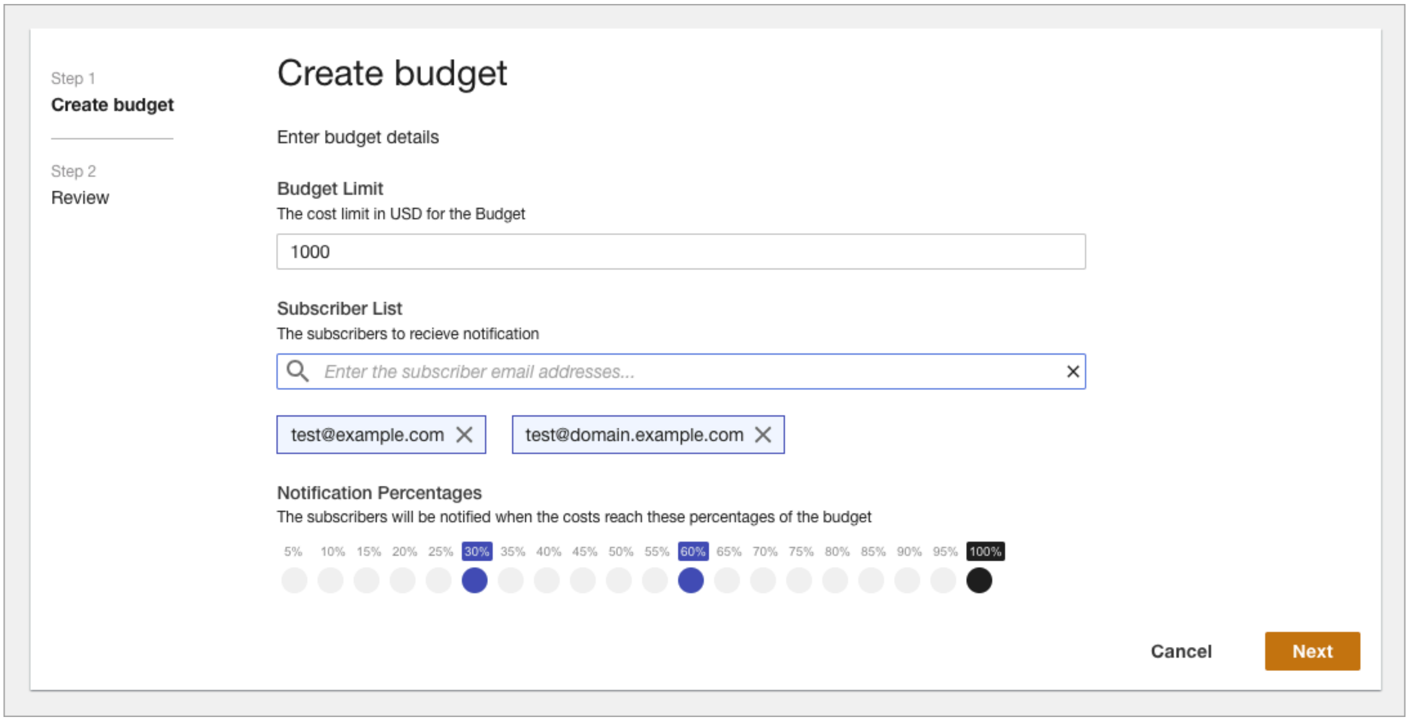


Budget

Admins can create a Budget with a cost limit to track costs and usage for the solution, and receive notifications if the costs exceed certain thresholds.

Budget

2. To add a budget for the solution, select **Create budget**. The **Create budget** page displays.



Create budget

Enter budget details

Budget Limit

The cost limit in USD for the Budget

1000

Subscriber List

The subscribers to recieve notification

Enter the subscriber email addresses...

test@example.com

test@domain.example.com

Notification Percentages

The subscribers will be notified when the costs reach these percentages of the budget

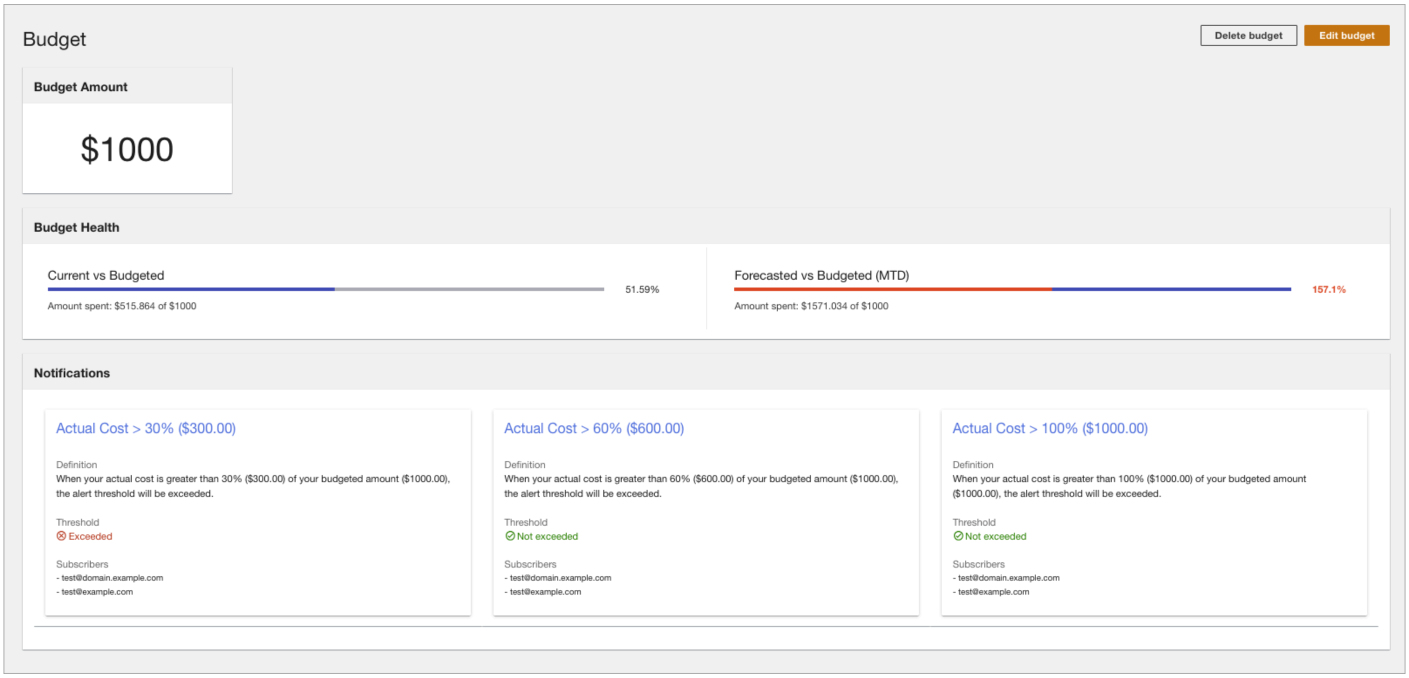
Create budget

3. On the **Create budget** page:

- Enter a cost limit value (in USD) for the budget.
- Enter a list of subscriber email addresses who will get notified when the thresholds are met.

- Select percentage threshold values for notifications and the subscribers will be notified when the current spend is greater than these percentages of the budget limit. You can select multiple threshold percentages when you want to get notifications.

4. Select **Next** to save your changes and create a budget. Once a budget is created, the **Budget Details** page shows the budget amount, budget health and any associated notifications.



Budget

Budget Amount

$1000

Budget Health

Current vs Budgeted 51.59%

Amount spent: $515.864 of $1000

Forecasted vs Budgeted (MTD) 157.1%

Amount spent: $1571.034 of $1000

Notifications

Actual Cost > 30% ($300.00)

Definition
When your actual cost is greater than 30% ($300.00) of your budgeted amount ($1000.00), the alert threshold will be exceeded.

Threshold
⊗ Exceeded

Subscribers
- test@domain.example.com
- test@example.com

Actual Cost > 60% ($600.00)

Definition
When your actual cost is greater than 60% ($600.00) of your budgeted amount ($1000.00), the alert threshold will be exceeded.

Threshold
✓ Not exceeded

Subscribers
- test@domain.example.com
- test@example.com

Actual Cost > 100% ($1000.00)

Definition
When your actual cost is greater than 100% ($1000.00) of your budgeted amount ($1000.00), the alert threshold will be exceeded.

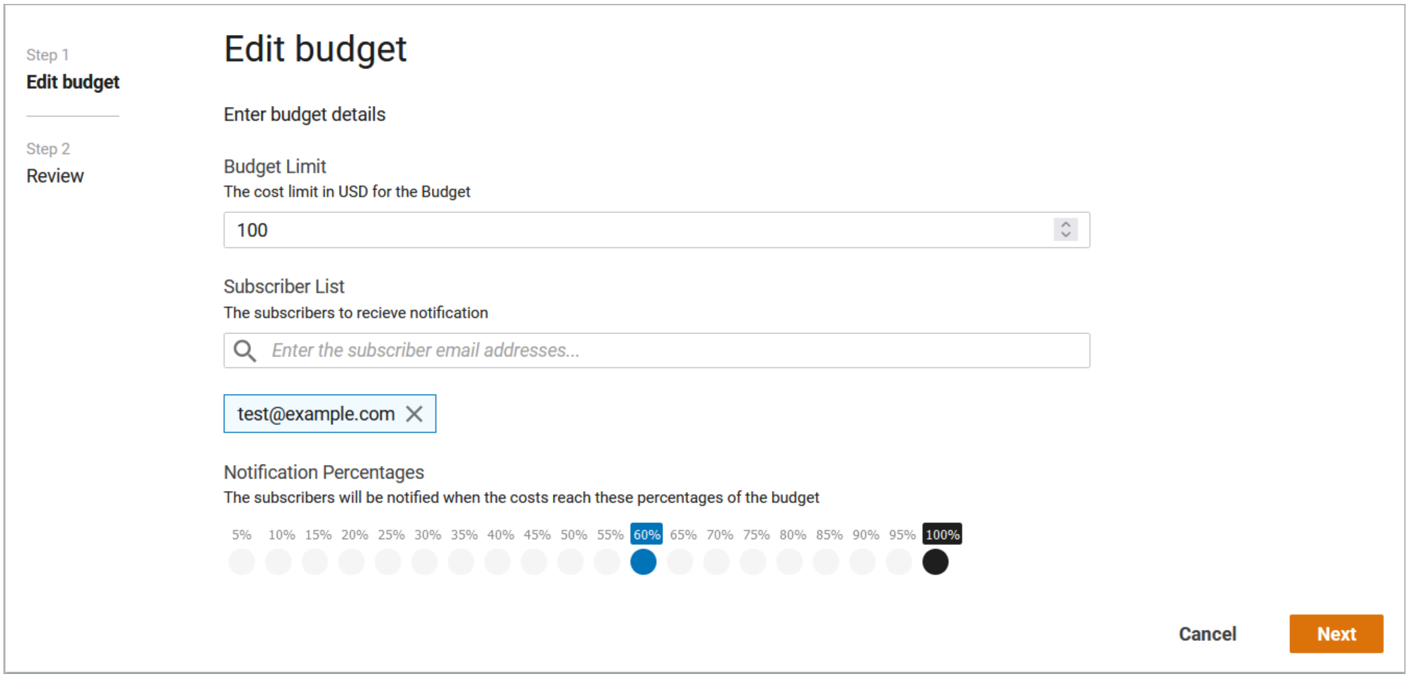
Threshold
✓ Not exceeded

Subscribers
- test@domain.example.com
- test@example.com

Budget details

Editing a budget

1. To edit an existing budget, from the Automated Data Analytics **Budget** page, select the budget and choose **Edit budget**. The **Edit budget** page displays.



Step 1

Edit budget

Step 2

Review

Edit budget

Enter budget details

Budget Limit

The cost limit in USD for the Budget

100

Subscriber List

The subscribers to recieve notification

Enter the subscriber email addresses...

test@example.com

Notification Percentages

The subscribers will be notified when the costs reach these percentages of the budget

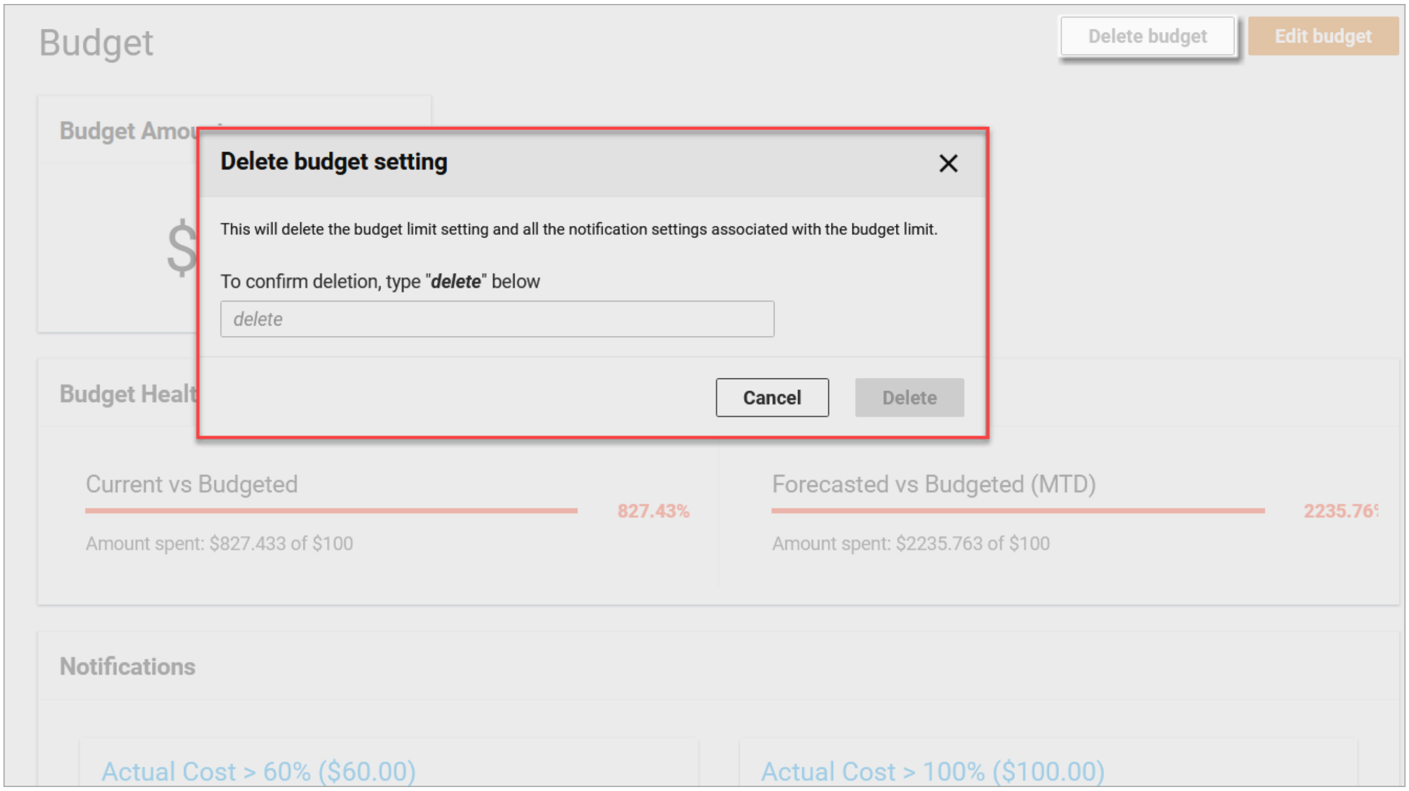
5%	10%	15%	20%	25%	30%	35%	40%	45%	50%	55%	60%	65%	70%	75%	80%	85%	90%	95%	100%
○	○	○	○	○	○	○	○	○	○	○	●	○	○	○	○	○	○	○	●

Edit budget

2. Update the budget fields as required, and select **Next**. The budget details are updated and displayed on the Budget details page.

Deleting a budget

1. To delete an existing budget, from the Automated Data Analytics **Budget** page, select the budget and choose **Delete budget**. The **Delete budget** dialog box displays.



Delete budget setting

This will delete the budget limit setting and all the notification settings associated with the budget limit.

To confirm deletion, type "***delete***" below

delete

Cancel Delete

Current vs Budgeted 827.43%

Amount spent: $827.433 of $100

Forecasted vs Budgeted (MTD) 2235.76

Amount spent: $2235.763 of $100

Actual Cost > 60% ($60.00)

Actual Cost > 100% ($100.00)

Delete budget

2. To confirm deletion, type **delete** in the text box and select **Delete**. The budget is now deleted and all associated notifications are removed.

Visualization

Use the **Visualization** page to deploy Apache Superset as a data analytics and visualization platform for Automated Data Analytics on AWS (ADA) solution.

> **Note**
>
> Only *admin* users can can deploy Apache Superset for visualization purposes.

Deploying Apache Superset

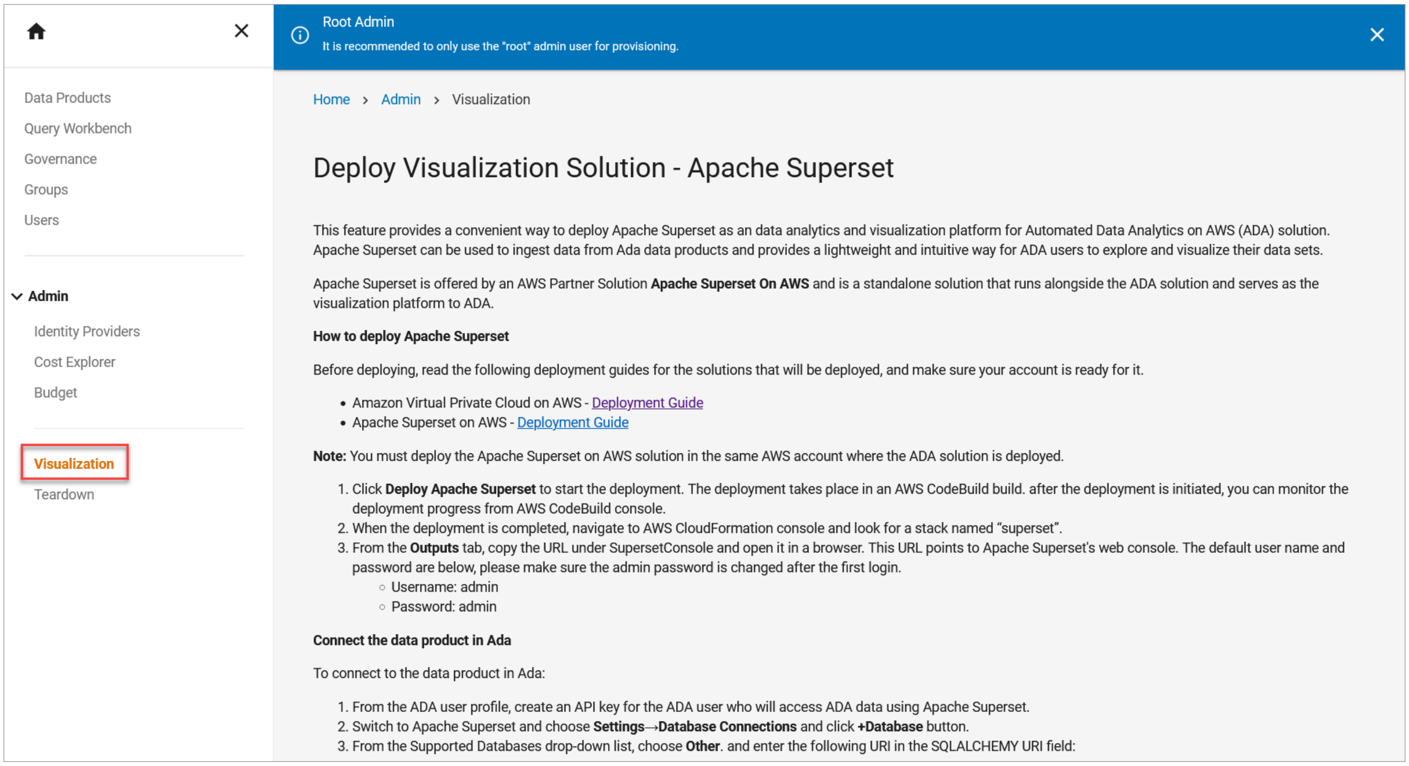
Before deploying, read the following deployment guides for the solutions that will be installed, and make sure your AWS account is set up for this deployment.

- Amazon Virtual Private Cloud on AWS (Deployment guide)

- [Apache Superset on AWS (Deployment guide)]

> **Note**
>
> You must deploy the Apache Superset on AWS solution in the same AWS account where the ADA solution is deployed.

1. From the Automated Data Analytics on AWS home screen, on the left menu, under **Admin**, select **Visualization**.



Root Admin
It is recommended to only use the "root" admin user for provisioning.

Deploy Visualization Solution - Apache Superset

This feature provides a convenient way to deploy Apache Superset as an data analytics and visualization platform for Automated Data Analytics on AWS (ADA) solution. Apache Superset can be used to ingest data from Ada data products and provides a lightweight and intuitive way for ADA users to explore and visualize their data sets.

Apache Superset is offered by an AWS Partner Solution **Apache Superset On AWS** and is a standalone solution that runs alongside the ADA solution and serves as the visualization platform to ADA.

How to deploy Apache Superset

Before deploying, read the following deployment guides for the solutions that will be deployed, and make sure your account is ready for it.

- Amazon Virtual Private Cloud on AWS - Deployment Guide
- Apache Superset on AWS - Deployment Guide

Note: You must deploy the Apache Superset on AWS solution in the same AWS account where the ADA solution is deployed.

1. Click **Deploy Apache Superset** to start the deployment. The deployment takes place in an AWS CodeBuild build. after the deployment is initiated, you can monitor the deployment progress from AWS CodeBuild console.
2. When the deployment is completed, navigate to AWS CloudFormation console and look for a stack named “superset”.
3. From the **Outputs** tab, copy the URL under SupersetConsole and open it in a browser. This URL points to Apache Superset's web console. The default user name and password are below, please make sure the admin password is changed after the first login.
   - Username: admin
   - Password: admin

Connect the data product in Ada

To connect to the data product in Ada:

1. From the ADA user profile, create an API key for the ADA user who will access ADA data using Apache Superset.
2. Switch to Apache Superset and choose **Settings→Database Connections** and click **+Database** button.
3. From the Supported Databases drop-down list, choose **Other**. and enter the following URI in the SQLALCHEMY URI field:

Visualization

2. On the **Deploy Visualization solution - Apache Superset** page, choose **Deploy Apache Superset** to start the deployment. The deployment takes place in an AWS CodeBuild build. After the deployment is initiated, you can monitor the deployment progress from AWS CodeBuild console.
3. After the deployment is completed, go to the AWS CloudFormation console and navigate to the **superset** stack to verify that it has been deployed successfully.

For more information on how to connect a data product in ADA and viewing the visualization in Apache Superset, refer to the Apache Superset section in the Extensions guide.

Teardown

Use the **Teardown** page to permanently remove the Automated Data Analytics on AWS solution from your account.

> **Note**
>
> The Teardown option is only available for users with `root_admin` access.

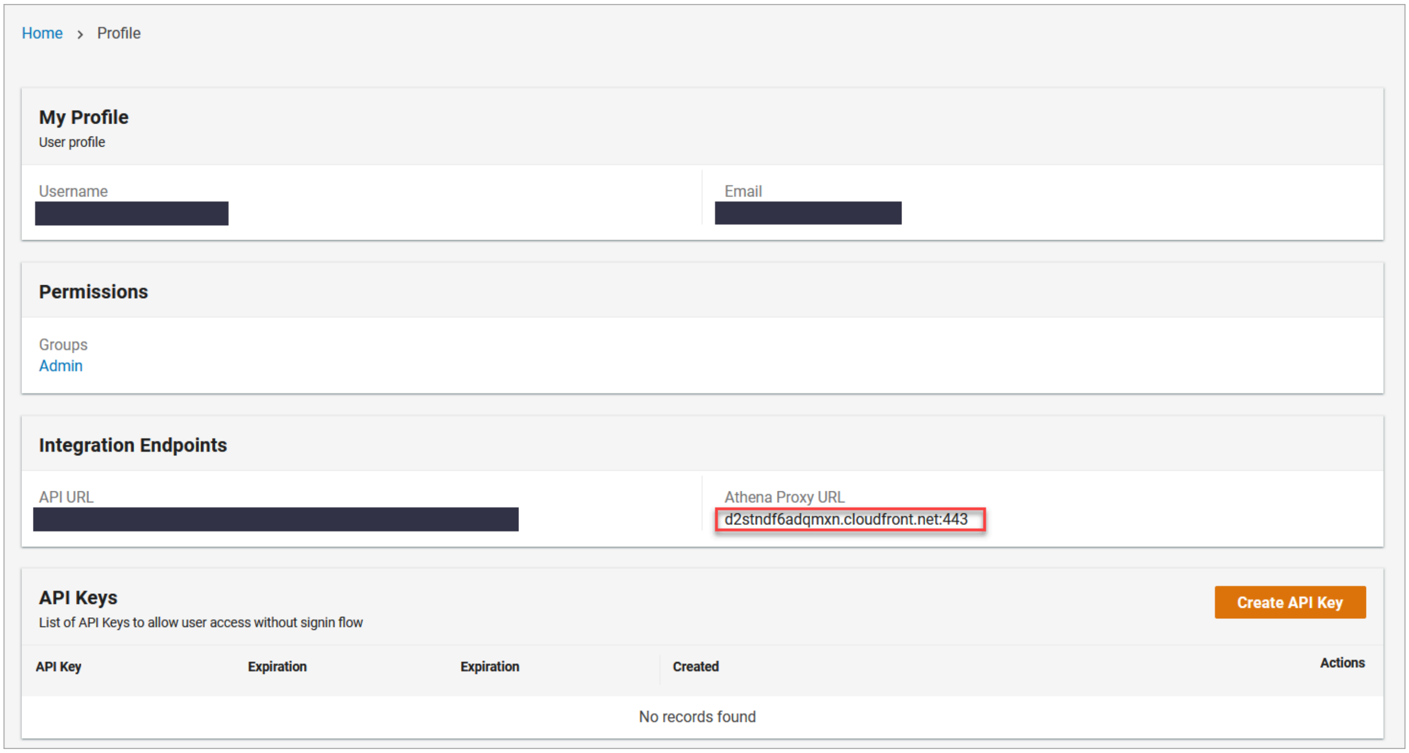
For more information, refer to the Teardown using the web UI section.

Connecting to third party tools

Automated Data Analytics on AWS provides native integrations with third party business intelligence tools such as Tableau and PowerBI. This allows users to visualize data within data products using these tools.

To integrate with these solutions, you must first establish a private connection through an API Key.

1. To find your connection URL and API keys, select **Profile** from the username dropdown on the top right.
2. On the My Profile page, copy the **Athena Proxy URL** in a text file.



My Profile

User profile

Username [redacted]

Email [redacted]

Permissions

Groups

Admin

Integration Endpoints

API URL [redacted]

Athena Proxy URL

d2stndf6adqmxn.cloudfront.net:443

API Keys

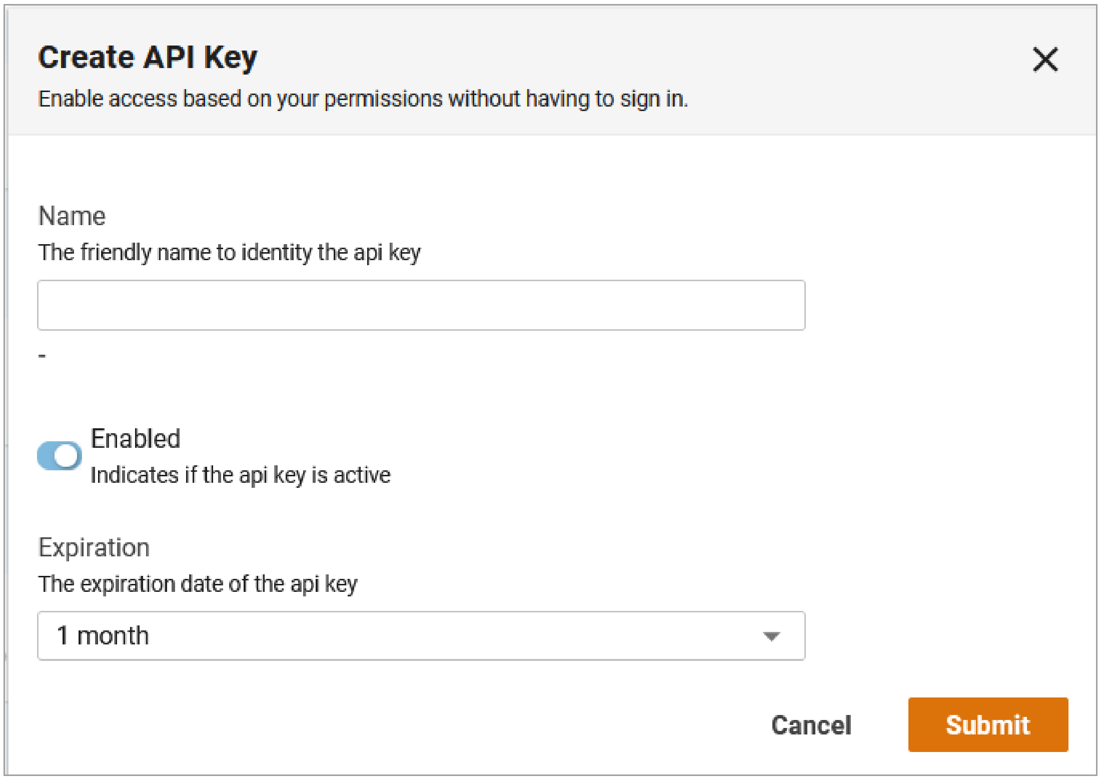
List of API Keys to allow user access without signin flow

API Key	Expiration	Expiration	Created	Actions
No records found				

Athena Proxy URL

3. Select **Create API Key**. The **Create API Key** dialog box is displayed.

4. On the Create API Key dialog, add a name, expiration date for the API Key, and select **Enable** to indicate the API Key is active.



Create API Key

Enable access based on your permissions without having to sign in.

Name

The friendly name to identity the api key

-

Enabled

Indicates if the api key is active

Expiration

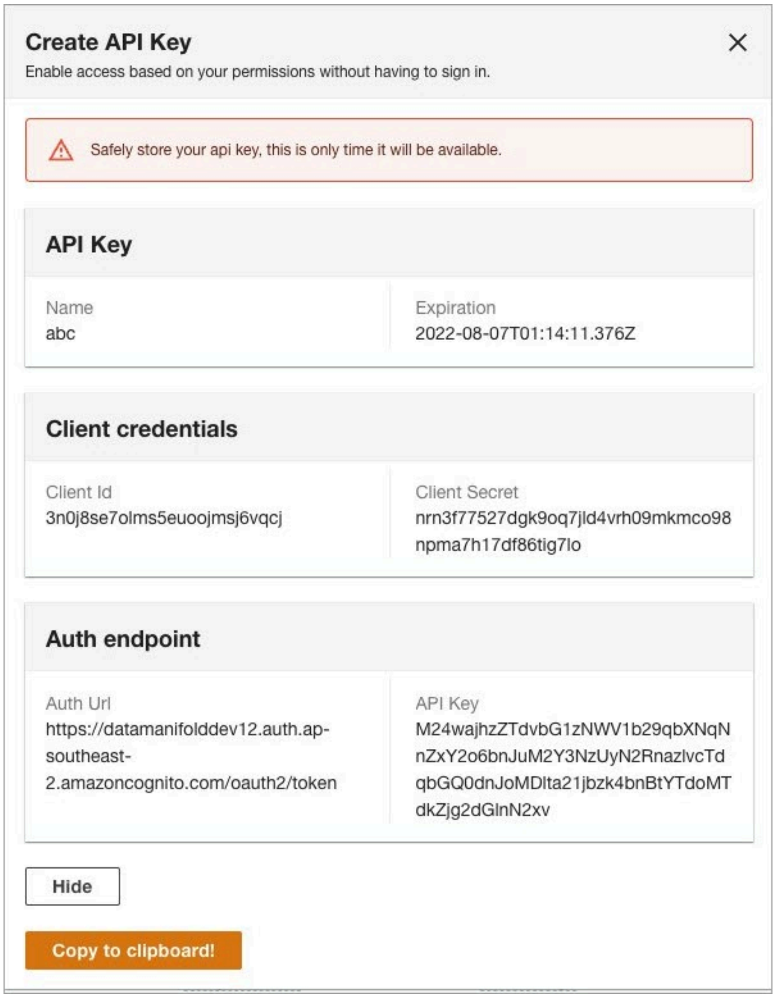
The expiration date of the api key

1 month

Create API key

5. Select **Submit**. An API key is created for you.

6. Copy the **API Key** to a text file along with the **Athena Proxy URL.**



Create API Key

Enable access based on your permissions without having to sign in.

Safely store your api key, this is only time it will be available.

API Key

Name
abc

Expiration
2022-08-07T01:14:11.376Z

Client credentials

Client Id
3n0j8se7olms5euoojmsj6vqcj

Client Secret
nrn3f77527dgk9oq7jld4vrh09mkmco98npma7h17df86tig7lo

Auth endpoint

Auth Url
https://datamanifolddev12.auth.ap-southeast-2.amazoncognito.com/oauth2/token

API Key
M24wajhzZTdvbG1zNWV1b29qbXNqNnZxY2o6bnJuM2Y3NzUyN2RnazlvcTdqbGQ0dnJoMDlta21jbzk4bnBtYTdoMTdmZjg2dGlnN2xv

API key details

After you have generated an API Key, you need to connect to it via the AWS Athena driver.

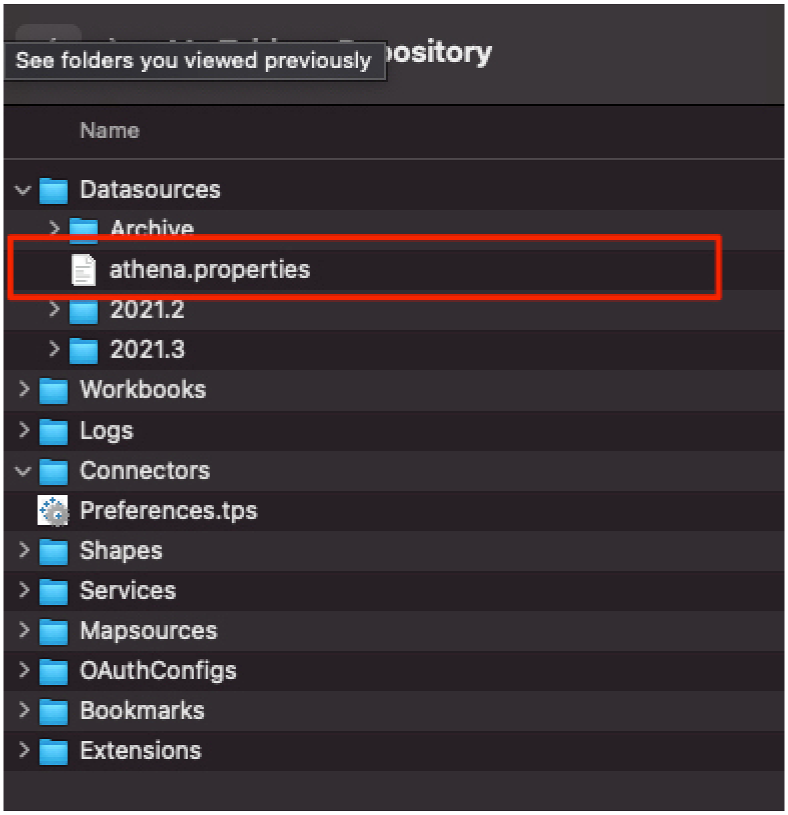
Tableau configuration

1. Download the JDBC Driver with AWS SDK (AthenaJDBC42_2.0.25.1001.jar) from the Amazon Athena User Guide on Amazon's website.
2. Extract the contents of the `.zip` file and copy the JDBC 4.2 version of the extracted .jar file to:
   - For Mac, copy the .jar file to the `~/Library/Tableau/Drivers` directory. You may need to create the directory if it doesn't already exist.
   - For Windows, copy the jar file to `C:\Program Files\Tableau\Drivers`.
   - For other operating systems, refer to the Tableau Direct Download page for more details.

3. To proxy through an Athena Proxy, you need to configure the JDBC driver by creating a file `athena.properties` and copy it to the following locations:

   - **Mac**: `~/Documents/My Tableau Repository\Datasources`
   - **Windows**: `C:\Users\<user-name>\Documents\My Tableau Repository\Datasources`
   - **Linux**: `~/Documents/My Tableau Repository/Datasources`

   Example of the file on MacOS.



   MacOS Athena properites file

4. Enter the following values in the `athena.properties` file.

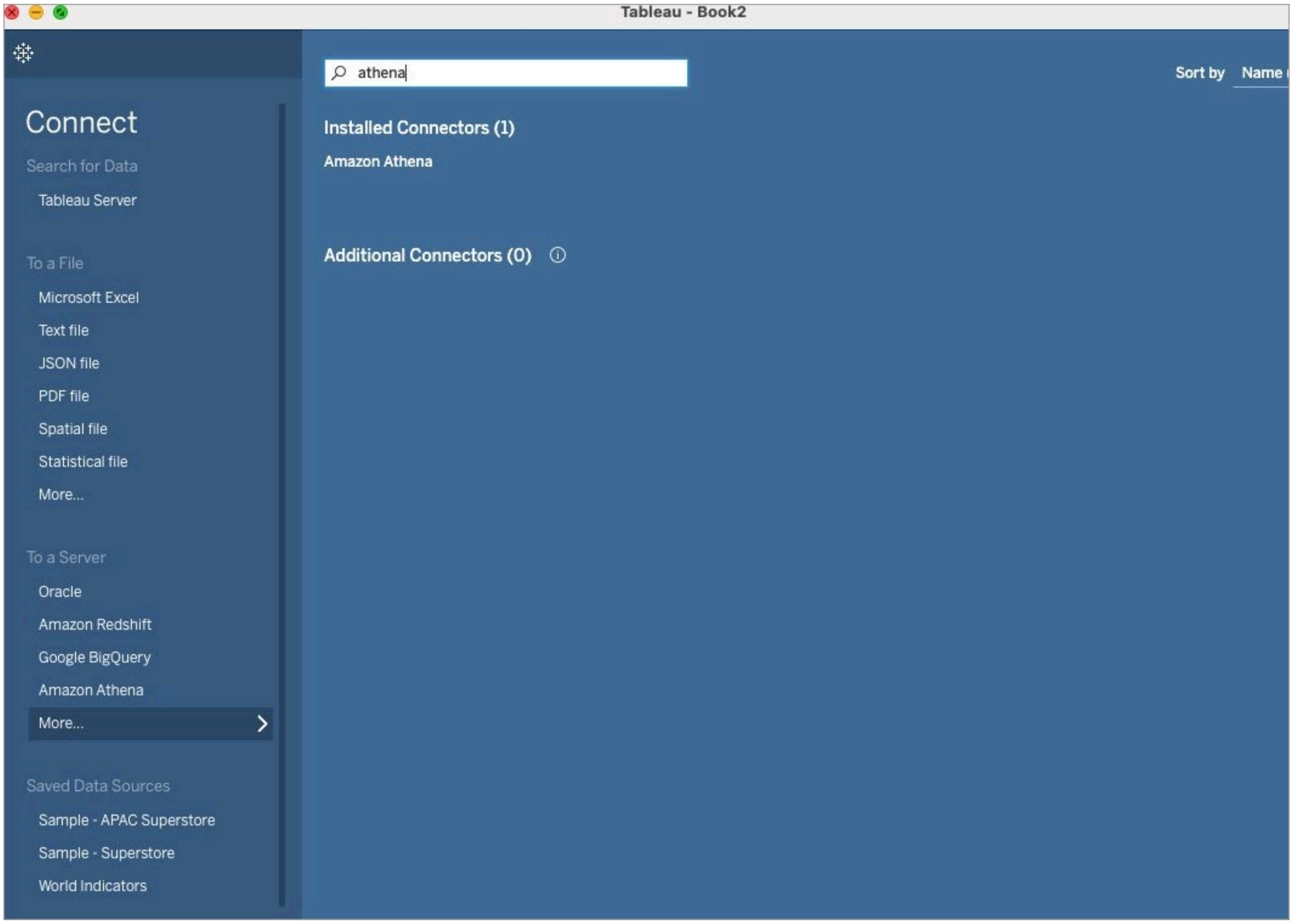
> **Note**
>
> Replace the `EndpointOverride` value with the **Athena Proxy URL** above.

```
MetadataRetrievalMethod=ProxyAPI
EndpointOverride=[Replace the endpoint with the Athena Proxy URL]
```

```
UseResultsetStreaming=0
```

For more information, refer to the athena.properties example.

5. Restart Tableau, and search and select **Amazon Athena** as the connector.



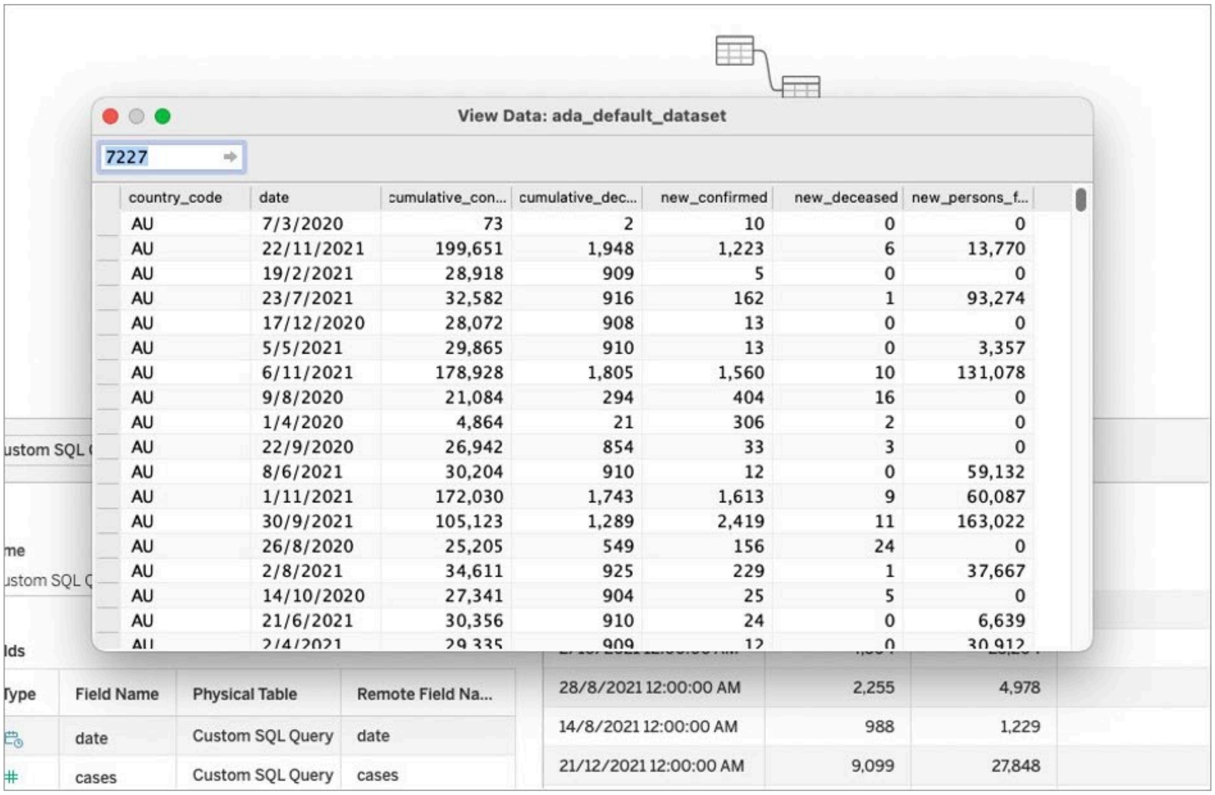
Search for Athena connector

6. Configure Amazon Athena using the following information:

- Server: athena.ap-southeast-2.amazonaws.com
- Port: 443
- S3 staging directory: s3://
- Access Key ID: api-key [Replace with the API key copied from the solution web UI]. *Ensure that you copy the API key without the quotation marks and do not include any spaces before or after the API key value.*
- Secret Access Key: Any 4 digit characters

7. Select **Sign In**. This displays the default AwsDataCatalog along with Automated Data Analytics on AWS domains in the **Catalog** drop down. If you select the Domain, it will list available data products within the **Databases** drop down.

8. Select the desired data product to display all tables within the data product.



View Data: ada_default_dataset

7227

country_code	date	cumulative_con...	cumulative_dec...	new_confirmed	new_deceased	new_persons_f...
AU	7/3/2020	73	2	10	0	0
AU	22/11/2021	199,651	1,948	1,223	6	13,770
AU	19/2/2021	28,918	909	5	0	0
AU	23/7/2021	32,582	916	162	1	93,274
AU	17/12/2020	28,072	908	13	0	0
AU	5/5/2021	29,865	910	13	0	3,357
AU	6/11/2021	178,928	1,805	1,560	10	131,078
AU	9/8/2020	21,084	294	404	16	0
AU	1/4/2020	4,864	21	306	2	0
AU	22/9/2020	26,942	854	33	3	0
AU	8/6/2021	30,204	910	12	0	59,132
AU	1/11/2021	172,030	1,743	1,613	9	60,087
AU	30/9/2021	105,123	1,289	2,419	11	163,022
AU	26/8/2020	25,205	549	156	24	0
AU	2/8/2021	34,611	925	229	1	37,667
AU	14/10/2020	27,341	904	25	5	0
AU	21/6/2021	30,356	910	24	0	6,639
AU	2/4/2021	29,335	909	12	0	30,912

Type	Field Name	Physical Table	Remote Field Na...
	date	Custom SQL Query	date
#	cases	Custom SQL Query	cases

28/8/2021 12:00:00 AM	2,255	4,978
14/8/2021 12:00:00 AM	988	1,229
21/12/2021 12:00:00 AM	9,099	27,848

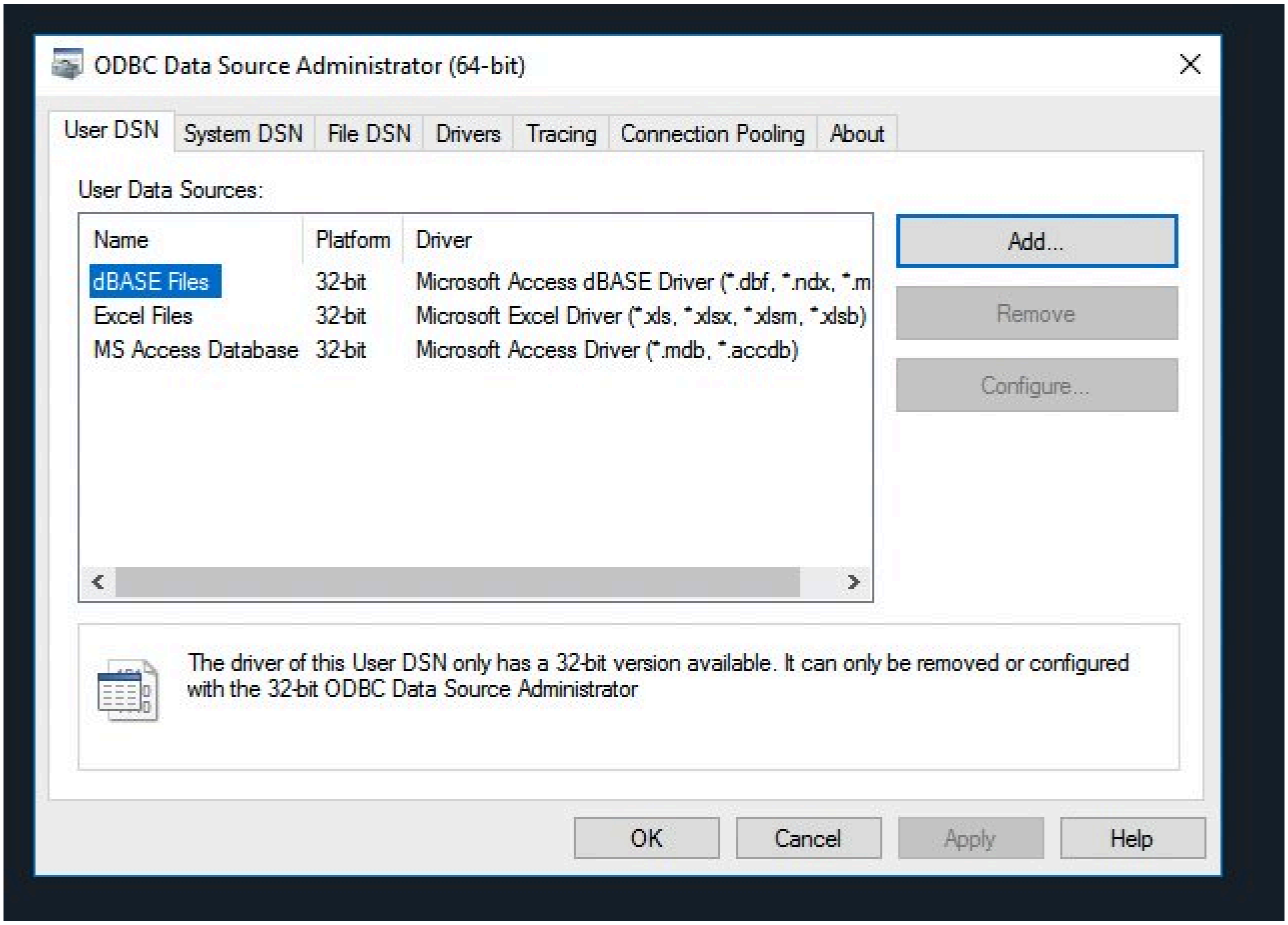
Display all tables

You can use **New Custom SQL** to run queries from Automated Data Analytics on AWS directly within Tableau.

PowerBI configuration

1. To configure integration with PowerBI using the Athena ODBC driver, download and install the driver from the Connecting to Amazon Athena with ODBC page.

2. Under the User DSN section, create a new ODBC data source using ODBC Data Source Administrator instructions.



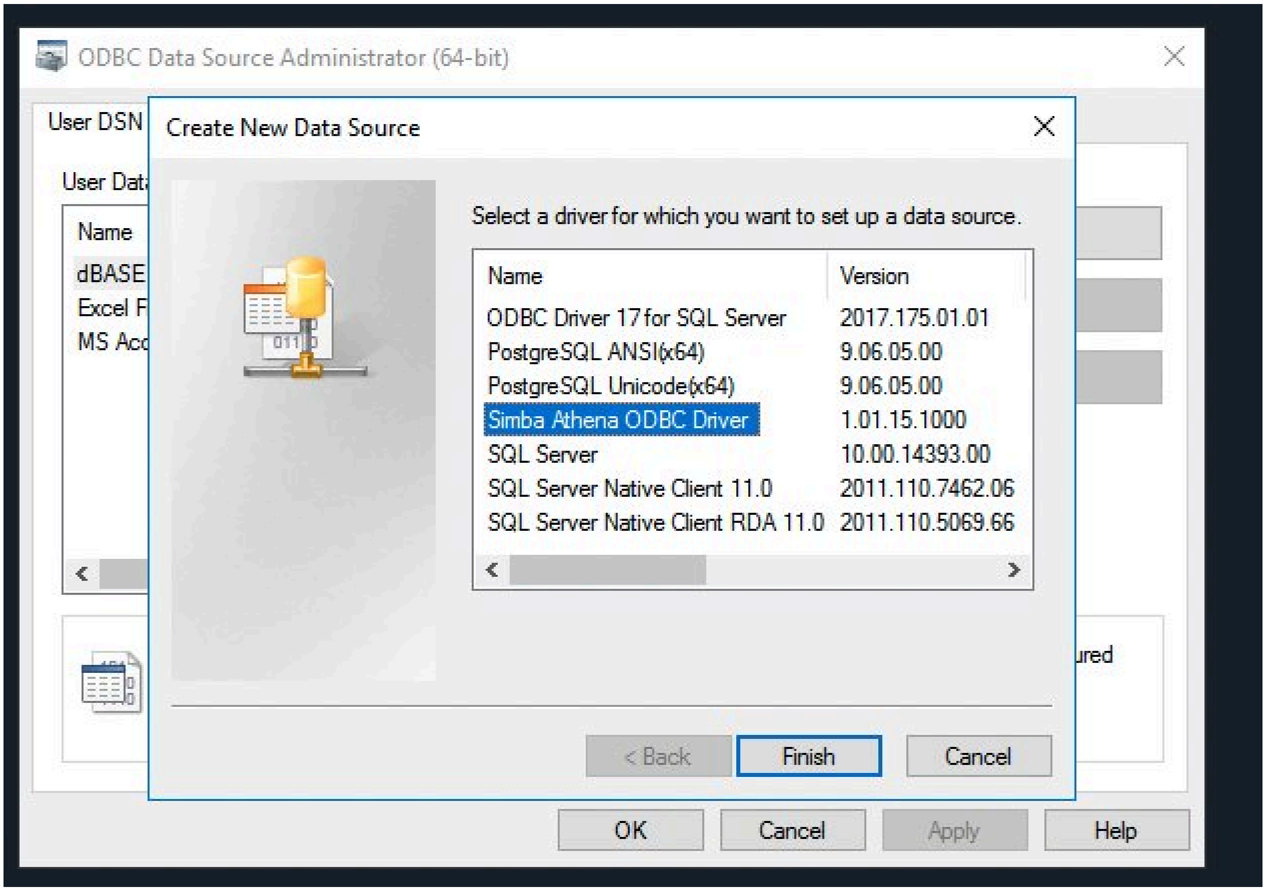
User Data Sources:

Name	Platform	Driver
dBASE Files	32-bit	Microsoft Access dBASE Driver (*.dbf, *.ndx, *.m
Excel Files	32-bit	Microsoft Excel Driver (*.xls, *.xlsx, *.xlsm, *.xlsb)
MS Access Database	32-bit	Microsoft Access Driver (*.mdb, *.accdb)

The driver of this User DSN only has a 32-bit version available. It can only be removed or configured with the 32-bit ODBC Data Source Administrator

ODBC data source administrator

3. From the driver list, select the installed Simba Athena ODBC Driver, and select **Finish**.

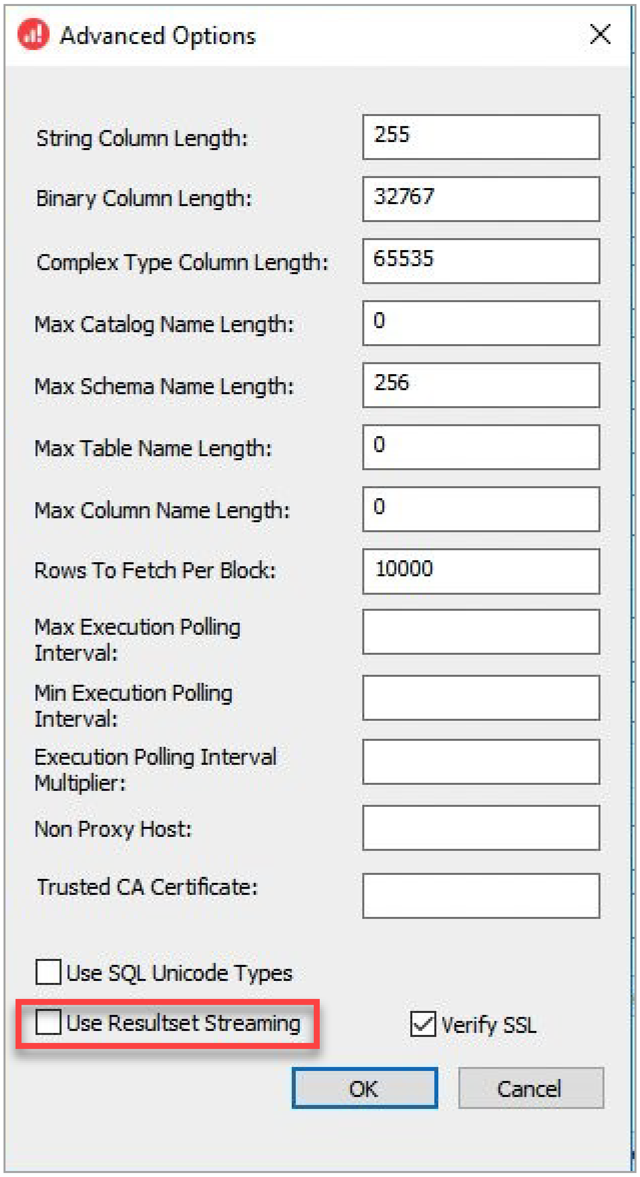


Select a driver for which you want to set up a data source.

Name	Version
ODBC Driver 17 for SQL Server	2017.175.01.01
PostgreSQL ANSI(x64)	9.06.05.00
PostgreSQL Unicode(x64)	9.06.05.00
Simba Athena ODBC Driver	1.01.15.1000
SQL Server	10.00.14393.00
SQL Server Native Client 11.0	2011.110.7462.06
SQL Server Native Client RDA 11.0	2011.110.5069.66

Simba Athena ODBC driver

4. Enter the data source name, e.g. Automated Data Analytics on AWS.
   - AWS Region: ap-southeast-2
   - Catalog: AwsDataCatalog
   - Schema: `default`
   - Workgroup: `primary`
   - Metadata Retrieval Method: `ProxyAPI`
   - S3 Output Location: `s3://`
   - Endpoint Override: Paste the Athena Proxy endpoint from Automated Data Analytics on AWS.
5. Select **Authentication Options** and select IAM Credentials for the Authentication Type.
   - **User**: api-key [Paste the API-Key from Automated Data Analytics on AWS copied above]
   - **Password**: `any digits`
6. Select **OK.**
7. In the **Advanced Options**, uncheck the `Use Resultset Streaming` and select **OK** .



String Column Length: 255

Binary Column Length: 32767

Complex Type Column Length: 65535

Max Catalog Name Length: 0

Max Schema Name Length: 256

Max Table Name Length: 0

Max Column Name Length: 0

Rows To Fetch Per Block: 10000

Max Execution Polling Interval:

Min Execution Polling Interval:

Execution Polling Interval Multiplier:

Non Proxy Host:

Trusted CA Certificate:

□ Use SQL Unicode Types

□ Use Resultset Streaming

☑ Verify SSL

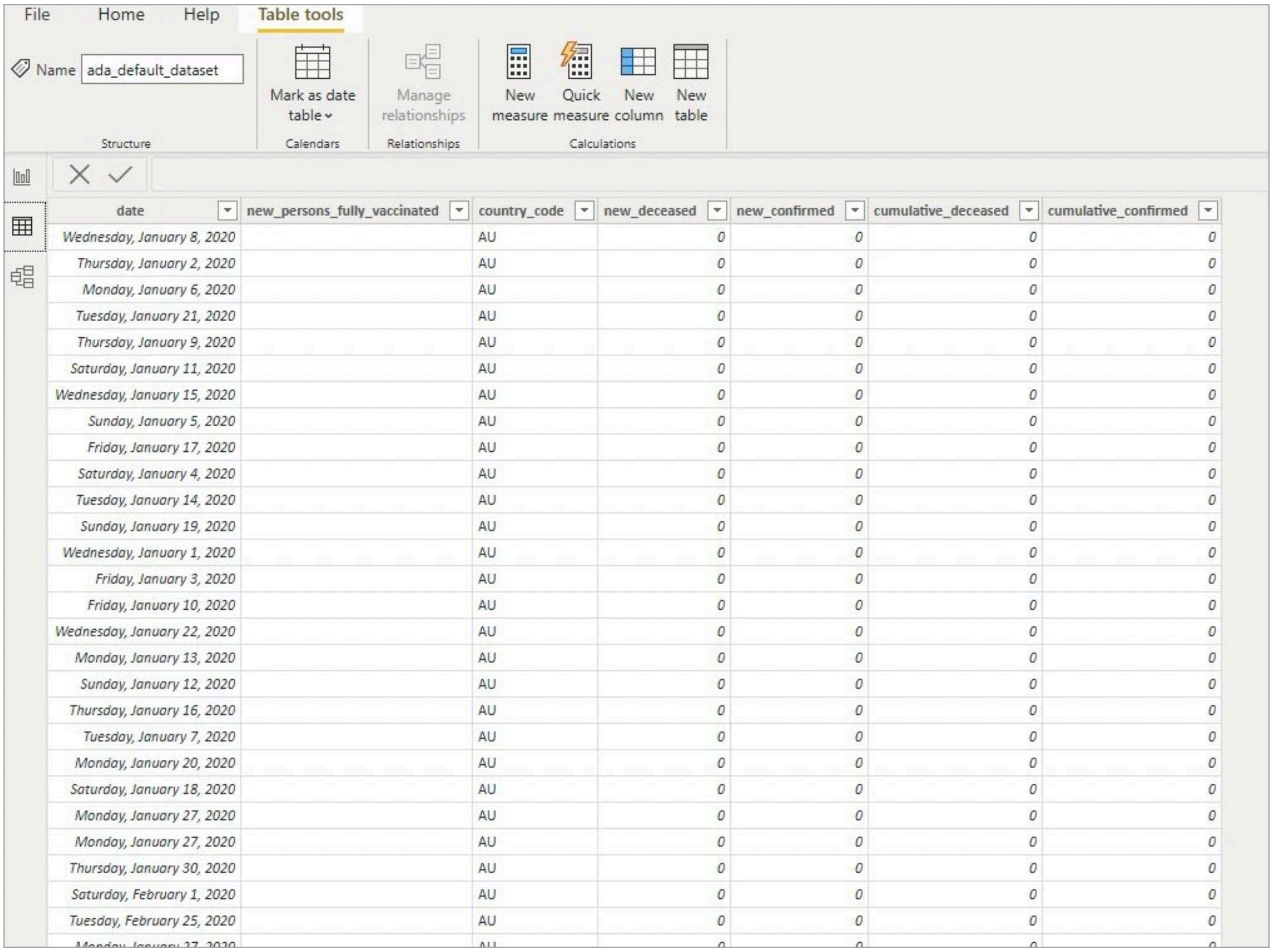
Advanced options

8. Test and select **OK**. Automated Data Analytics on AWS will be added as a new Data Source.
9. Open the Power BI application, and select ODBC as the data source type and select the Data source name created previously. (for example, **Automated Data Analytics**).

10Enter the `api-key` [API Key copied from the previous step].

11Enter a 4 digit password and select **Save.**You will see Domains loaded in the PowerBI navigator.

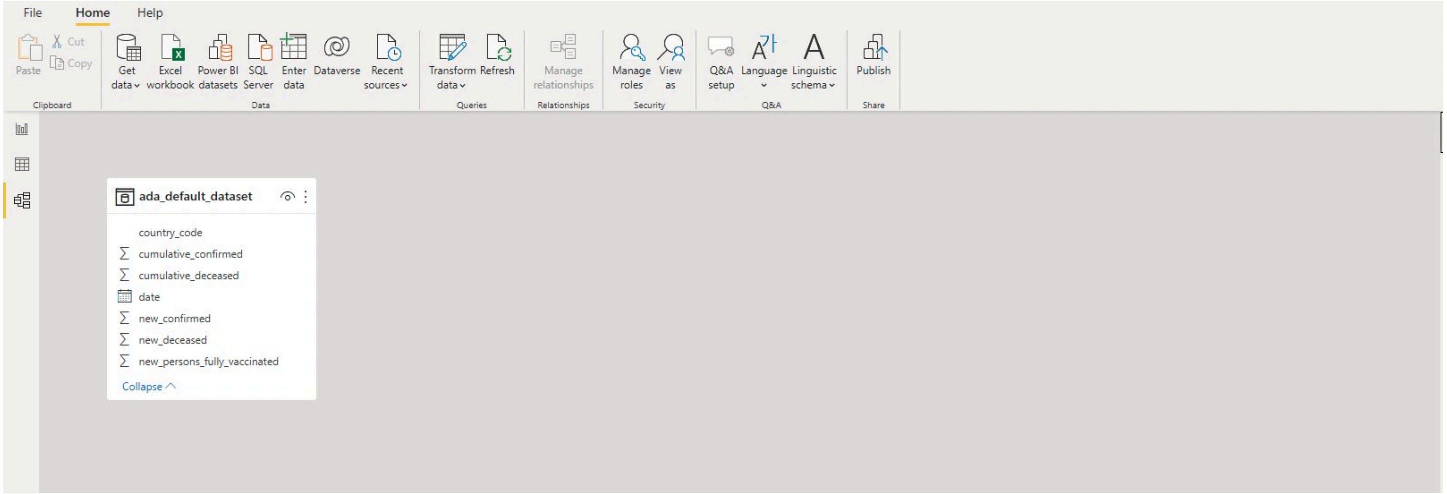
12Expand the domain to see data products and tables. Select the table to import as a dataset into PowerBI for further analysis.



Name: ada_default_dataset

date	new_persons_fully_vaccinated	country_code	new_deceased	new_confirmed	cumulative_deceased	cumulative_confirmed
Wednesday, January 8, 2020		AU	0	0	0	0
Thursday, January 2, 2020		AU	0	0	0	0
Monday, January 6, 2020		AU	0	0	0	0
Tuesday, January 21, 2020		AU	0	0	0	0
Thursday, January 9, 2020		AU	0	0	0	0
Saturday, January 11, 2020		AU	0	0	0	0
Wednesday, January 15, 2020		AU	0	0	0	0
Sunday, January 5, 2020		AU	0	0	0	0
Friday, January 17, 2020		AU	0	0	0	0
Saturday, January 4, 2020		AU	0	0	0	0
Tuesday, January 14, 2020		AU	0	0	0	0
Sunday, January 19, 2020		AU	0	0	0	0
Wednesday, January 1, 2020		AU	0	0	0	0
Friday, January 3, 2020		AU	0	0	0	0
Friday, January 10, 2020		AU	0	0	0	0
Wednesday, January 22, 2020		AU	0	0	0	0
Monday, January 13, 2020		AU	0	0	0	0
Sunday, January 12, 2020		AU	0	0	0	0
Thursday, January 16, 2020		AU	0	0	0	0
Tuesday, January 7, 2020		AU	0	0	0	0
Monday, January 20, 2020		AU	0	0	0	0
Saturday, January 18, 2020		AU	0	0	0	0
Monday, January 27, 2020		AU	0	0	0	0
Monday, January 27, 2020		AU	0	0	0	0
Thursday, January 30, 2020		AU	0	0	0	0
Saturday, February 1, 2020		AU	0	0	0	0
Tuesday, February 25, 2020		AU	0	0	0	0
Monday, January 27, 2020		AU	0	0	0	0

Expand domain



ada_default_dataset

- country_code
- cumulative_confirmed
- cumulative_deceased
- date
- new_confirmed
- new_deceased
- new_persons_fully_vaccinated

Select table to import

Set up ADA to access data sources within the VPC or on-premises network

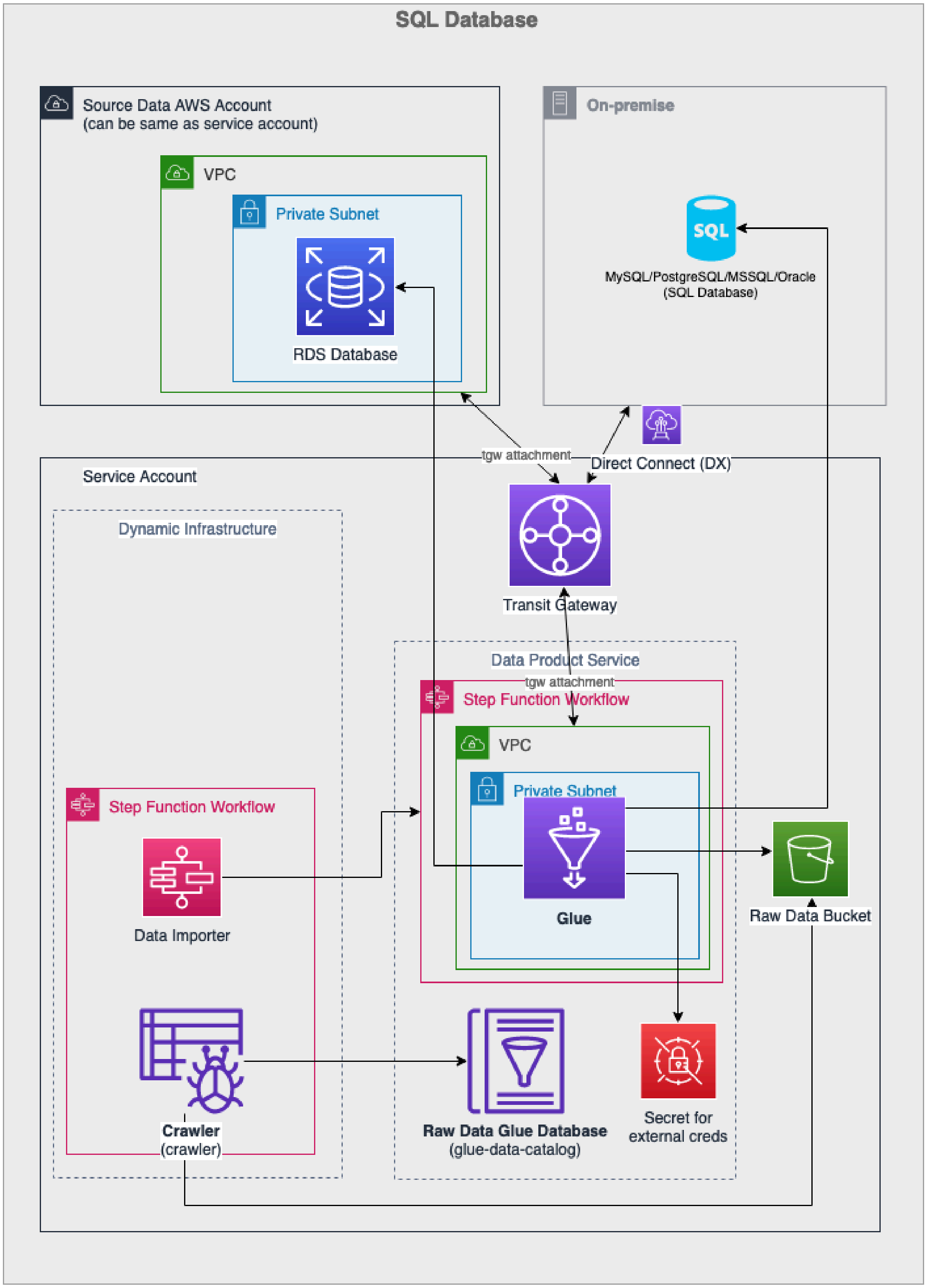
If you are leveraging a hybrid network configuration, you can configure ADA to access the data sources that reside within an isolated network in another AWS VPC or operating in your on-premise network. Using this, ADA users can import data directly from enterprise databases that may be running in the corporate network and use them for analysis.

Note

For security reasons, the network connectivity to the private network cannot be automatically granted or configured by ADA or by ADA users through the ADA user interface. It must be explicitly granted by your system administrators and configured for specific data sources to be accessible to ADA and its users. These steps requires basic network knowledge and access to AWS Console page of the account that ADA was deployed to, as well as access to the corporate network infrastructure.

To facilitate network connectivity, ADA provides its Data Ingress Gateway, which is backed up by AWS Transit Gateway service and provides a unified central place for your network to be connected to ADA. It also provides isolation between individual data sources.

The following diagram displays the network topology for the ADA Data Ingress Gateway.

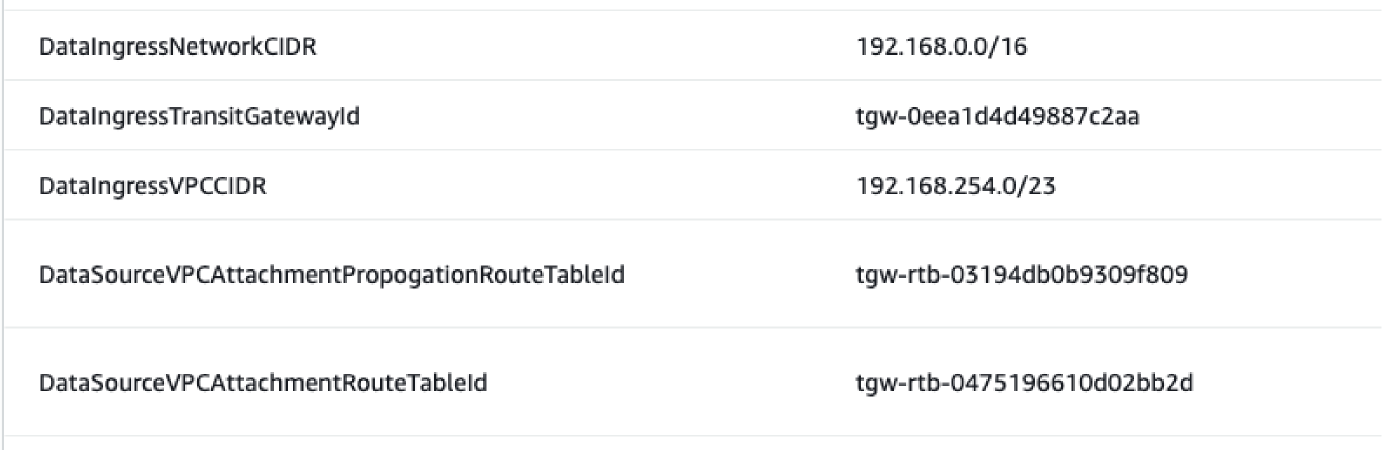


ADA Data Ingress Gateway

This section describes the steps for your system administrator or ADA admin to connect ADA with other data sources in an isolated VPC or on-premises network.

Before you begin

1. After you deploy ADA, find the Data Ingress Network information from CloudFormation console. Navigate to the **ADA stack > Outputs** tab. Make a note of this information for later use.



DataIngressNetworkCIDR	192.168.0.0/16
DataIngressTransitGatewayId	tgw-0eea1d4d49887c2aa
DataIngressVPCCIDR	192.168.254.0/23
DataSourceVPCAttachmentPropogationRouteTableId	tgw-rtb-03194db0b9309f809
DataSourceVPCAttachmentRouteTableId	tgw-rtb-0475196610d02bb2d

   ADA Data Ingress Network information

2. Plan your network Classless Inter-Domain Routing (CIDR) allocation.

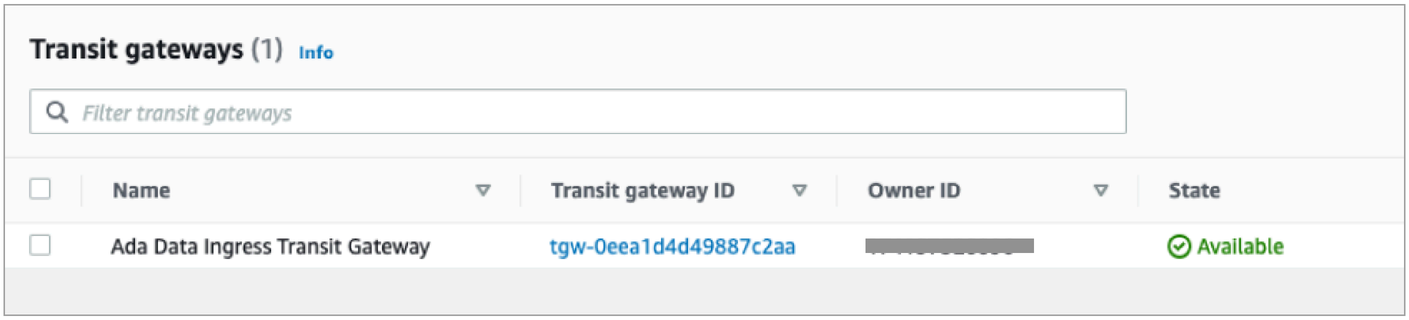
   - By default, ADA assigns `192.168.254.0/23` for its own data ingress VPC, so the CIDR of the data sources subnets to be connected to ADA cannot use this CIDR or have any overlaps with it.
   - The ADA Data Ingress Network has a default CIDR of `192.168.0.0/16`. If you can route your data source to/from this CIDR range, you can connect it to the ADA Data Ingress Network without any further changes. However, if the default Data Ingress Network CIDR or the Data Ingress VPC CIDR conflict with your network CIDR allocation plan, you can configure them to different CIDR by using the advanced configuration settings and deploy the solution from the source code with CDK toolkit.

> **Note**
>
> The CloudFormation deployment format does not support the advanced configuration settings. For more information on how change advanced configuration settings and how to deploy with CDK, refer to the CDK Deployment section in this document.

Set up AWS Transit Gateway service

For this tutorial, let us consider an example that a RDS database to be connected to ADA as a data source resides in an isolated subnet in another VPC with CIDR `192.168.1.0/24`.

1. To view the Ada Data Ingress Gateway, navigate to the AWS VPC page from your AWS Console and select **Transit Gateway** on the left navigation panel. The transit gateway for ADA Data Ingress is displayed.



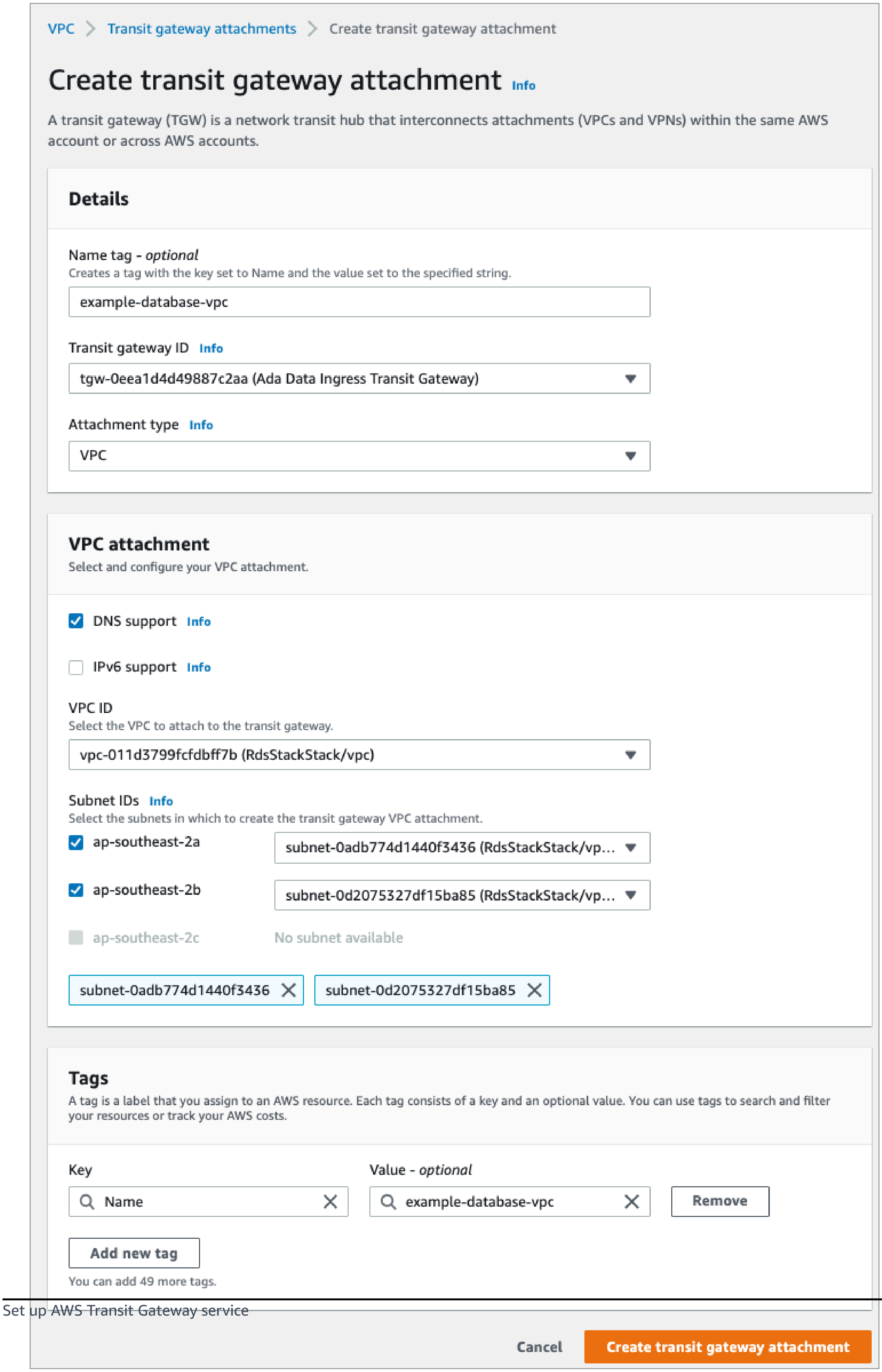
Transit gateways (1) Info

Filter transit gateways

□	Name	Transit gateway ID	Owner ID	State
□	Ada Data Ingress Transit Gateway	tgw-0eea1d4d49887c2aa	[redacted]	✓ Available

ADA Transit Gateway List

2. Select **Transit Gateway attachments** on the left navigation panel, to display Transit gateway attachments, and then click the **Create transit gateway attachment**t button from the top right corner.
3. Fill in the information about the VPC that the data source database resides in. Choose the subnets that the RDS database resides within or the subnets that can route the traffic to the RDS database subnets. Once completed, click the Create transit gateway attachment button at the bottom of the page.



VPC > Transit gateway attachments > Create transit gateway attachment

Create transit gateway attachment Info

A transit gateway (TGW) is a network transit hub that interconnects attachments (VPCs and VPNs) within the same AWS account or across AWS accounts.

Details

Name tag - *optional*
Creates a tag with the key set to Name and the value set to the specified string.
example-database-vpc

Transit gateway ID Info
tgw-0eea1d4d49887c2aa (Ada Data Ingress Transit Gateway)

Attachment type Info
VPC

VPC attachment
Select and configure your VPC attachment.

☑ DNS support Info

□ IPv6 support Info

VPC ID
Select the VPC to attach to the transit gateway.
vpc-011d3799fcfdbff7b (RdsStackStack/vpc)

Subnet IDs Info
Select the subnets in which to create the transit gateway VPC attachment.

☑ ap-southeast-2a	subnet-0adb774d1440f3436 (RdsStackStack/vp...
☑ ap-southeast-2b	subnet-0d2075327df15ba85 (RdsStackStack/vp...
□ ap-southeast-2c	No subnet available

subnet-0adb774d1440f3436 ✕ subnet-0d2075327df15ba85 ✕

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - *optional*	
Name	example-database-vpc	Remove

Add new tag

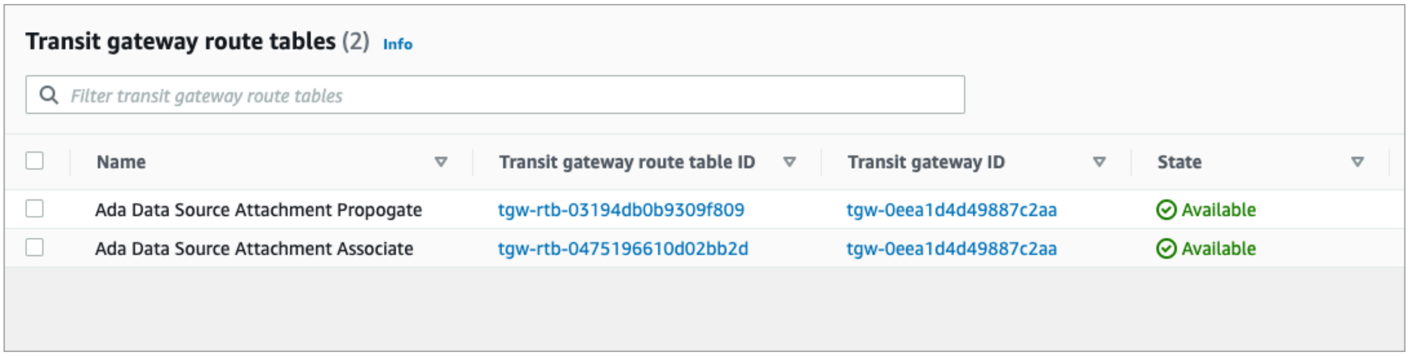
You can add 49 more tags.

Cancel Create transit gateway attachment

Create Transit Gateway Attachment

4. After the transit gateway attachment is created, it is displayed on the **Transit gateway attachments** page. Confirm that the state displays **Available**. You can also share the Transit Gateway with another AWS Account using Resource Sharing. Before you share, make sure the request is approved in the ADA Transit Gateway. Once ready, the page displays this information.

5. From the left navigation panel, click **Transit gateway route tables**. ADA provides two route tables by default. The new Transit Gateway Attachment for the data source VPC must be set to associate and propagate routes into those two route tables respectively.

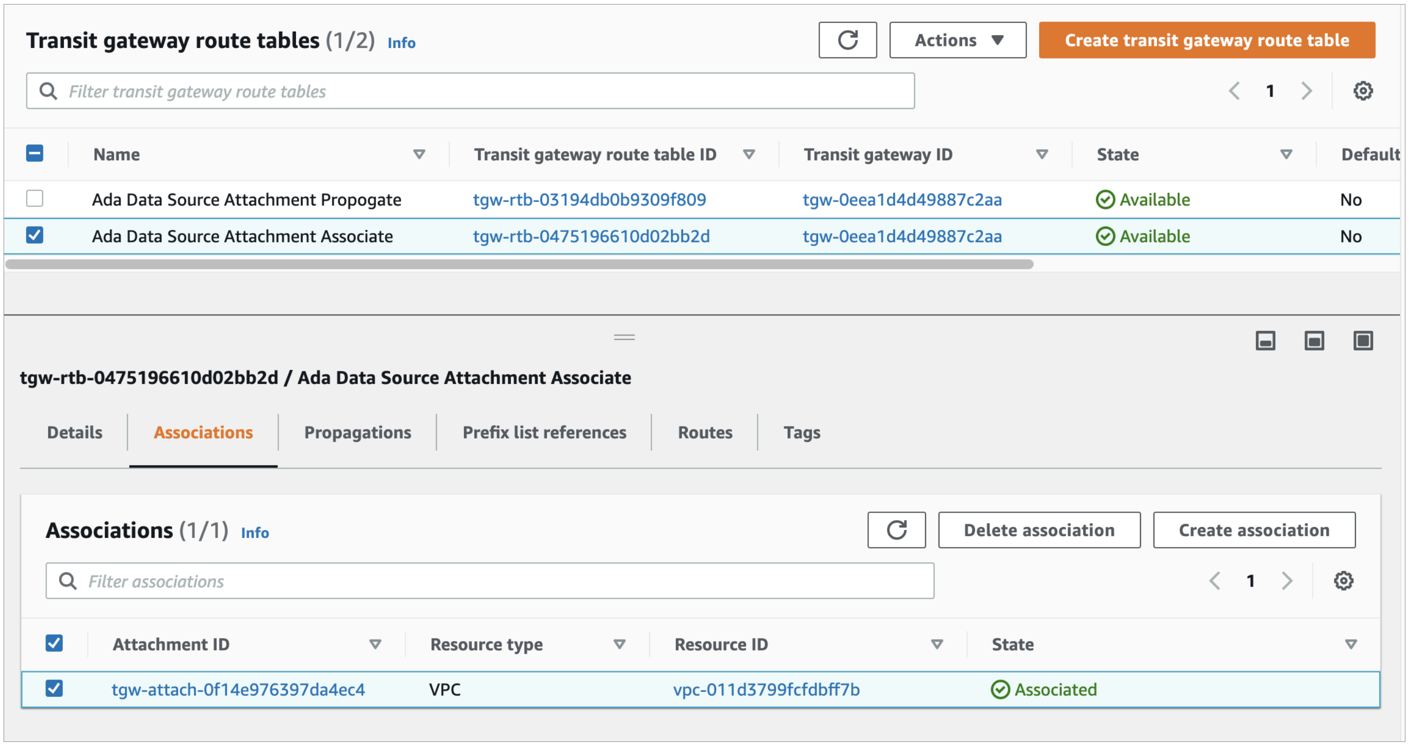


Transit gateway route tables (2) Info

□	Name	Transit gateway route table ID	Transit gateway ID	State
□	Ada Data Source Attachment Propogate	tgw-rtb-03194db0b9309f809	tgw-0eea1d4d49887c2aa	✓ Available
□	Ada Data Source Attachment Associate	tgw-rtb-0475196610d02bb2d	tgw-0eea1d4d49887c2aa	✓ Available

Transit Gateway Route Tables

6. To associate a route table with the data source VPC attachment:
   - Select the **ADA Data Source Attachment Associate** route table checkbox.
   - In the lower panel, select the **Associations** tab and click the **Create association** button.
   - Select the Data Source VPC attachment created in step 3 and click the **Create association** button.


Transit gateway route tables (1/2) Info
Actions
Create transit gateway route table
Filter transit gateway route tables

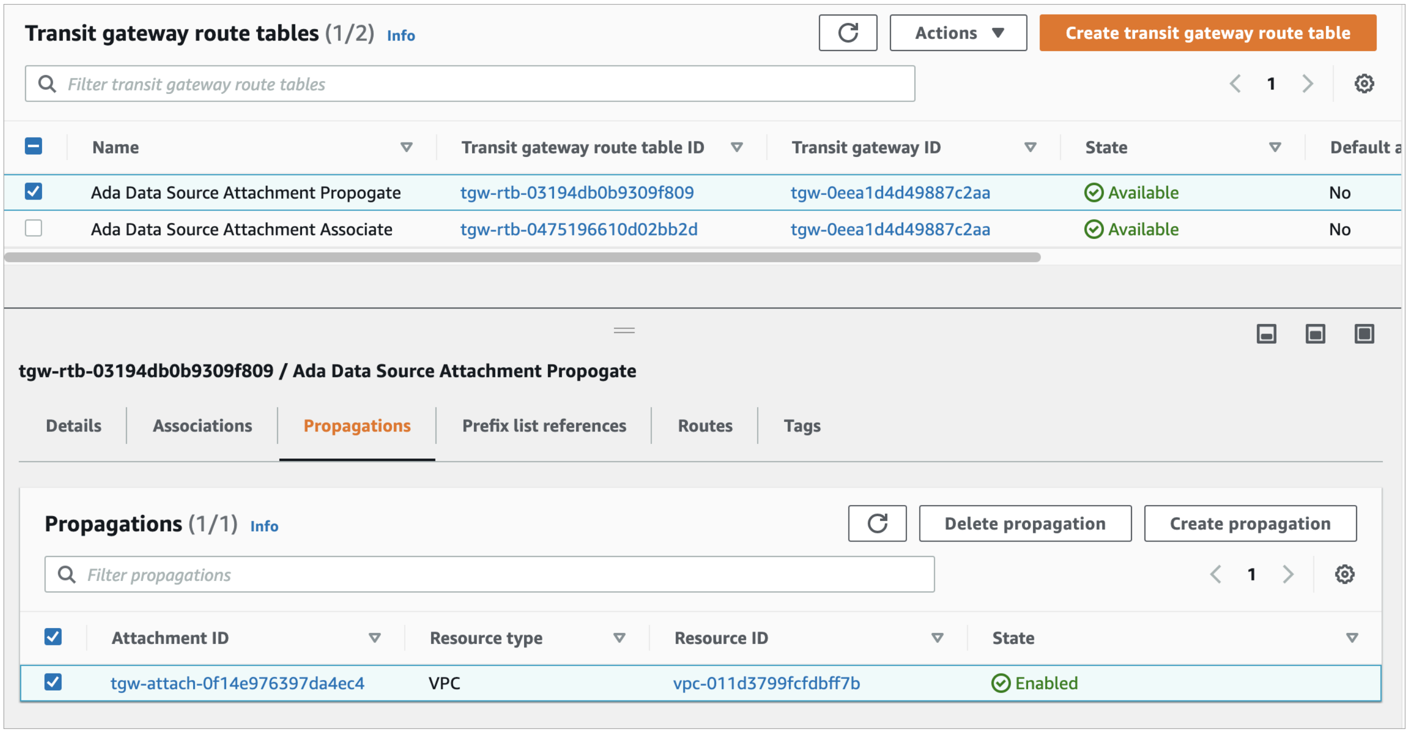
	Name	Transit gateway route table ID	Transit gateway ID	State	Default
□	Ada Data Source Attachment Propogate	tgw-rtb-03194db0b9309f809	tgw-0eea1d4d49887c2aa	Available	No
☑	Ada Data Source Attachment Associate	tgw-rtb-0475196610d02bb2d	tgw-0eea1d4d49887c2aa	Available	No

tgw-rtb-0475196610d02bb2d / Ada Data Source Attachment Associate

	Attachment ID	Resource type	Resource ID	State
☑	tgw-attach-0f14e976397da4ec4	VPC	vpc-011d3799fcfdbff7b	Associated

Transit Gateway Route Association

7. To set the propagation route table of the data source VPC attachment:
    - Select the **ADA Data Source Attachment Propagate** route table checkbox.
    - In the lower panel, select the **Propagations** tab and click the **Create propagation** button.
    - Select the Data Source VPC attachment created in step 3 and click the **Create propagtion** button.



Transit gateway route tables (1/2) Info

■	Name	Transit gateway route table ID	Transit gateway ID	State	Default a
☑	Ada Data Source Attachment Propogate	tgw-rtb-03194db0b9309f809	tgw-0eea1d4d49887c2aa	✓ Available	No
☐	Ada Data Source Attachment Associate	tgw-rtb-0475196610d02bb2d	tgw-0eea1d4d49887c2aa	✓ Available	No

tgw-rtb-03194db0b9309f809 / Ada Data Source Attachment Propogate

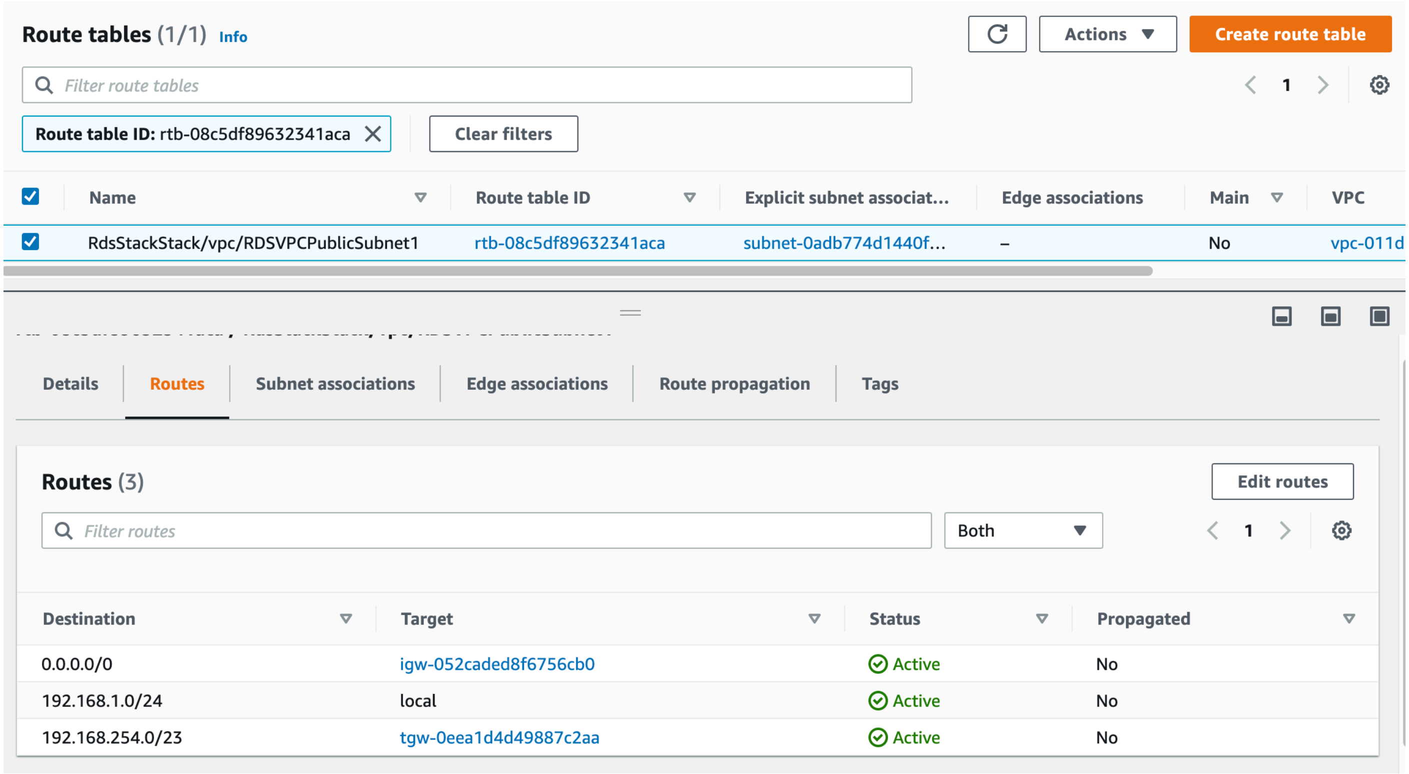
Propagations (1/1) Info

☑	Attachment ID	Resource type	Resource ID	State
☑	tgw-attach-0f14e976397da4ec4	VPC	vpc-011d3799fcfdbff7b	✓ Enabled

Transit Gateway Route Propagation

8. Add the route entry in the route table in the data source subnet, to route traffic to the newly created transit gateway attachment.

> **Note**
>
> You will need to complete this step in the VPC that the data source resides in. Normally you should add the route entry in to the route table of the subnets that was chosen in step 3. The static route entry should route any traffic that is headed to Ada Data Ingress VPC CIDR (`192.168.254.0/23` in this case) to the newly created transit gateway attachment. Add the same static route to the route table for both subnets.



☑	Name	Route table ID	Explicit subnet associat...	Edge associations	Main	VPC
☑	RdsStackStack/vpc/RDSVPCPublicSubnet1	rtb-08c5df89632341aca	subnet-0adb774d1440f...	–	No	vpc-011d

Destination	Target	Status	Propagated
0.0.0.0/0	igw-052caded8f6756cb0	✓ Active	No
192.168.1.0/24	local	✓ Active	No
192.168.254.0/23	tgw-0eea1d4d49887c2aa	✓ Active	No

Route Tables

Test the network connectivity

1. Create a data product from importing the data from a database in the isolated VPC.
2. Choose the corresponding database connector from the list and fill in the database information.

If you have successfully completed the network connectivity setup, data from the database will be available on the preview screen.

Data connectors guide

You can ingest data from a range of data sources via an intuitive wizard into ADA. Automated Data Analytics on AWS currently supports the following data connectors:

Note

AWS will periodically build and release connectors for additional data sources and integrate connectors built by the open source community through GitHub.

Connector	Description	Preview*	Source Query**
AWS CloudTrail	Source data from AWS CloudTrail. It supports filtering on importing for different CloudTrail Event Types and also supports importing from a different account. The connector also supports incremental daily import when it is set to scheduled mode. Due to data volumes that could be potentially massive, the automatic PII detection feature is currently unavailable for CloudTrail connector.	X	X

Connector	Description	Preview*	Source Query**
Amazon CloudWatch	Source data from Amazon CloudWatch logs. The connector queries CloudWatch Logs Groups with the specific CloudWatc h query and then imports the result logs into Automated Data Analytics on AWS. It also supports increment al importing by schedule.	X	X
Amazon Kinesis Data Stream	Source data from an existing Kinesis DataStream within the same account in the same region. If data stream is encrypted using custom managed key, decrypt access is required for the Automated Data Analytics application.		
Amazon Redshift	Source data from an existing Amazon Redshift Cluster or Amazon Redshift Serverless table in an AWS account.		

Connector	Description	Preview*	Source Query**
Amazon S3	Source data from existing Amazon S3 data supports the same data file types and formats as AWS Glue. The provided object path must be readable by the Automated Data Analytics on AWS application.	X	X
DynamoDB	Source data from AWS DynamoDB tables. It supports importing data from DynamoDB tables from a different account. The data is transferred into a S3 bucket that is managed by Automated Data Analytics for AWS.	X	X

Connector	Description	Preview*	Source Query**
File Upload	Source data from file upload supports .csv, .json, .par and .gz file formats through the UI. The uploaded files are stored in a shared Amazon S3 buckets for all uploads and only accessible through the solution. Once a file is uploaded as source data, it is treated the same as Amazon S3 sourced data.	X	X

Connector	Description	Preview*	Source Query**
Google Analytics	Source data from Google Analytics supports import of analytics dimensions and metrics. It supports both full import and incremental import. The authentication requires a service account to be provisioned in order to connect to the API for continuous import. For details on creating service account for Google Analytics, refer to [Create a client ID for Google Analytics API](#) topic.	X	

Connector	Description	Preview*	Source Query**
Google BigQuery	Source data from Google BigQuery supports any queries that is runnable within BigQuery. Service account that has read permission to the BigQuery API is required for authentication through Automated Data Analytics on AWS. To provision a service account in Google BigQuery, refer to Managing service accounts page for more details.	X	

Connector	Description	Preview*	Source Query**
Google Storage	Data from Google Storage supports folder level import. The connector uses RSync API to synchronize between source bucket in Google Storage and destination in shared S3 Bucket managed by Automated Data Analytics on AWS. Data removed from the source bucket will also be removed from the destination.		
Microsoft SQL Server	Source data from a Microsoft SQL Server database. The connector uses JDBC to connect to the target database and import the data from the specific table into a S3 bucket managed by Automated Data Analytics on AWS.	X	

Connector	Description	Preview*	Source Query**
MongoDB	Source data from MongoDB or Amazon DocumentDB. The connector supports server TLS and client certificate. It also offers a bookmark field to support incremental importing.	X	X
MySQL5	Source data from MySQL Server 5. The connector uses JDBC to connect to the target database and import the data from the specific table into a S3 bucket managed by Automated Data Analytics on AWS.	X	
Oracle	Source data from Oracle databases. The connector uses JDBC to connect to the target database and import the data from the specific table into a S3 bucket managed by Automated Data Analytics on AWS.		

Connector	Description	Preview*	Source Query**
PostgreSQL	Source data from a PostgreSQL database. The connector uses JDBC to connect to the target database and import the data from the specific table into a S3 bucket managed by Automated Data Analytics on AWS.	X	

* Indicates the source type supports **preview** feature for fast schema inference and managing schema during full import process.

** Indicates the source type supports **source query** feature to grant creator direct query access to the original source data through queries.

To connect a data source to the Automated Data Analytics on AWS solution, you will need to create a domain, and then create a data product to import the dataset. You can choose the data source when you create the data product.

Refer to the following sections for more information on the type of connectors and steps involved to ingest data from these sources.

> **Note**
>
> The data connectors guide only shows the steps involved in importing a data source.

AWS CloudTrail

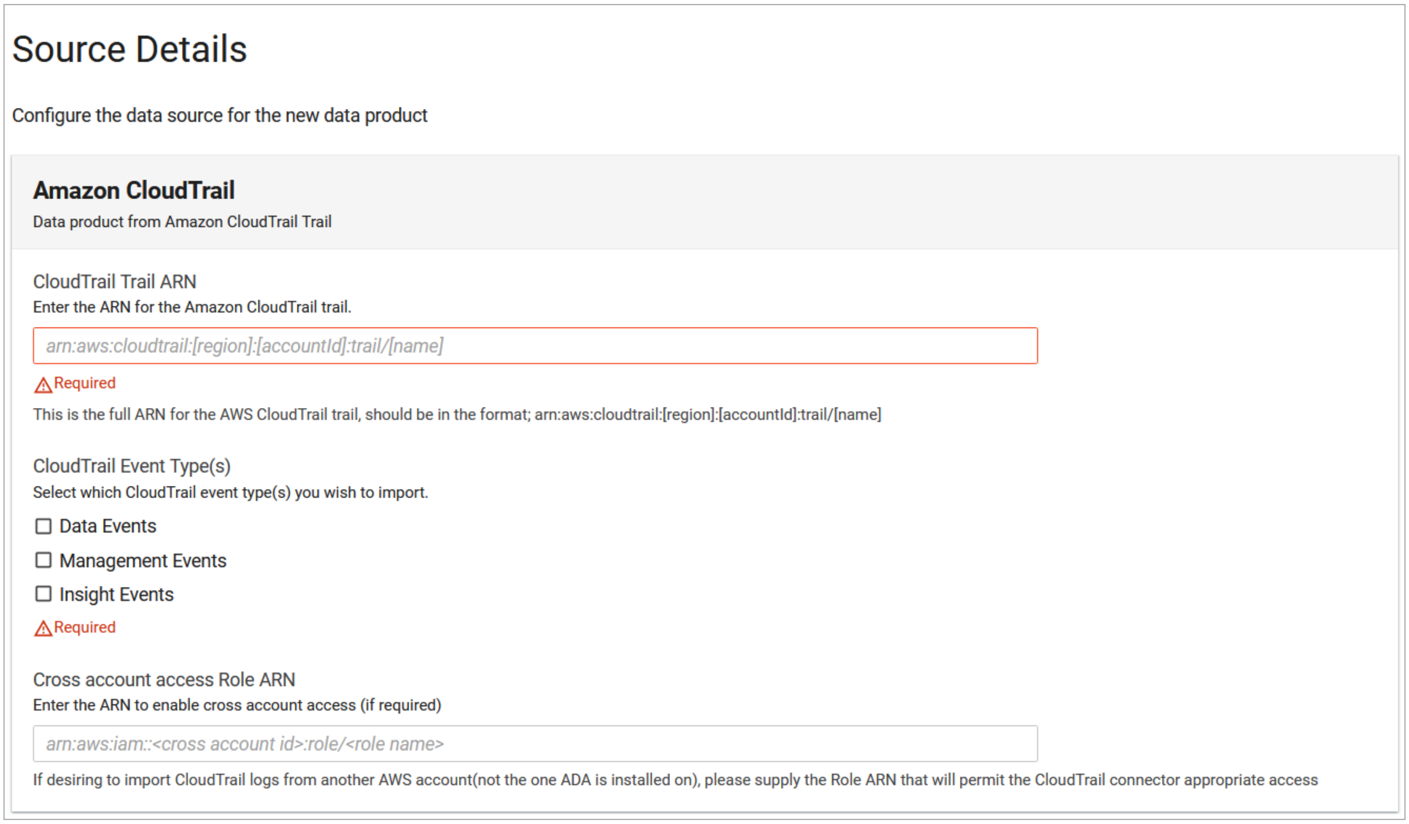
The AWS CloudTrail data connector allows you to ingest data from trails in AWS CloudTrail from the same account or different accounts..

Note

Automatic PII is not available with CloudTrail.

To import data from CloudTrail:

1. On the **Source Details** page, enter the CloudTrail ARN, and select the CloudTrail Event Types you want to import. You can choose from **Data Event, Management Events** or **Insight Events.** If the CloudTrail data to be imported is from a different AWS account to where the ADA solution is deployed, you must provide a Cross account access role ARN. See Working with cross account CloudTrails below for how to configure cross account access role.



Source Details

Configure the data source for the new data product

Amazon CloudTrail

Data product from Amazon CloudTrail Trail

CloudTrail Trail ARN

Enter the ARN for the Amazon CloudTrail trail.

arn:aws:cloudtrail:[region]:[accountId]:trail/[name]

⚠Required

This is the full ARN for the AWS CloudTrail trail, should be in the format; arn:aws:cloudtrail:[region]:[accountId]:trail/[name]

CloudTrail Event Type(s)

Select which CloudTrail event type(s) you wish to import.

☐ Data Events

☐ Management Events

☐ Insight Events

⚠Required

Cross account access Role ARN

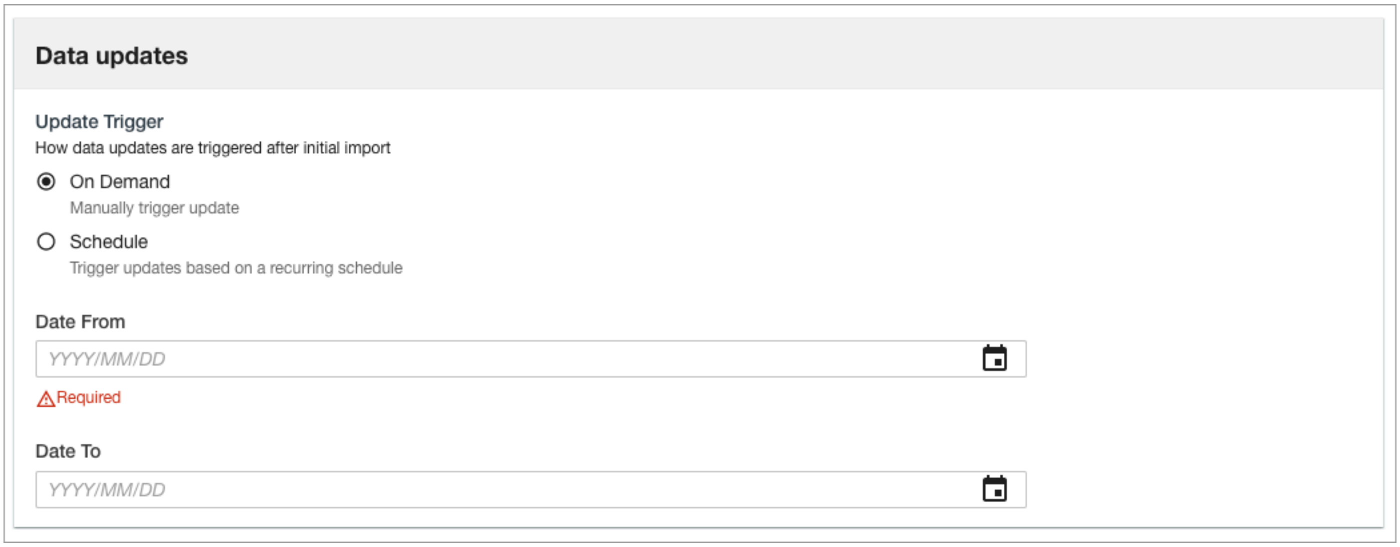
Enter the ARN to enable cross account access (if required)

arn:aws:iam::<cross account id>:role/<role name>

If desiring to import CloudTrail logs from another AWS account(not the one ADA is installed on), please supply the Role ARN that will permit the CloudTrail connector appropriate access

AWS CloudTrail connector

2. Select the data updates frequency and date range.



Data updates

Update Trigger

How data updates are triggered after initial import

● On Demand

Manually trigger update

○ Schedule

Trigger updates based on a recurring schedule

Date From

YYYY/MM/DD

⚠Required

Date To

YYYY/MM/DD

AWS CloudTrail connector

- If you select **On Demand** , enter a value for **Data From** (required). If you omit **Date To** , it will use the current date by default.
- If you select **Schedule**, enter a value for **Date From** and the data product will periodically import data incrementally based on the chosen interval.

> **Note**
>
> The date used for **Date From** and **Date To** is in UTC format, and you can only import data for a completed day. If you set the **Date To** as the current day, the data imported is up to the beginning of the current day. We recommend using **daily** as the shortest import interval.

Working with cross account CloudTrails

If your CloudTrail log is in a separate account than where the ADA stack is deployed, you will need to provide the **Cross Account Role Arn** when you enter the Cross account access Role Arn. You will need to set up an IAM role in the TARGET_ACCOUNT to allow processes to read the CloudTrail log.

If your CloudTrail table will be imported from a separate account than where the ADA stack is deployed, you will need to provide the **Cross Account Role Arn** when you enter the Dynamo Table Arn. Use this script to set up the Cross Account IAM role in the account where you have the DynamoDB table.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
    "Effect": "Allow",
    "Principal": {
    "AWS": "<AWS account number that Ada solution deployed>"
    },
  "Action": []"sts:AssumeRole", "sts:TagSession"]
}
]
}
```

Set the identity policy using this script.

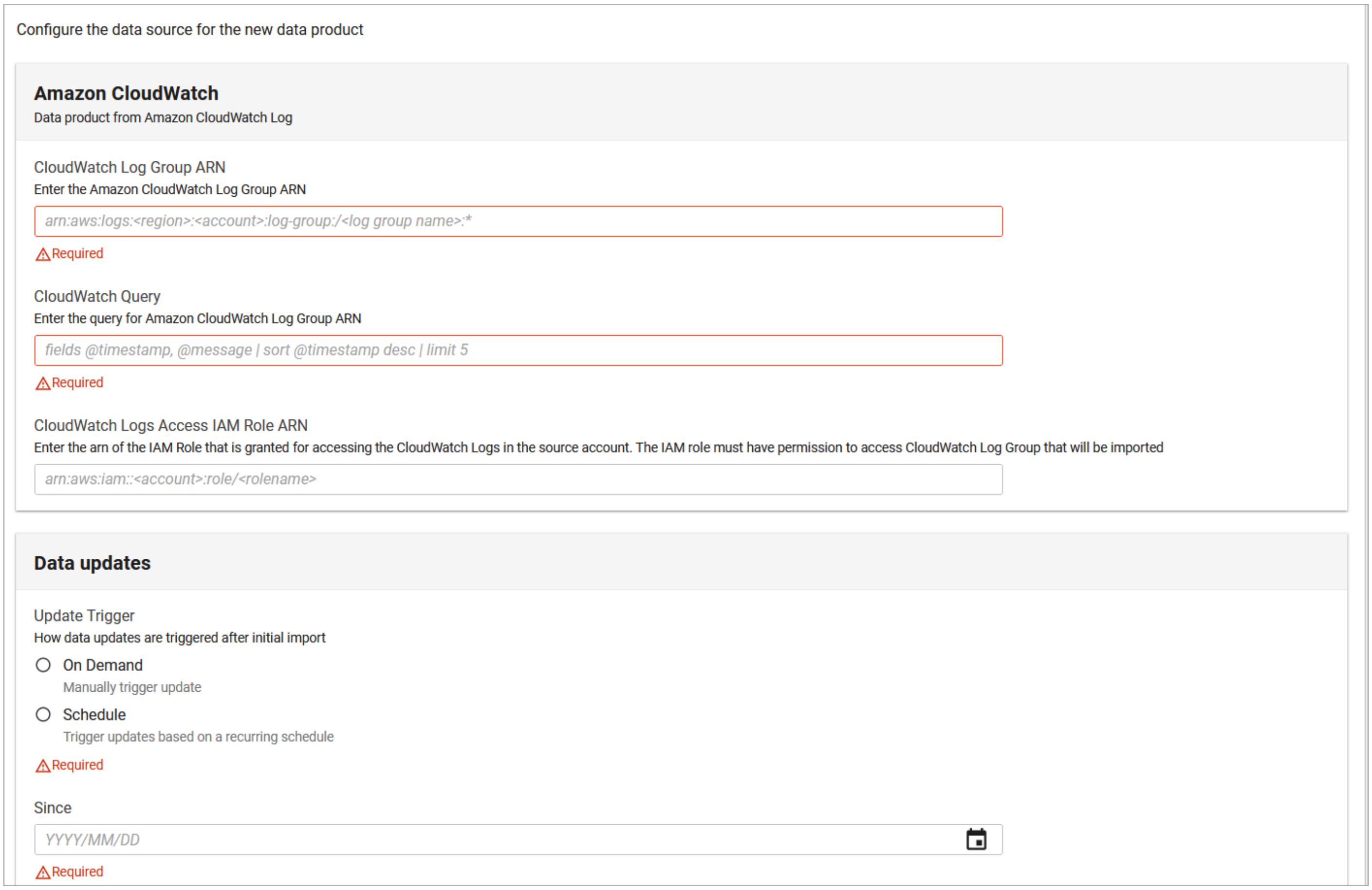
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
      "cloudtrail:Get",
      "s3:Get",
      "s3:ListBucket",
      "ec2:describeRegions"
    ],
  "Resource": [
    "<ARN of the CloudTrail Trail to be imported>",
    "<ARN of the S3 bucket that store the CloudTrail Data>",
    "<ARN of the S3 bucket that store the CloudTrail Data>/*"
    ]
  }
  ]
}
```

Amazon CloudWatch

The Amazon CloudWatch data connector allows data ingestion from CloudWatch logs in the same AWS account in which ADA has been deployed, or an external AWS Account.

To import data from Amazon CloudWatch:

1. Enter the CloudWatch Log Group ARN, CloudWatch Query and the CloudWatch Logs Access IAM Role ARN (only required if the logs are in an external AWS account).



Configure the data source for the new data product

Amazon CloudWatch

Data product from Amazon CloudWatch Log

CloudWatch Log Group ARN

Enter the Amazon CloudWatch Log Group ARN

arn:aws:logs:<region>:<account>:log-group:/<log group name>:*

⚠Required

CloudWatch Query

Enter the query for Amazon CloudWatch Log Group ARN

fields @timestamp, @message | sort @timestamp desc | limit 5

⚠Required

CloudWatch Logs Access IAM Role ARN

Enter the arn of the IAM Role that is granted for accessing the CloudWatch Logs in the source account. The IAM role must have permission to access CloudWatch Log Group that will be imported

arn:aws:iam::<account>:role/<rolename>

Data updates

Update Trigger

How data updates are triggered after initial import

○ On Demand

Manually trigger update

○ Schedule

Trigger updates based on a recurring schedule

⚠Required

Since

YYYY/MM/DD

⚠Required

Amazon CloudWatch connector

2. Select the appropriate data trigger for your data source, and select **Next**.

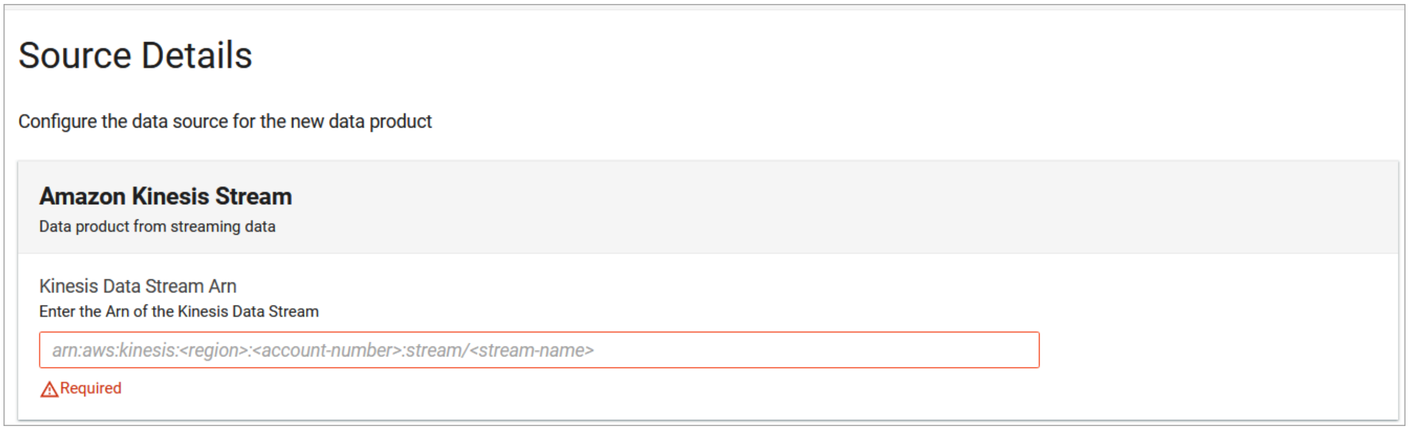
Note

If the logs are in an external AWS account, you will need to provide an IAM Role that exists in the external account, has access to read the CloudWatch Logs, and has a trust relationship, defining the account ADA is deployed in as a Principal, with access to `sts:AssumeRole`, and `sts:TagSession`. For more information, refer to the Controlling access to and for IAM roles page.

Amazon Kinesis Stream

To configure an Amazon Kinesis Stream:

1. Specify the Arn source location and provide access.



Source Details

Configure the data source for the new data product

Amazon Kinesis Stream

Data product from streaming data

Kinesis Data Stream Arn

Enter the Arn of the Kinesis Data Stream

arn:aws:kinesis:<region>:<account-number>:stream/<stream-name>

⚠Required

Amazon Kinesis Stream connector

2. Select the appropriate data trigger for your data source, and select **Next**.

You will need to grant access to the bucket from the associated AWS account. Access to the bucket can be granted through IAM principals. For more information on how to do this, refer to the IAM documentation.

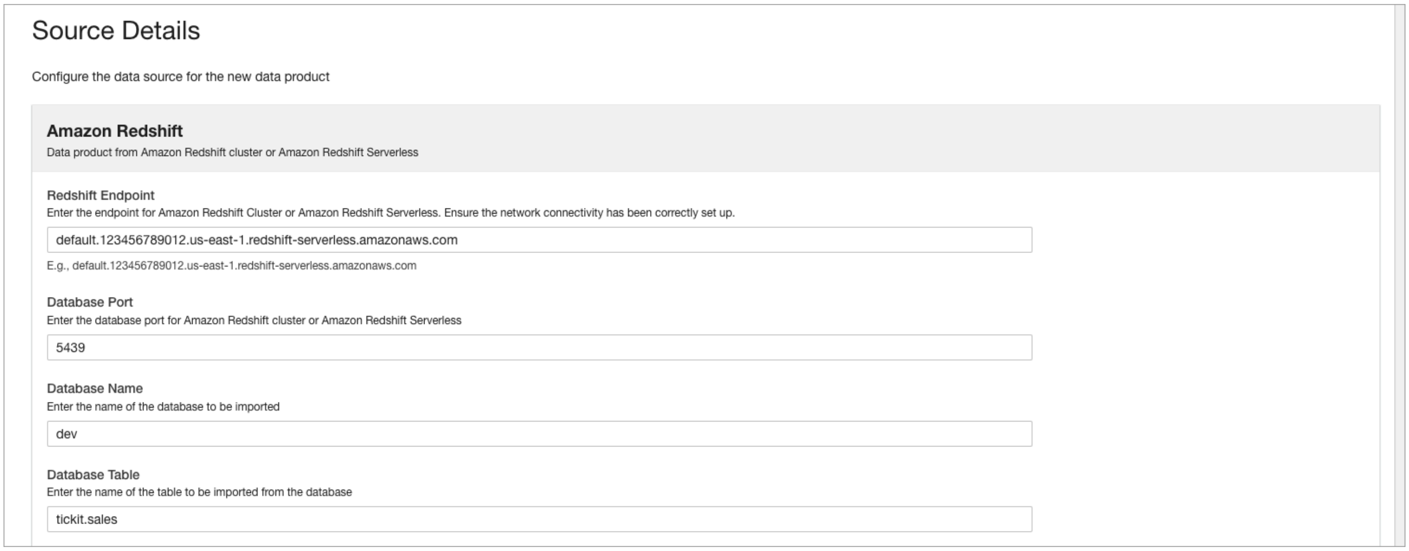
For more information on where to find your Kinesis Stream ARN, refer to the Controlling access documentation.

Amazon Redshift

The Amazon Redshift data connector allows data ingestion from Amazon Redshift cluster or Amazon Redshift serverless table in an AWS account.

To configure a Amazon Redshift connector:

1. On the **Source Details** page, under **Amazon Redshift**, enter the database endpoint, database port, database name, and the database table to be imported for the Amazon Redshift cluster or Redshift serverless instance. If you are using a VPC endpoint, enter the VPC endpoint as the database endpoint.



Source Details

Configure the data source for the new data product

Amazon Redshift

Data product from Amazon Redshift cluster or Amazon Redshift Serverless

Redshift Endpoint

Enter the endpoint for Amazon Redshift Cluster or Amazon Redshift Serverless. Ensure the network connectivity has been correctly set up.

default.123456789012.us-east-1.redshift-serverless.amazonaws.com

E.g., default.123456789012.us-east-1.redshift-serverless.amazonaws.com

Database Port

Enter the database port for Amazon Redshift cluster or Amazon Redshift Serverless

5439

Database Name

Enter the name of the database to be imported

dev

Database Table

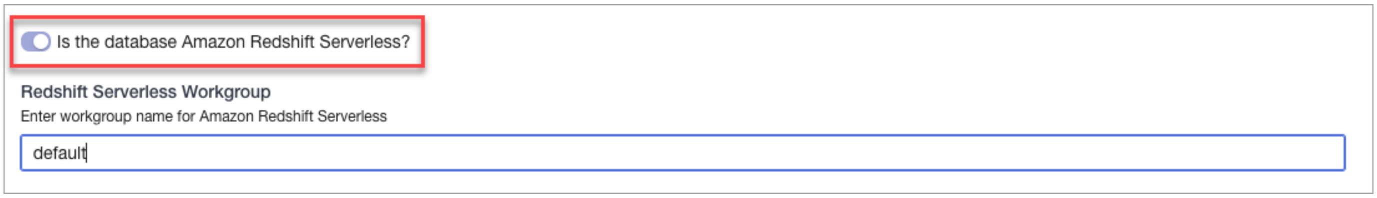
Enter the name of the table to be imported from the database

tickit.sales

Amazon Redshift connector

2. Depending on the type of instance, enter the following information:

   - For Redshift serverless instance, select the **Is the database Amazon Redshift Serverless** toggle, and enter the workgroup.



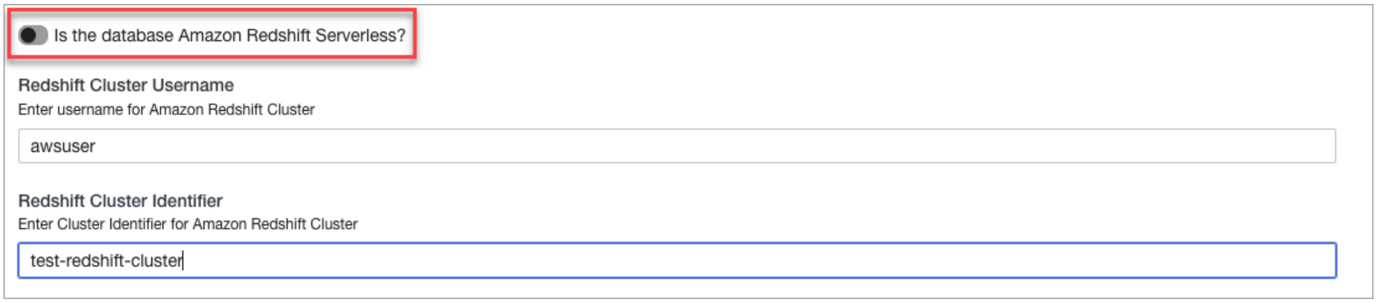
Is the database Amazon Redshift Serverless?

Redshift Serverless Workgroup

Enter workgroup name for Amazon Redshift Serverless

default

   - For Redshift cluster, toggle the **Is the database Amazon Redshift Serverless** option and enter the Redshift cluster username and cluster identifier.



Is the database Amazon Redshift Serverless?

Redshift Cluster Username

Enter username for Amazon Redshift Cluster

awsuser

Redshift Cluster Identifier

Enter Cluster Identifier for Amazon Redshift Cluster

test-redshift-cluster

> **Note**
>
> If your Redshift instance is from a different AWS account other than where ADA solution is deployed, you must provide a Cross Account Role ARN. You will need to provision this IAM role in the account where the Redshift instance is located and has the correct

> permissions to allow ADA to access it. For more information, refer to the Working with cross account CloudTrails section.

3. Select the appropriate data update trigger for your data source, and select **Next**.
4. Select **Continue your Current Schema**, and then **Submit**. This will start the data import workflow, and make the data available for querying.

Working with cross account CloudTrails

If your Redshift instance is hosted from a separate account than where the ADA stack is deployed, you must provide the Cross Account Role ARN. Use this template to set up the Cross Account IAM role in the account where you have the Redshift instance.

Set up trusted entities:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
    "Effect": "Allow",
    "Principal": {
    "AWS": "<AWS account number that Ada solution deployed>"
    },
  "Action": []"sts:AssumeRole", "sts:TagSession"]
}
]
}
```

Set up permissions policies for the Redshift Serverless workgroup:

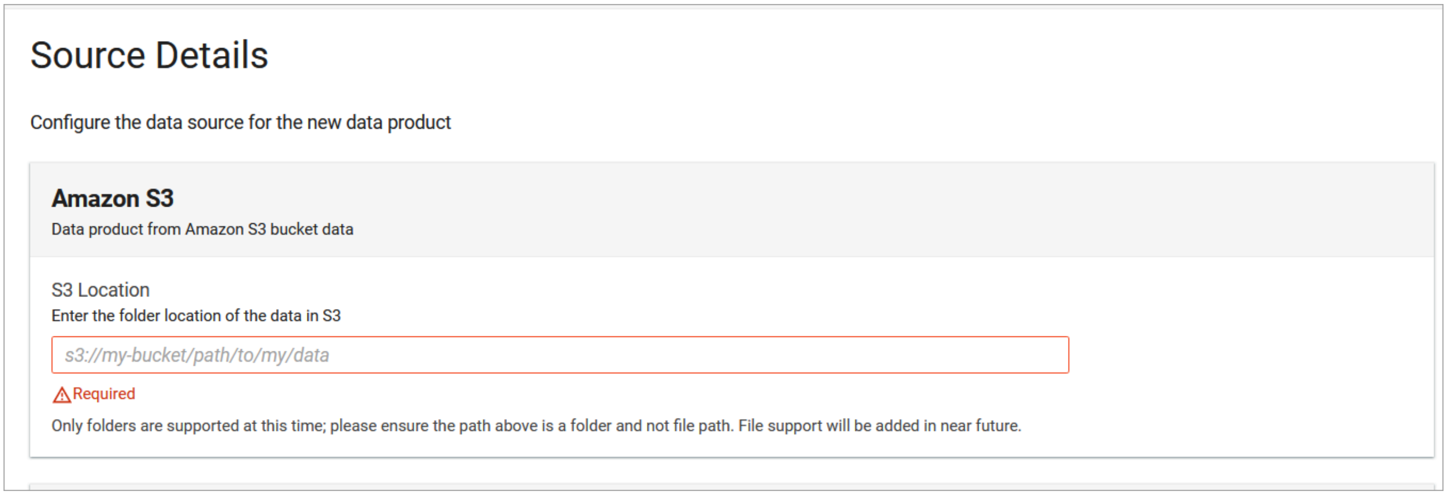
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "redshift-serverless:GetCredentials",
            "Resource": "<Your redshift serverless workgroup ARN>"
        }
    ]
}
```

Set up permissions policies for the Redshift Cluster:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "redshift:GetClusterCredentials",
            "Resource": "arn:aws:redshift:<region>:<account-id>:dbuser:<cluster-
identifier>/<dbuser-name>"
        }
    ]
}
```

Amazon S3

To configure an Amazon S3 bucket, provide the URL to the bucket location in your AWS account.



Source Details

Configure the data source for the new data product

Amazon S3

Data product from Amazon S3 bucket data

S3 Location

Enter the folder location of the data in S3

s3://my-bucket/path/to/my/data

⚠Required

Only folders are supported at this time; please ensure the path above is a folder and not file path. File support will be added in near future.

Amazon S3 connector

To learn more on where to find the URL, refer to the S3 documentation. You will need to grant access to the bucket from the associated AWS account. Access to the bucket can be granted through IAM principals. For more information on how to do this, refer to the IAM documentation.

Example Policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Grant Automated Data Analytics on AWS User access",
            "Effect": "Allow",
            "Action": "s3:Get*",
            "Resource": "arn:aws:s3:::<bucket-name>/*",
            "Principal": {
                "AWS": ["arn:aws:iam::<manifold-account>:root"]
            },
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalTag/manifold:user": "<manifold-user>"
                }
            }
        },
        {
            "Sid": "Grant Automated Data Analytics on AWS Group access",
            "Effect": "Allow",
            "Action": "s3:Get*",
            "Resource": "arn:aws:s3:::<bucket-name>/*",
            "Principal": {
                "AWS": ["arn:aws:iam::<manifold-account>:root"]
            },
            "Condition": {
                "StringLike": {
                    "aws:PrincipalTag/manifold:groups": "*:power-user:*"
                }
            }
        },
        {
            "Sid": "Grant Automated Data Analytics on AWS Group access in one of
group",
            "Effect": "Allow",
            "Action": "s3:Get*",
            "Resource": "arn:aws:s3:::<bucket-name>/*",
            "Principal": {
                "AWS": ["arn:aws:iam::<manifold-account>:root"]
            },
            "Condition": {
                "ForAnyValue:StringLike": {
                    "aws:PrincipalTag/manifold:groups": [
                        "*:power-user:*",
                        "*:some-custom-group:*"
                    ]
                }
            }
        },
        {
            "Sid": "Grant Automated Data Analytics on AWS Federated Query access",
            "Effect": "Allow",
            "Action": "s3:Get*",
            "Resource": "arn:aws:s3:::<bucket-name>/*",
            "Principal": {
                "AWS": ["arn:aws:iam::<manifold-account>:root"]
            },
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalTag/manifold:service": "query"
                }
            }
        },
        {
            "Sid": "Grant access from any Automated Data Analytics on AWS
 microservice",
            "Effect": "Allow",
            "Action": "s3:Get*",
            "Resource": "arn:aws:s3:::<bucket-name>/*",
            "Principal": {
                "AWS": ["arn:aws:iam::<manifold-account>:root"]
            },
            "Condition": {
                "StringEquals": {
                    "aws:PrincipalTag/manifold:service": "*"
                }
            }
        }
    ]
}
```

Automated Data Analytics on AWS supports folder level path configuration only, and does not support mixed file type in the same path.

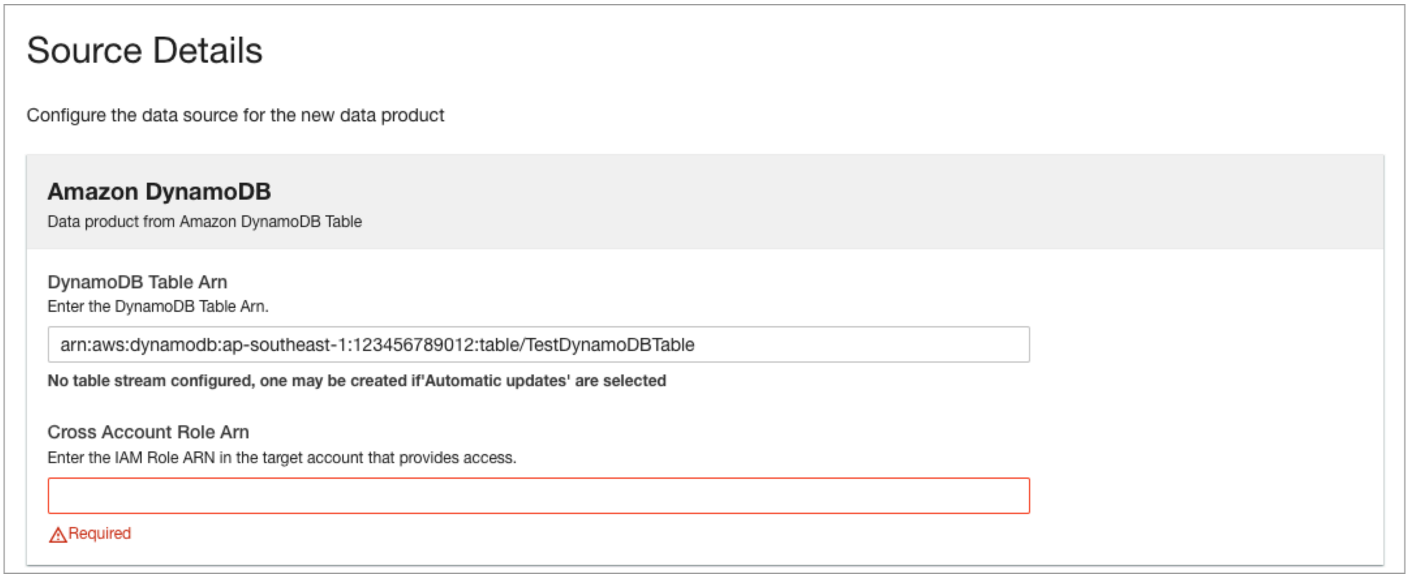
To enable ingest on mixed file types, it is recommended to keep those files in separate folder/path, and configure the S3 path to the parent folder.

DynamoDB

The DynamoDB data connector allows you to ingest data from an existing DynamoDB table in an AWS account. If the table is from a different AWS account other than where ADA solution is deployed, you must provide a Cross Account Role Arn. This IAM role needs to be provisioned in the account where DynamoDB table is located and have proper permissions to allow ADA to access it. For more information, refer to the Working with cross accounts section.

To import data from DynamoDB:

1. On the details page, under Amazon DynamoDB, enter the DynamoDB table stream arn. ADA will automatically validate the Arn via an API call to ensure it can retrieve data from the database table. If the table is from a different AWS account, it will display an additional field to request Cross Account Role.



Source Details

Configure the data source for the new data product

Amazon DynamoDB

Data product from Amazon DynamoDB Table

DynamoDB Table Arn

Enter the DynamoDB Table Arn.

arn:aws:dynamodb:ap-southeast-1:123456789012:table/TestDynamoDBTable

No table stream configured, one may be created if'Automatic updates' are selected

Cross Account Role Arn

Enter the IAM Role ARN in the target account that provides access.

⚠Required

Microsoft SQL Server connector

2. Under the Data updates section, select **On Demand**, and select **Next**. The Schema page displays.
3. (Optional) To transform your schema, or select **Continue with Current Schema**, and **Submit.** This will start the workflow to import the data, making it available to be queried.

Working with cross accounts

If your DynamoDB table will be imported from a separate account than where the ADA stack is deployed, you will need to provide the **Cross Account Role Arn** when you enter the Dynamo

Table Arn. Use this script to set up the Cross Account IAM role in the account where you have the DynamoDB table.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
    "Effect": "Allow",
    "Principal": {
    "AWS": "<AWS account number that Ada solution deployed>"
    },
  "Action": []"sts:AssumeRole", "sts:TagSession"]
}
]
}
```

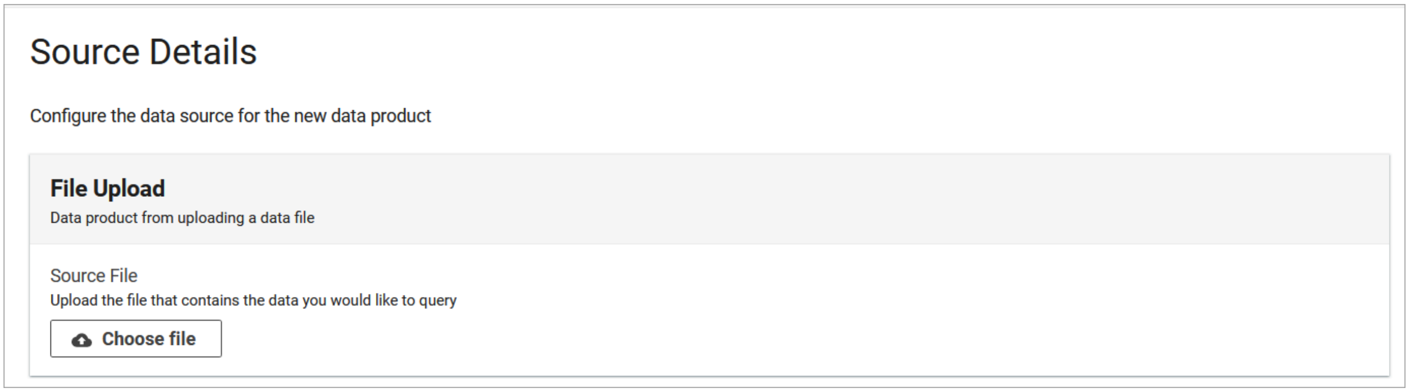
Set the identity policy using this script.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
    "Effect": "Allow",
    "Action": [
        "dynamodb:GetItem",
        "dynamodb:BatchGetItem",
        "dynamodb:Scan",
        "dynamodb:Query",
        "dynamodb:ConditionCheckItem",
    ],
  "Resource": "<ARN of the DynamoDB table to be imported>"
  }
  ]
}
```

File Upload

To upload a file from your local device:

1. Select **Choose file**.



Source Details

Configure the data source for the new data product

File Upload

Data product from uploading a data file

Source File

Upload the file that contains the data you would like to query

File upload connector

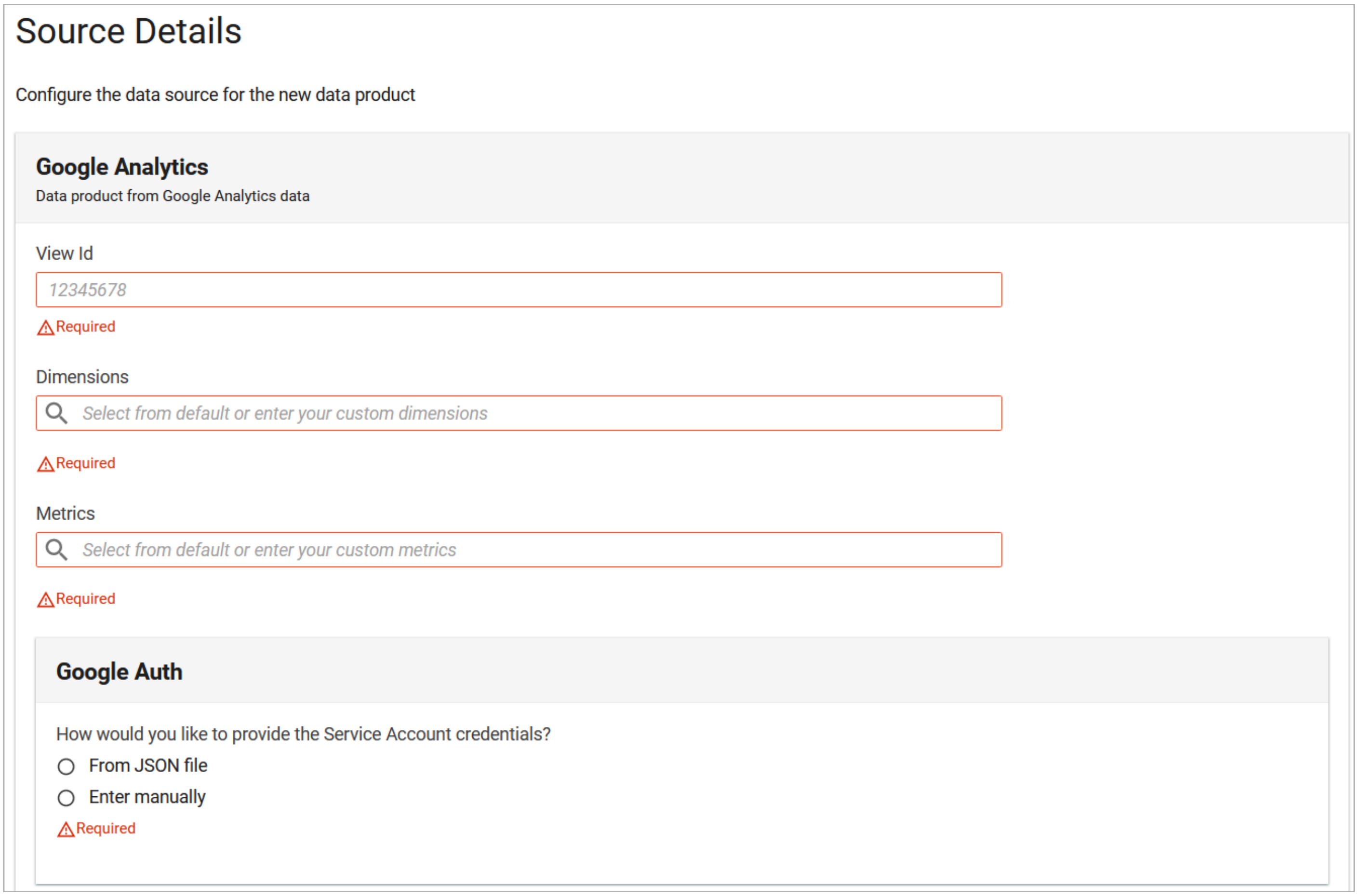
2. Choose the file you would like to upload, and select **Open.** File upload supports the .csv, .json, .parquet, and .gz file formats.
3. Select **Next** to proceed. The uploaded file is parsed and available for querying.

> **Note**
>
> The uploaded files are stored in a shared Amazon S3 buckets for all uploads and only accessible through the solution. Once a file is uploaded as source data, it is treated the same as Amazon S3 sourced data.

Google Analytics

To configure the Google Analytics connector:

1. Provide the View ID, dimensions and metrics. Custom dimensions and metrics are also supported through free entry field.



Source Details

Configure the data source for the new data product

Google Analytics

Data product from Google Analytics data

View Id

12345678

⚠Required

Dimensions

Select from default or enter your custom dimensions

⚠Required

Metrics

Select from default or enter your custom metrics

⚠Required

Google Auth

How would you like to provide the Service Account credentials?

○ From JSON file

○ Enter manually

⚠Required

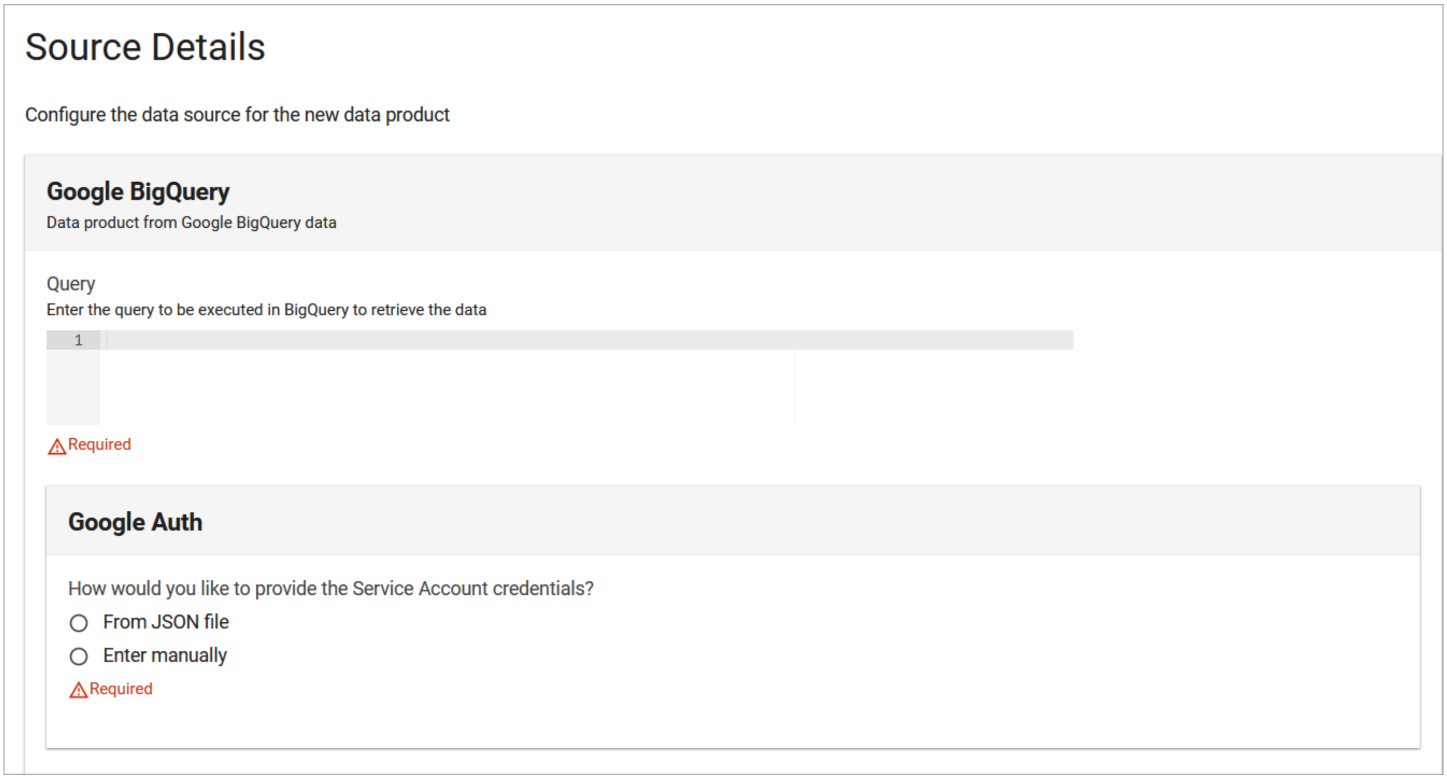
Google Analytics connector

2. Choose to provide access manually or by uploading Service Account JSON file similar to BigQuery.
3. Select the appropriate data trigger and policy for your data source, and select **Next.**

For more information, refer to Google Analytics' authorization documentation.

Google BigQuery

To configure a BigQuery table, select the location URL to provide access manually or by uploading a Service Account json file.



Source Details

Configure the data source for the new data product

Google BigQuery

Data product from Google BigQuery data

Query

Enter the query to be executed in BigQuery to retrieve the data

```
1
```

⚠Required

Google Auth

How would you like to provide the Service Account credentials?

○ From JSON file

○ Enter manually

⚠Required

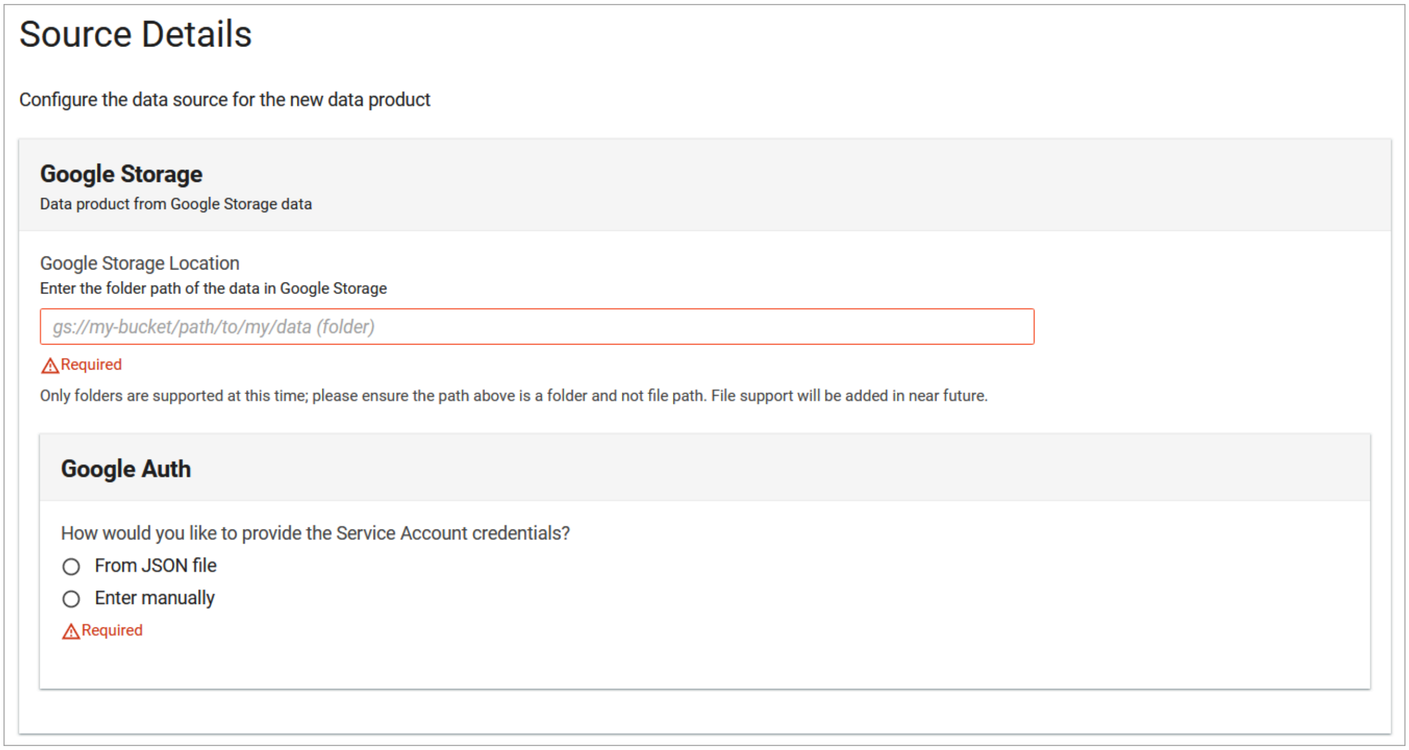
Google BigQuery connector

For more information on where to find URL’s and credentials, refer to Google BigQuery’s authentication documentation.

Google Storage

To configure a Google Cloud Storage bucket,

1. Enter the bucket location URL and method of providing credentials. You can provide credentials manually or by uploading a Service Account json file.



Source Details

Configure the data source for the new data product

Google Storage

Data product from Google Storage data

Google Storage Location

Enter the folder path of the data in Google Storage

gs://my-bucket/path/to/my/data (folder)

⚠Required

Only folders are supported at this time; please ensure the path above is a folder and not file path. File support will be added in near future.

Google Auth

How would you like to provide the Service Account credentials?

○ From JSON file

○ Enter manually

⚠Required

Google Storage connector

2. Select the appropriate data trigger for your data source, and select **Next**.

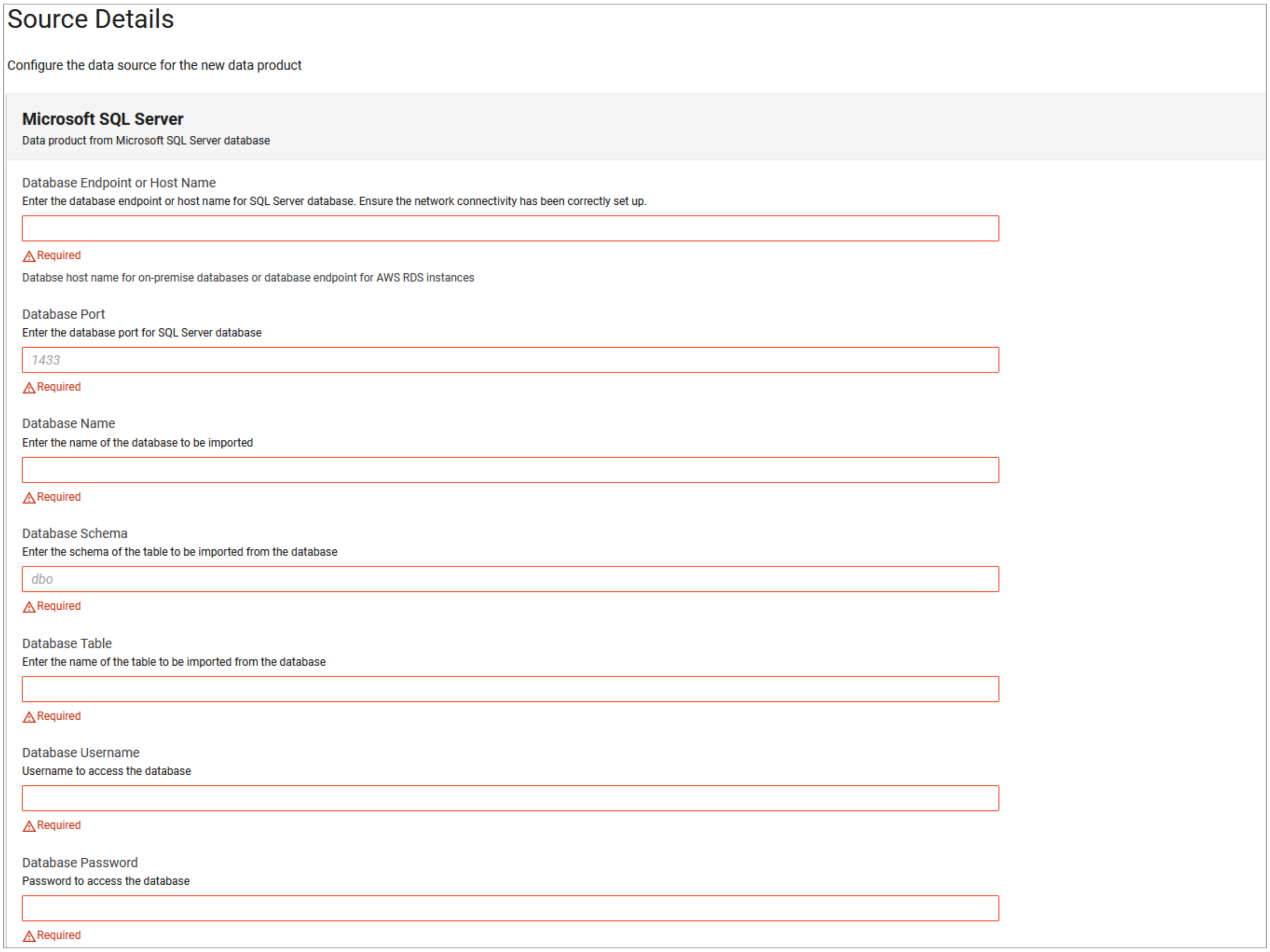
For more information on where to find URL's and credentials, refer to Google Cloud Storage's documentation for access controls and authentication.

> **Note**
>
> The current implementation of Google Storage connector uses a sync between the source bucket (Google Storage) and the target bucket (Amazon S3) and works only for folders. This will remove files from the target bucket if they are removed from the source bucket.

Microsoft SQL Server

To import data from a Microsoft SQL Server database,

1. Enter the database endpoint or host name, port number, database name, schema name, table name and the database user credentials that have been granted read-only permission to access the source database and tables.



Source Details

Configure the data source for the new data product

Microsoft SQL Server

Data product from Microsoft SQL Server database

Database Endpoint or Host Name

Enter the database endpoint or host name for SQL Server database. Ensure the network connectivity has been correctly set up.

⚠Required

Databse host name for on-premise databases or database endpoint for AWS RDS instances

Database Port

Enter the database port for SQL Server database

1433

⚠Required

Database Name

Enter the name of the database to be imported

⚠Required

Database Schema

Enter the schema of the table to be imported from the database

dbo

⚠Required

Database Table

Enter the name of the table to be imported from the database

⚠Required

Database Username

Username to access the database

⚠Required

Database Password

Password to access the database

⚠Required

Microsoft SQL Server connector

2. Select the appropriate data trigger for your data source, and select **Next.**

MongoDB

The MongoDB data connector allows you to ingest data from MongoDB and DocumentDB instances, provided you have set up ADA to access data sources within the VPC or on-premises network. For more information, refer to the Set up ADA to access data sources within the VPC or on-premises network topic.

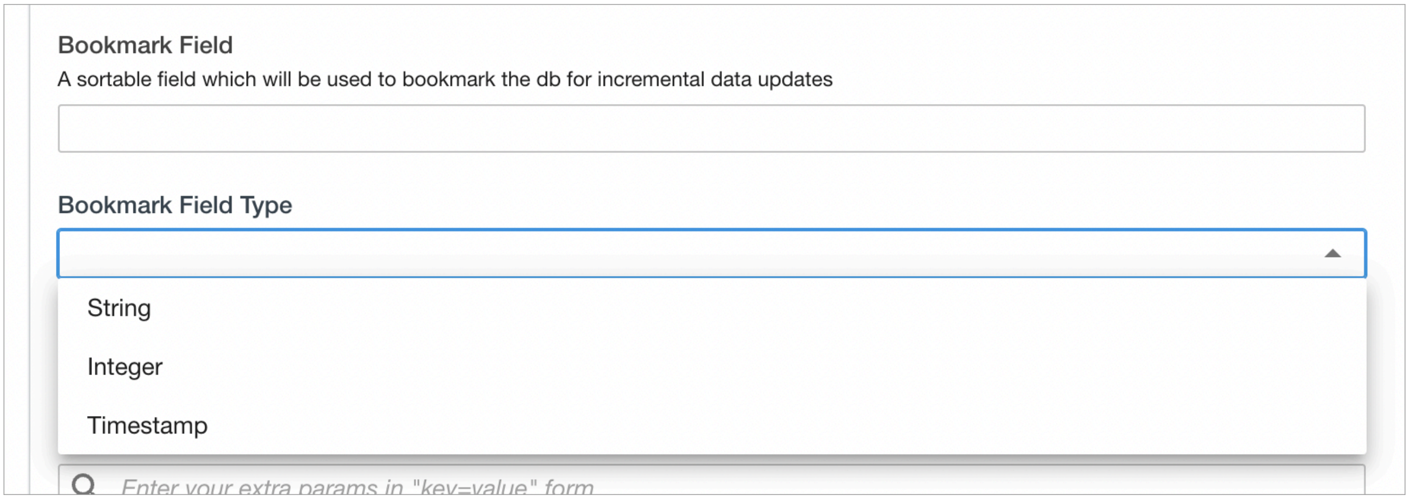
To import data from MongoDB:

1. On the source details page, for the mandatory database details, enter the following information:

- **Database Endpoint or Host Name**: Enter the database endpoint or host name. You will need to ensure you have set up ADA to access data sources within the VPC or on-premises network.
- **Database Port**: Enter the port number for the database.
- **Database Name**: Enter the name of the database you want to import.
- **Database Collection**: Enter the name of the collection you want to import.
- **Database Username**: Provide the username required to access the database.
- **Database Password**: Type in the password required to access the database. By default, the password will be hidden/masked.

(Optional) To allow incremental data updates on a MongoDB data source, provide these following details:

2. **Bookmark Field**: Enter a sortable field from the database to be used as a bookmark.
3. **Bookmark Field Type**: Select the data type for the bookmark field from the dropdown. You can choose from String, Integer or Timestamp.



Bookmark Field

A sortable field which will be used to bookmark the db for incremental data updates

Bookmark Field Type

String

Integer

Timestamp

Enter your extra params in "key=value" form

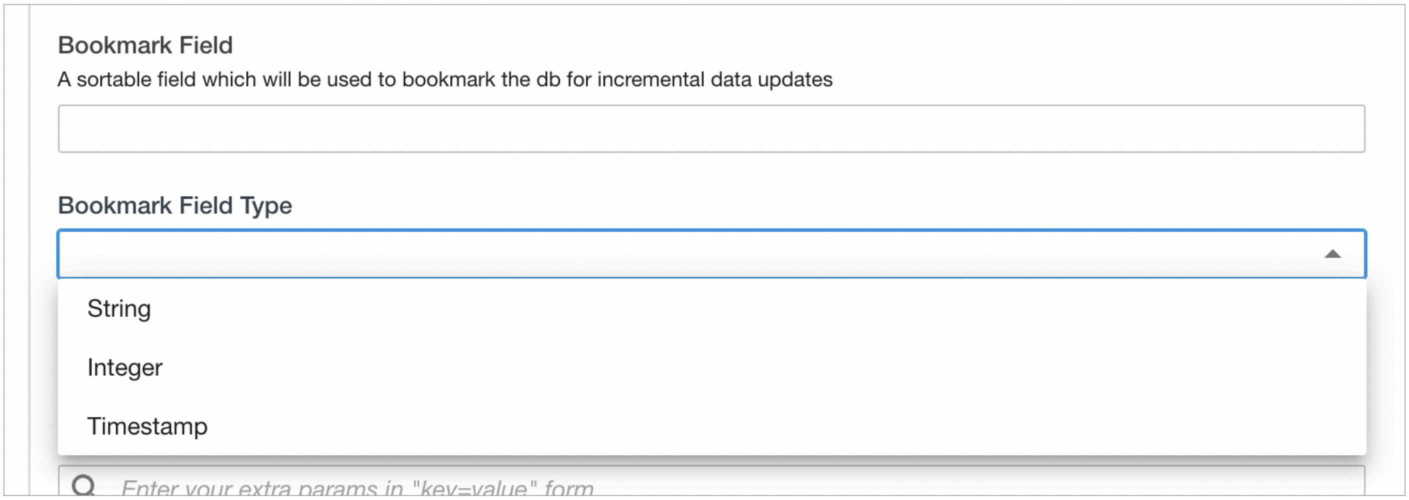
Bookmark field type

> **Note**
>
> If the Bookmark Field Type value in the database does not align with the data type entered in ADA, it will result in an error and the data product will stop building.

Using the bookmark fields

Once you set a bookmark field during the initial import of data, all the data in the database and collection are imported and the max value that exists in the set Bookmark Field is stored by ADA. On subsequent data updates, ADA will only query the data for database elements where the value in the Bookmark Field is greater than the value stored in ADA from the previous data update.

1. To communicate with MongoDB databases over TLS/SSL, tick the TLS box to enter the following information:



Bookmark Field

A sortable field which will be used to bookmark the db for incremental data updates

Bookmark Field Type

MongoDB bookmark

2. **URI to fetch CA file**: Enter the path to download the CA file. For example, the Amazon S3 bucket or a http datasource.
3. **Client Certificate file**: Copy and paste the client certificate file in the field. This file is stored in AWS Secrets Manager.

Using PyMongo

If you want to configure ADA through PyMongo to connect using a specific set of CA certificates, you can supply your own CA file. To do this, in the **URI to fetch CA file** field, enter the path to the location where your file is located (Amazon S3 buckets or http sources).

You can also configure ADA to present a client certificate. If the private key for a client certificate is stored in a separate file, concatenate it with the certificate file and copy and paste the file in the **Client Certificate file** field.

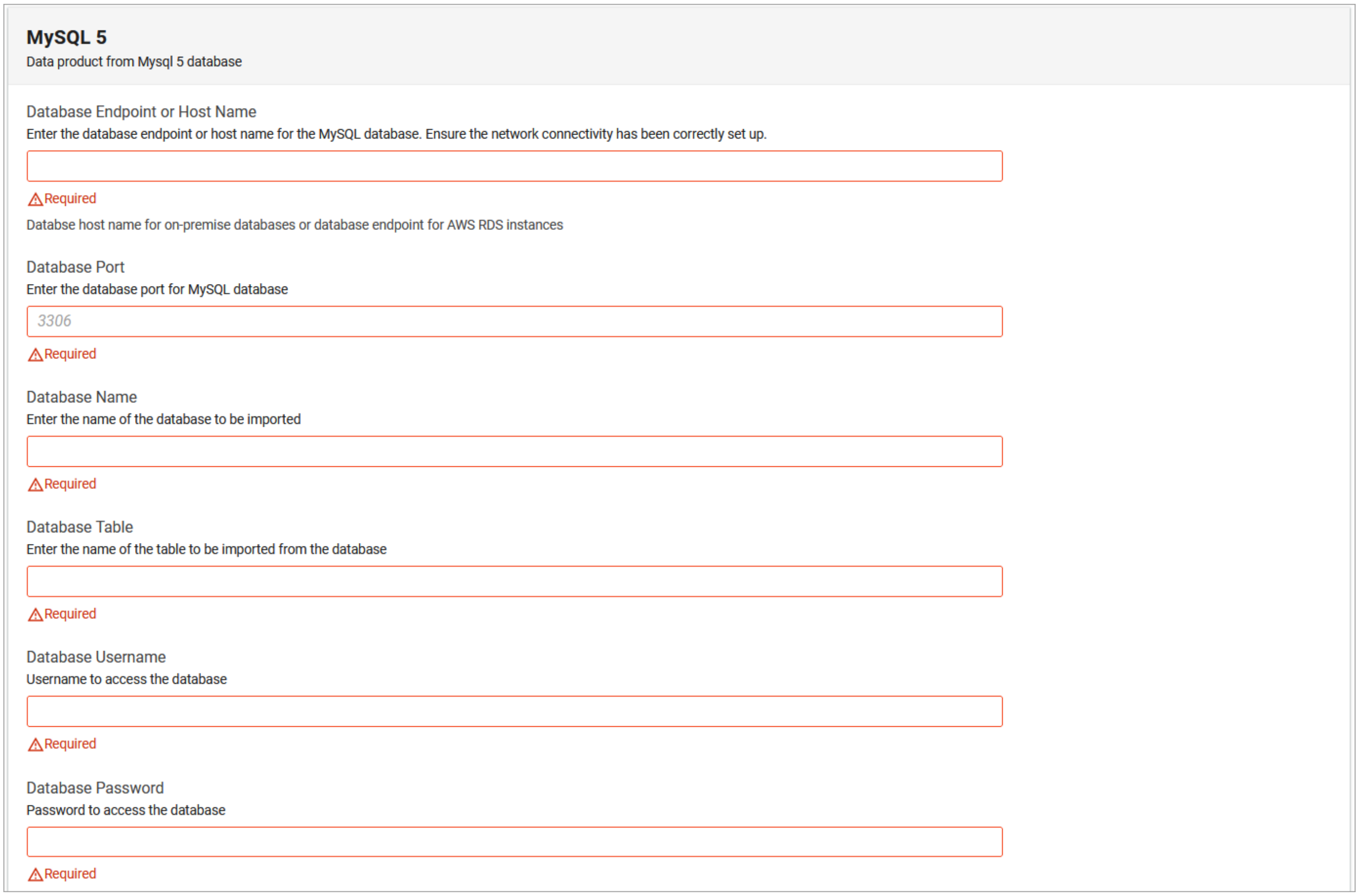
For more information, refer to TLS/SSL and PyMongo documentation.

1. In the **Extra Parameters** field, enter any parameter key values for the connection string. For example, if you have ticked the TLS option, and want to pass an extra parameter such as tlsAllowInvalidCertifications=true, enter it in this field. When ADA builds the MongoDB connection string, it will append it at the end of the URI string: `mongodb://example.com/?tls=true&tlsAllowInvalidCertificates=true`
2. Click **Next** to continue. The **Data updates** page displays.
3. On the **Data updates** page, choose a **Update Trigger** strategy. MongoDB support two data update strategies.
   - **On Demand**: You will need to manually trigger the data updates on demand.
   - **Schedule**: You can set the interval for automatic data updates. You can choose from hourly, daily, weekly, monthly, or a custom schedule.
4. Click **Next** to continue.

MySQL5.x

To import data from a MySQL database:

1. Enter the database endpoint or host name, port number, database name, table name and the database user credentials that have been granted read-only permission to access the source database and tables.



MySQL 5

Data product from Mysql 5 database

Database Endpoint or Host Name

Enter the database endpoint or host name for the MySQL database. Ensure the network connectivity has been correctly set up.

⚠Required

Databse host name for on-premise databases or database endpoint for AWS RDS instances

Database Port

Enter the database port for MySQL database

3306

⚠Required

Database Name

Enter the name of the database to be imported

⚠Required

Database Table

Enter the name of the table to be imported from the database

⚠Required

Database Username

Username to access the database

⚠Required

Database Password

Password to access the database

⚠Required

MySQL connector

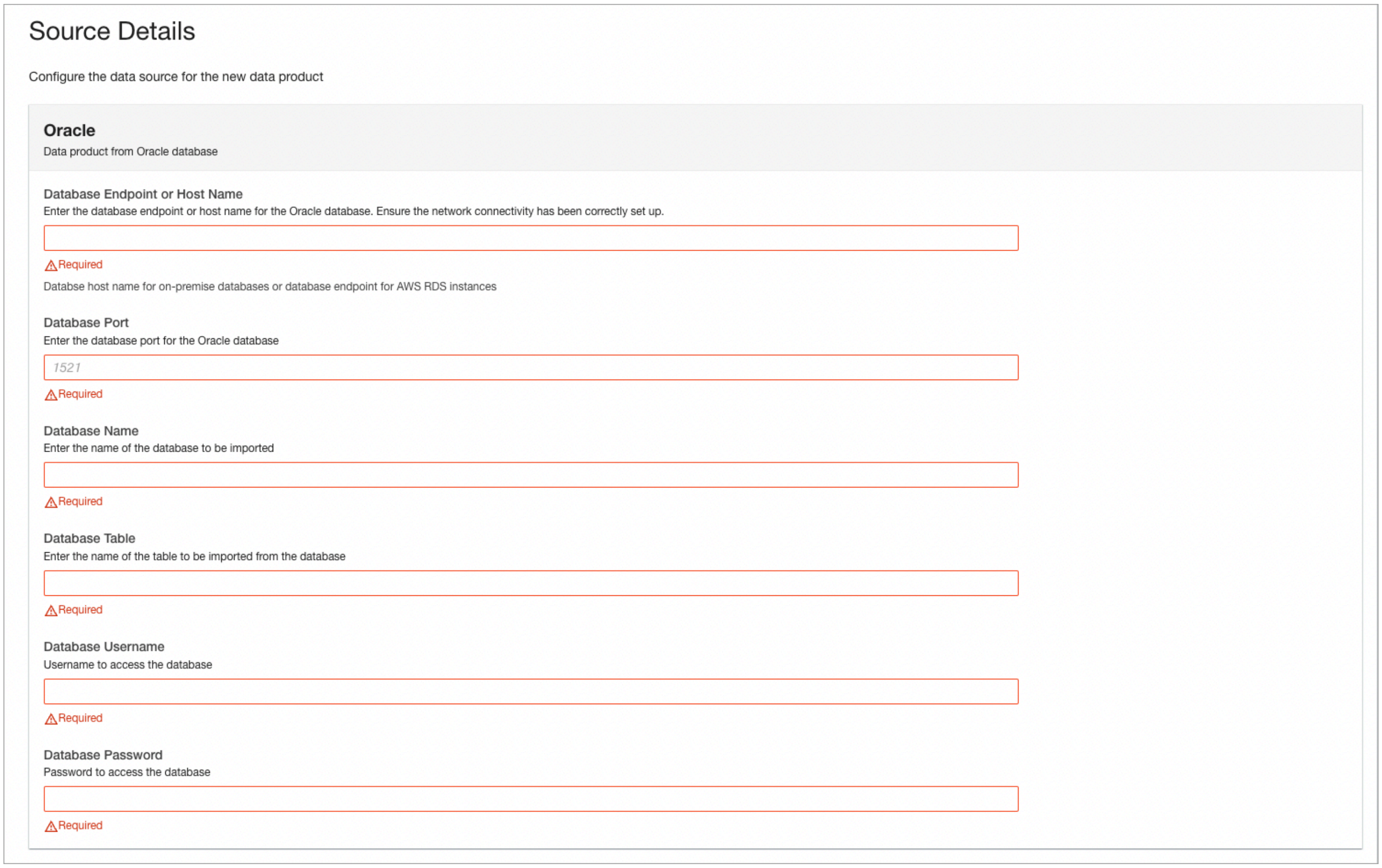
2. Select the appropriate data trigger for your data source, and select **Next**.

> **Note**
>
> The ADA MySQL connector currently only supports MySQL 5.x

Oracle database

To import data from an Oracle database:

1. Enter the database endpoint or host name, port number, database name, table name and the database user credentials that have been granted read-only permission to access the source database and tables.



Source Details

Configure the data source for the new data product

Oracle

Data product from Oracle database

Database Endpoint or Host Name

Enter the database endpoint or host name for the Oracle database. Ensure the network connectivity has been correctly set up.

⚠Required

Databse host name for on-premise databases or database endpoint for AWS RDS instances

Database Port

Enter the database port for the Oracle database

1521

⚠Required

Database Name

Enter the name of the database to be imported

⚠Required

Database Table

Enter the name of the table to be imported from the database

⚠Required

Database Username

Username to access the database

⚠Required

Database Password

Password to access the database

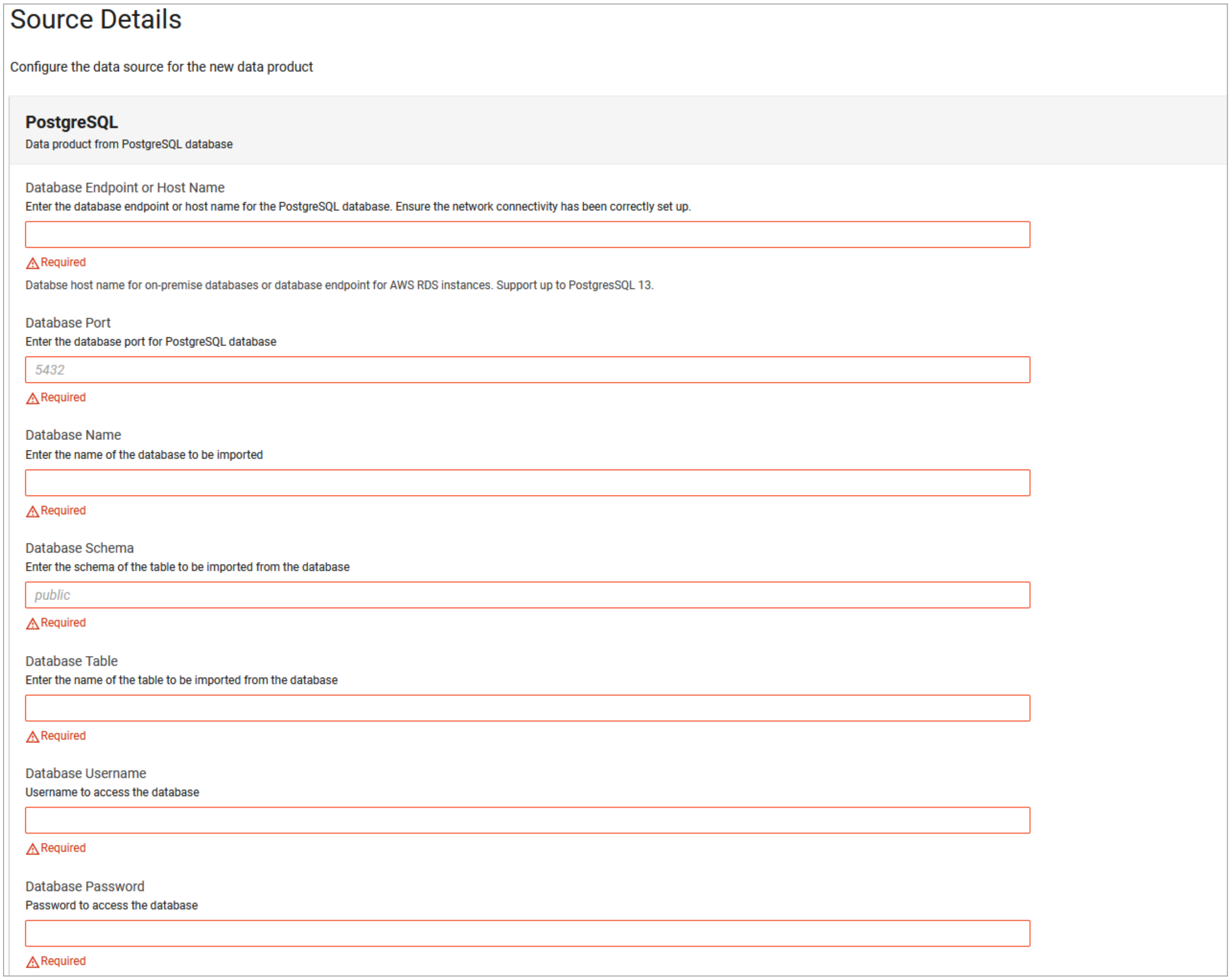
⚠Required

2. Select the appropriate data trigger for your data source, and select **Next**.

PostgreSQL13.x

To import data from PostgreSQL database:

1. Enter the database endpoint or host name, port number, database name, schema name, table name and the database user credentials that have been granted read-only permission to access the source database and table.



Source Details

Configure the data source for the new data product

PostgreSQL

Data product from PostgreSQL database

Database Endpoint or Host Name

Enter the database endpoint or host name for the PostgreSQL database. Ensure the network connectivity has been correctly set up.

⚠Required

Databse host name for on-premise databases or database endpoint for AWS RDS instances. Support up to PostgresSQL 13.

Database Port

Enter the database port for PostgreSQL database

5432

⚠Required

Database Name

Enter the name of the database to be imported

⚠Required

Database Schema

Enter the schema of the table to be imported from the database

public

⚠Required

Database Table

Enter the name of the table to be imported from the database

⚠Required

Database Username

Username to access the database

⚠Required

Database Password

Password to access the database

⚠Required

PostgreSQL connector

2. Select the appropriate data trigger for your data source, and select **Next.**

> **Note**
>
> The ADA PostgreSQL connector currently only supports PostgreSQL 13.x

SQL Database Connectors

ADA can connect to databases through a Java Database Connectivity (JDBC) connection which requires network connectivity between ADA and the data source databases.

Note

Before you create any data products with any of the following database connectors, ensure that the ADA system administrator has properly configured network connectivity. For more information, refer to the Set up ADA to access data sources within the VPC or on-premise network section.

Extensions guide

You can extend the Automated Data Analytics on AWS solution's capabilities by integrating with the following:

- Apache Superset for visualizing and analyzing data ingested into ADA, and
- Amazon AppFlow to securely transfer data between Software as a service (SaaS) applications and AWS services without code

Apache Superset

This extension allows you to deploy Apache Superset as data analytics and visualization platform for the Automated Data Analytics on AWS (ADA) solution. You can use Apache Superset to ingest data from ADA data products and help ADA users explore and visualize their data sets.

> **Note**
>
> Apache Superset is offered by an AWS Partner Solution **Apache Superset On AWS,** and is a standalone solution that runs alongside the ADA solution, serving as the visualization platform for ADA.

Connecting a data product in ADA

After you have you deployed Apache Superset in ADA, you can connect a data product in ADA to start visualizing data in Apache Superset.

To connect to a data product in ADA:

1. From the ADA user profile, create an API key for the ADA user who will access ADA data using Apache Superset.
2. Open Apache Superset and choose **Settings→ Database Connections**, and select **+Database.**
3. From the **Supported databases** drop-down list, choose **Other**, and enter the following URI in the SQLALCHEMY URI field.

```
awsathena+rest://api-key<Ada API Key>:1234@athena.<AWS
Region>.amazonaws.com:443?catalog_name=<ADA Data Product Domain
Identifier>&s3_staging_dir=s3%3A//
```

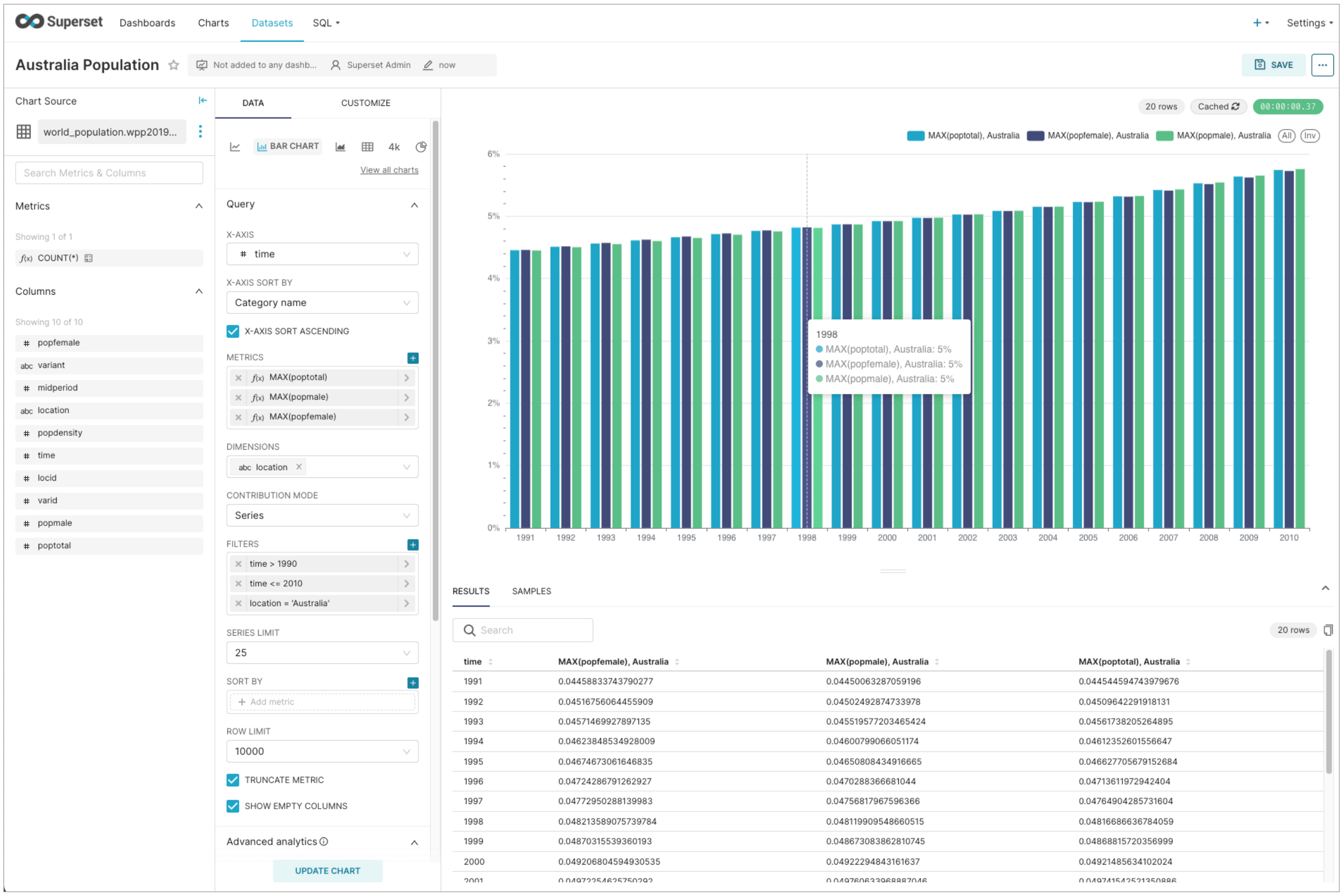
4. Edit the Display Name from Other to **ADA [Data Product Domain Name]**.
5. Choose **Test Connection** to make sure Apache Superset can connect to the ADA data product.
6. Once confirmed, choose **Connect** to create this database connection. You can now start using Apache Superset to analyze data from the ADA platform.

> **Note**
>
> Deploying Apache Superset requires user access to the AWS Management Console and an intermediate level of knowledge of AWS service administration. Make sure you have sufficient permissions and access rights before you start the deployment process.

Viewing the visualization in Apache Superset

Once you have deployed Apache Superset, and connected your data product to Apache Superset, you can view and use Apache Superset’s visualization platform for analyzing your data.

1. From the AWS CloudFormation console, navigate to the **superset** stack.
2. On the **Outputs** tab, under **SupersetConsole**, copy the URL and open it in a browser. This URL points to Apache Superset's web console. The default user name and password are below. You must change the admin password after the first login.
   - username: admin
   - password: admin
3. After you sign in, open the Datasets tab to view your visualizations. For example, a bar graph view of your data could be represented like this.



Australia Population ☆ Not added to any dashb... Superset Admin now

Chart Source

world_population.wpp2019...

Search Metrics & Columns

Metrics

Showing 1 of 1

- f(x) COUNT(*)

Columns

Showing 10 of 10

- # popfemale
- abc variant
- # midperiod
- abc location
- # popdensity
- # time
- # locid
- # varid
- # popmale
- # poptotal

Query

X-AXIS

- # time

X-AXIS SORT BY

Category name

☑ X-AXIS SORT ASCENDING

METRICS

- f(x) MAX(poptotal)
- f(x) MAX(popmale)
- f(x) MAX(popfemale)

DIMENSIONS

abc location

CONTRIBUTION MODE

Series

FILTERS

- time > 1990
- time <= 2010
- location = 'Australia'

SERIES LIMIT

25

SORT BY

+ Add metric

ROW LIMIT

10000

☑ TRUNCATE METRIC

☑ SHOW EMPTY COLUMNS

Advanced analytics

time	MAX(popfemale), Australia	MAX(popmale), Australia	MAX(poptotal), Australia
1991	0.04458833743790277	0.04450063287059196	0.044544594743979676
1992	0.04516756064455909	0.04502492874733978	0.04509642291918131
1993	0.04571469927897135	0.045519577203465424	0.04561738205264895
1994	0.04623848534928009	0.04600799066051174	0.04612352601556647
1995	0.04674673061646835	0.04650808434916665	0.046627705679152684
1996	0.04724286791262927	0.0470288366681044	0.04713611972942404
1997	0.04772950288139983	0.04756817967596366	0.04764904285731604
1998	0.048213589075739784	0.048119909548660515	0.04816686636784059
1999	0.04870315539360193	0.048673083862810745	0.04868815720356999
2000	0.049206804594930535	0.04922294843161637	0.04921485634102024
2001	0.04972254625750292	0.049760633968887046	0.049741542521350886

Apache Superset visualization

Amazon AppFlow

The AppFlow extension for ADA extends the existing out-of-the-box source data connectors from ADA to additional SaaS sources available through Amazon AppFlow.

The extension provides an AWS Cloud Development Kit (AWS CDK) stack for easy deployment of the relevant AWS services, facilitating the automated generation of data products based on your AppFlow flows.

How does it work

The extension automatically discovers your existing and new AppFlow flows with an S3 destination, and creates the corresponding ADA data products for you. This orchestration process is powered by Amazon EventBridge, AWS Lambda, and AWS Step Functions.

Whenever an AppFlow flow completes successfully, the extension automatically initiates the creation or update of the relevant ADA data product. Additionally, you can configure the extension to send you email notifications when data product updates are completed.

The extension is deployed in the same AWS account where you have set up your AppFlow flows, but it also works across different accounts if your ADA and AppFlow setups are in separate AWS accounts.

Using Amazon AppFlow

View our GitHub repository to access the Amazon AppFlow Extension for ADA and refer to the README.md file for additional details and deployment instructions.

Developer guide

Source code

Visit our GitHub repository to download the source files for this solution and to share your customizations with others.

The Automated Data Analytics on AWS templates are generated using the AWS Cloud Development Kit (AWS CDK) (AWS CDK). Refer to the README.md file for additional information.

CDK deployment

For configuring advanced settings or customizing the solution, it is recommended to download the source code from the GitHub source repository and build and deploy with AWS CDK. For more information, refer to the README file.

Access the ADA APIs

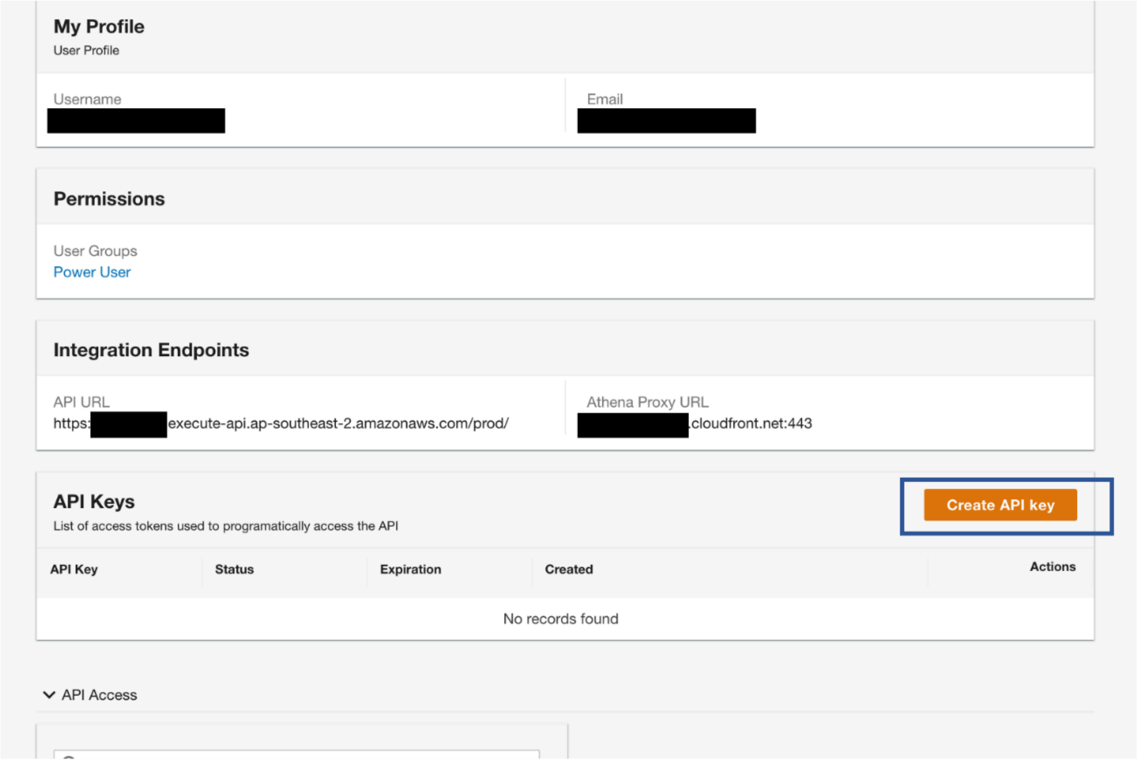
All features from the Automated Data Analytics (ADA) on AWS solution are also available as API endpoints.

You can build your own client application to interact with the ADA API endpoints to support your use cases.

Create an API key

To access the ADA APIs, you must create an API key under your user profile.

1. Under your Profile, select **Create API key**. The **Create API Key** dialog box is displayed.



My Profile

User Profile

Username
[redacted]

Email
[redacted]

Permissions

User Groups
Power User

Integration Endpoints

API URL
https:[redacted]execute-api.ap-southeast-2.amazonaws.com/prod/

Athena Proxy URL
[redacted].cloudfront.net:443

API Keys

List of access tokens used to programatically access the API

API Key	Status	Expiration	Created	Actions
No records found				

Create API key

2. On the **Create API Key** dialog, add a name, expiration date for the API Key, and select **Enable** to indicate the API key is active.
3. Select **Submit**. An API key is created for you.
4. Select **Click to clipboard** to copy the generated API key.

The generated API Key has the following format:

```
{
"name": "test_apikey",
"clientId": "<client_id>",
"clientSecret": "<client_secret>",
"authToken": "<auth_token>",
"authUrl": "<auth_url>"
}
```

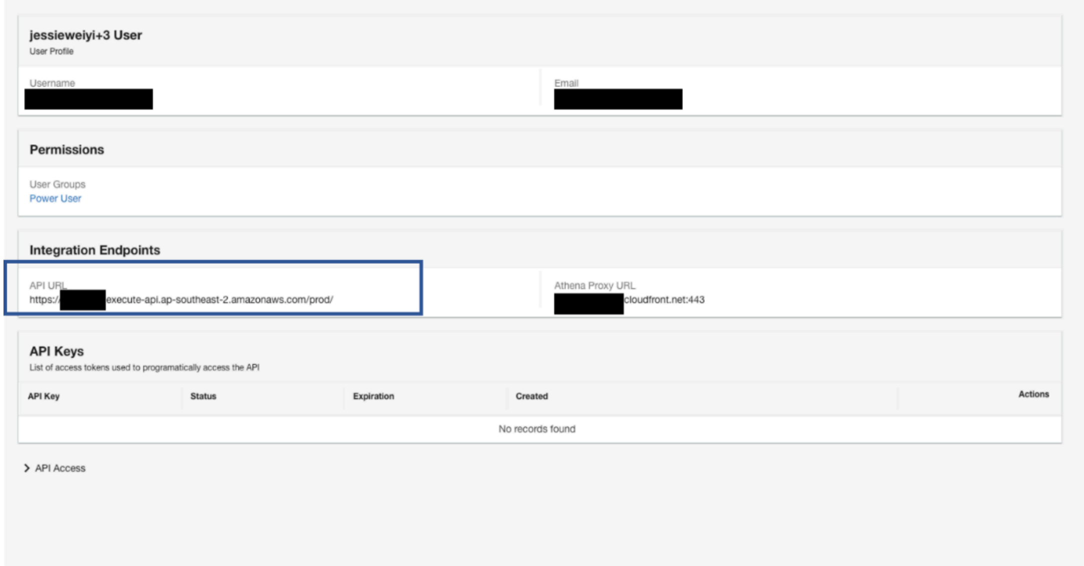
The value of `auth_Token` is a base 64 encoded ***clientId:clientSecret*** pair. You can use it as the API key for API requests from your customized app client. The created API key has the same permissions as the user who created them. We recommend saving this API key value credentials in a safe place, for example [AWS Secrets Manager](#) and retrieve the value from your application.

Under the hook, each API key maps to an App Client in Amazon Cognito. API keys are sent through the HTTP header from the original request and then exchanged for an access token via an oauth2 request using the client id and secret combination against the Amazon Cognito client app.

Currently, the maximum quota for an App Client for each Amazon Cognito user pool is 1,000 requests. This implies the limit of API keys that can be generated per Automated Data Analytics on AWS deployment is 1000, but you can increase this to 10000 requests. For most of the use cases in Automated Data Analytics on AWS, we foresee this to be sufficient.

Send requests to API endpoints

You can find the API URL under **Integration Endpoints** on your **Profile** page. You can send a HTTP request to the API endpoints using the authToken (noted as ***<api_key>*** in the code examples below). Here is an API request example using cURL.



jessieweiyi+3 User
User Profile

Username
[redacted]

Email
[redacted]

Permissions

User Groups
Power User

Integration Endpoints

API URL
https://[redacted]execute-api.ap-southeast-2.amazonaws.com/prod/

Athena Proxy URL
[redacted]cloudfront.net:443

API Keys
List of access tokens used to programatically access the API

API Key	Status	Expiration	Created	Actions
No records found				

Integration endpoint API URL

```
curl –request GET '<api_url>/<api_path>' \
--header 'Accept: application/json' \
--header 'Authorization: api-key <api-key>'
```

List of ADA API endpoints

To view the list of current ADA API endpoints in OpenAPI specification format, refer to the ADA API specification.

To view OpenAPI specification in human friendly format, you can use open source tools such as Redoc. For example, refer to this API documentation for ADA as rendered by Redoc.

Use the ADA API for query retrieval and execution

This example shows to use the ADA API to programmatically run a query, retrieve the query run status, get query results, and download query results on ADA using cURL.

1. Send a POST query request to submit a query.

```
curl –request POST ‘<api_url>/query' \
--header 'Authorization: api-key <api_key>' \
--header 'Content-Type: application/json' \
--data-raw '{"query":"SELECT * FROM test_domain.test_data_product"}'
```

The `query` in the payload should include your SQL query content.

The API returns an execution id (***<query_execution_id>*** in the code examples below) which can be used to query execution status, retrieve query result or download query result.

2. Send a GetQueryStatus request to query the run status.

```
curl --request GET '<api_url>/query/<query_execution_id>/status' \
--header 'Authorization: api-key <api_key>'
```

When the returned status is SUCCESS, you can retrieve the query result.

3. Send a ListQueryResults request to retrieve the query run result.

```
curl --request GET '<api_url>/query/<query_execution_id>/result' \
--header 'Authorization: api-key <api_key>'
```

4. To download the query result as a file, send a GetQueryResultDownload request. The API will return a S3 presigned URL that allows you to download the file.

```
curl --request GET '<api_url>/query/<query_execution_id>/result/download' \
--header 'Authorization: api-key <api_key>'
```

5. To download the file, copy/paste the S3 presigned url to a web browser or use cURL command.

```
curl <s3_presigned_url> --output <output_file_path>
```

Note

The S3 presigned url returned by the API has an expiration time of 1 minute. To avoid timeout, you can run step 4 and 5 in one cURL command.

```
curl --silent --request GET '<api_url>/query/<query_execution_id>/result/
download' \
--header 'Authorization: api-key <api_key> | jq '.signedUrl' | xargs curl
 <output_file_path>
```

Use case - Visualize your data using Amazon QuickSight

You can build a script or a custom data application enabling you to visualize your data on Amazon QuickSight.

1. Run a query and upload the query result to an Amazon S3 bucket.
2. Create a dataset using the uploaded file in Amazon QuickSight so that you can visualize your data.
3. Periodically repeat step 1 to update your query so that your QuickSight dashboard is updated.

For instructions on how to create a dataset using S3 files on Amazon QuickSight, refer to the Creating a dataset using Amazon S3 files page.

Reference

This section includes information about an optional feature for collecting unique metrics for this solution, pointers to related resources, and a list of builders who contributed to this solution.

Anonymized data collection

This solution includes an option to send anonymized operational metrics to AWS. We use this data to better understand how customers use this solution and related services and products. When invoked, the following information is collected and sent to AWS:

- **Solution ID** - The AWS solution identifier
- **Unique ID (UUID)** - Randomly generated, unique identifier for each Automated Data Analytics on AWS deployment
- **Timestamp** - Data-collection timestamp

AWS owns the data gathered though this survey. Data collection is subject to the AWS Privacy Policy.

Opt out of operational metrics collection

To opt out of this feature, for the solution **parameters**, set `sendAnonymousData` parameter to **No**.

Contributors

Product

- Kyle Fox
- Hafiz Saadullah
- Tim O’Hare
- Tony Spelde
- Marc Teichtahl

Engineering

- Jeremy Jonas
- Warren Wei
- Jack Stevenson
- Hu Jin
- Van Vo Thanh
- Rashim Rahman
- Jessie Wei
- Mirash Gjolaj
- Emily Ha
- APJ Prototyping team
- APJ Solutions Engineering team

Technical Writing and Documentation

- Swapnil Ogale
- Suyog Sainkar

Security Guardian review

- Deenadayaalan Thirugnanasambandam

Revisions

For more information, see the CHANGELOG.md file in the GitHub repository.

Date	Change
August 2022	v1.0.0 Initial release
December 2022	v1.1.0 • Updated the Architecture Overview diagram for new connectors. • Updated costs table with AWS Transit Gateway costs. • Added content for connectors: Amazon CloudWatch, SQL Database connectors, MySQL5.x, PostgreSQL and Microsoft SQL Server in the Provide source details section. • Added information for cross account assume role setup. • Added Update Policy information for data sources. • Added content for Setting up ADA to access data sources within VPN or on-premise network.
April 2023	v1.2.0 • Updated the Architecture Overview diagram for new connectors. • Added a new Data connectors guide to provides instructions on how to ingest data from a variety of data connectors. • Added instructions to update the solution.

Date	Change
	- Added a new section on monitoring the solution with Service Catalog AppRegistry. - Added a new Developer Guide, including instructions to access the ADA APIs. - Added a new Troubleshooting section for capturing common problems and how to resolve these issues. - Mitigated impact caused by new default settings for S3 Object Ownership (ACLs disabled) for all new S3 buckets. For more information, refer to Issue 40 in the GitHub repository.
September 2023	v1.3.0 - Added instructions for setting up governanc e and configuring governance attributes. - Added information for two new connectors: Amazon Redshift and Oracle databases. - Updated the architecture diagram to show the new connectors supported by the solution.
April 2024	v1.4.0 - Added instructions to create a budget with a cost limit to track costs and usage. - Added instructions on how to set up Apache Superset as a data analytics and visualiza tion platform. - Added a new Extensions guide for instructi ons on how to extend ADA's capabilities with Apache Superset and Amazon AppFlow.

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents AWS current product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. AWS responsibilities and liabilities to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.