

Building an interactive global map of power plants

Final Project for CSCI E90

Anthony Chon Lane

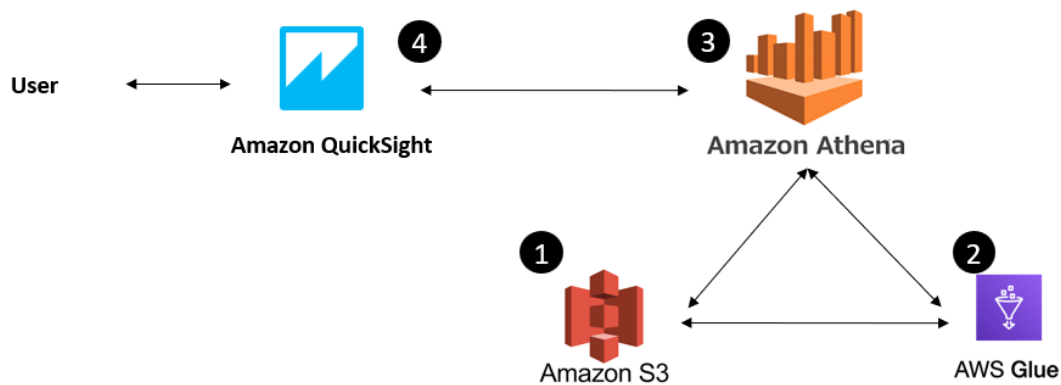
Harvard Extension School

1. Solution overview

In the context of inflation, climate change and the war in Ukraine, generating sufficient, sustainable, and cost-effective electrical power is a significant economic advantage for a country.

Using publicly available data, I want to uncover how countries power modern society through power plants against environmental impacts based on today's technology and capacity estimates. For instance, which are the top electricity power nations? To what extent do they rely on carbon-emitting power sources to fuel their plants?

To address these questions, I have built an interactive world map displaying all the power plants filtered by location, fuel type, and capacity. Leveraging the AWS services we learned in class, I have developed a practical application to uncover valuable insights.



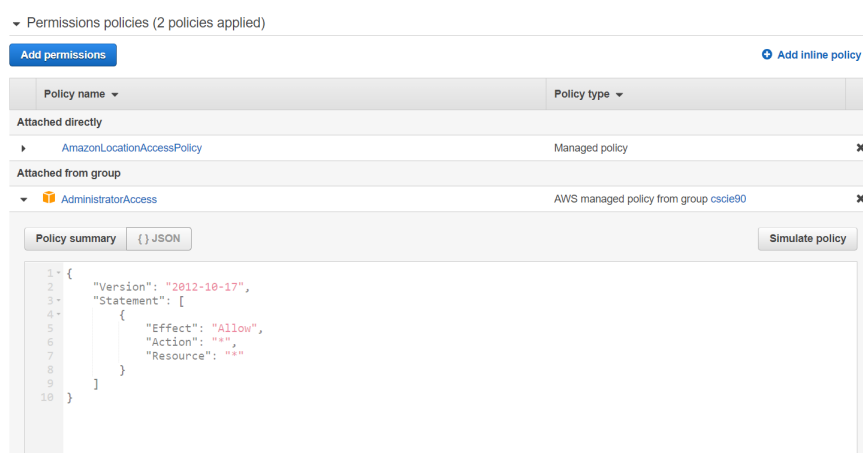
In this demo, you will explore how to use Amazon S3, Amazon Glue, Amazon Athena, and QuickSight to populate an interactive map and display organized information. The above diagram summarizes the different components of this solution.

2. Prerequisites

1. Set up AWS Account

You need access to an AWS account with the necessary permissions to create new AWS resources, and access to the AWS Management Console.

You also need to have Amazon QuickSight enabled within your AWS account. For the sake of convenience, I will use the existing AWS user account (anthonycl-e90) I created for this course.



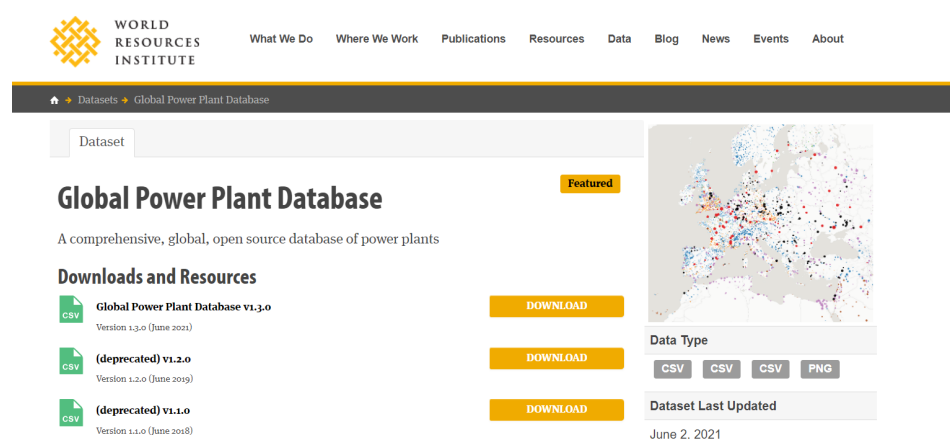
The screenshot shows the AWS IAM console interface. At the top, it indicates 'Permissions policies (2 policies applied)'. Below this, there are two sections: 'Attached directly' and 'Attached from group'. The 'Attached from group' section is expanded to show the 'AdministratorAccess' policy. The 'Policy summary' tab is active, displaying the following JSON policy document:

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": "*",
7       "Resource": "*"
8     }
9   ]
10 }
```

2. Download dataset

To populate relevant data, we download a .csv file of dataset obtained from the [World Resources Institute](#) that contains the geocoordinates of power plants around the world. We will place our dataset in Amazon S3 under a new bucket.

The World Resources Institute is a global research non-profit organization that provides research across various issues such as environmental sustainability, economic opportunity, and human health and well-being.



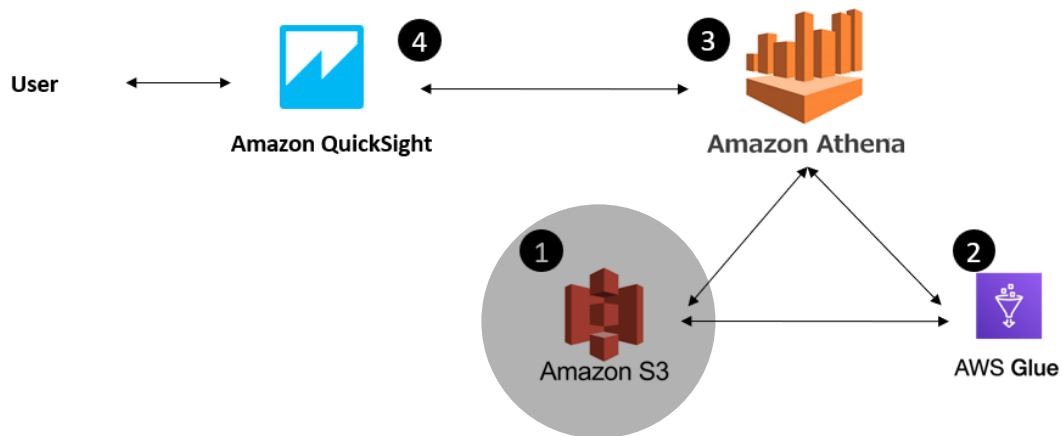
The screenshot shows the World Resources Institute website. The header includes the logo and navigation links: 'What We Do', 'Where We Work', 'Publications', 'Resources', 'Data', 'Blog', 'News', 'Events', and 'About'. The main content area is titled 'Global Power Plant Database' and is marked as 'Featured'. It describes the database as 'A comprehensive, global, open source database of power plants'. Under 'Downloads and Resources', there are three download options for CSV files:

- Global Power Plant Database v1.3.0 (Version 1.3.0 (June 2021))
- (deprecated) v1.2.0 (Version 1.2.0 (June 2019))
- (deprecated) v1.1.0 (Version 1.1.0 (June 2018))

Each option has a 'DOWNLOAD' button. To the right, there is a map of the world showing power plant locations. Below the map, there are options for 'Data Type' (CSV, CSV, CSV, PNG) and 'Dataset Last Updated' (June 2, 2021).

3. Walkthrough

1. Create a new bucket in Amazon S3



In the console, go to Amazon S3 and create a new bucket named "power-plants" in the region of your choice*.

Create bucket [Info](#)

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name

Bucket name must be globally unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

AWS Region

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

* Ensure you enable access to your Amazon QuickSight account in the same region.

Leave the other bucket settings by default. For this project, we do not need ACLs nor bucket versioning and block all public access.

Object Ownership [Info](#)

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership
Bucket owner enforced

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through new access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through new public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through any public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning

Disable

Enable

Tags (0) - optional

You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

No tags associated with this bucket.

Default encryption

Automatically encrypt new objects stored in this bucket. [Learn more](#)

Server-side encryption

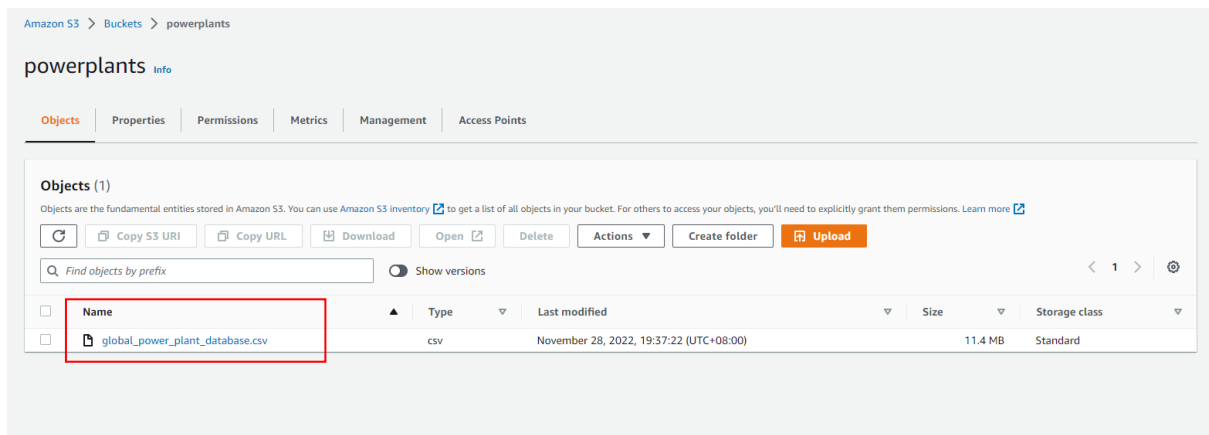
Disable

Enable

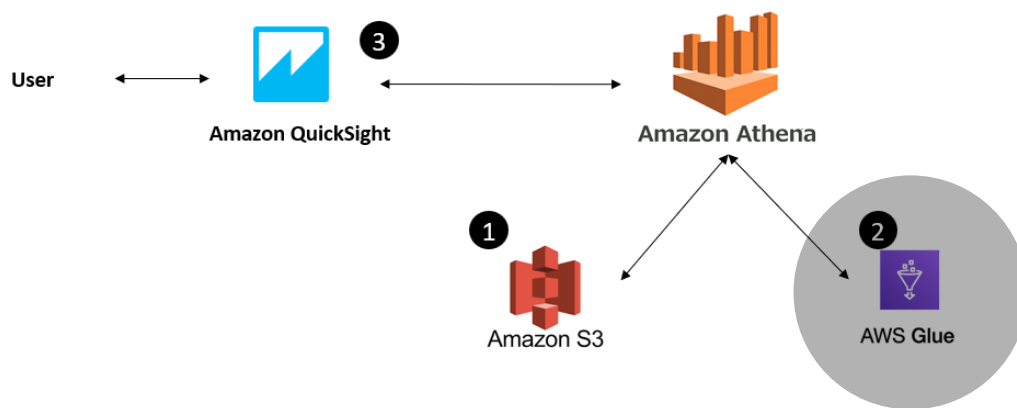
► **Advanced settings**

ⓘ After creating the bucket you can upload files and folders to the bucket, and configure additional bucket settings.

Upload the .csv file into the new bucket.

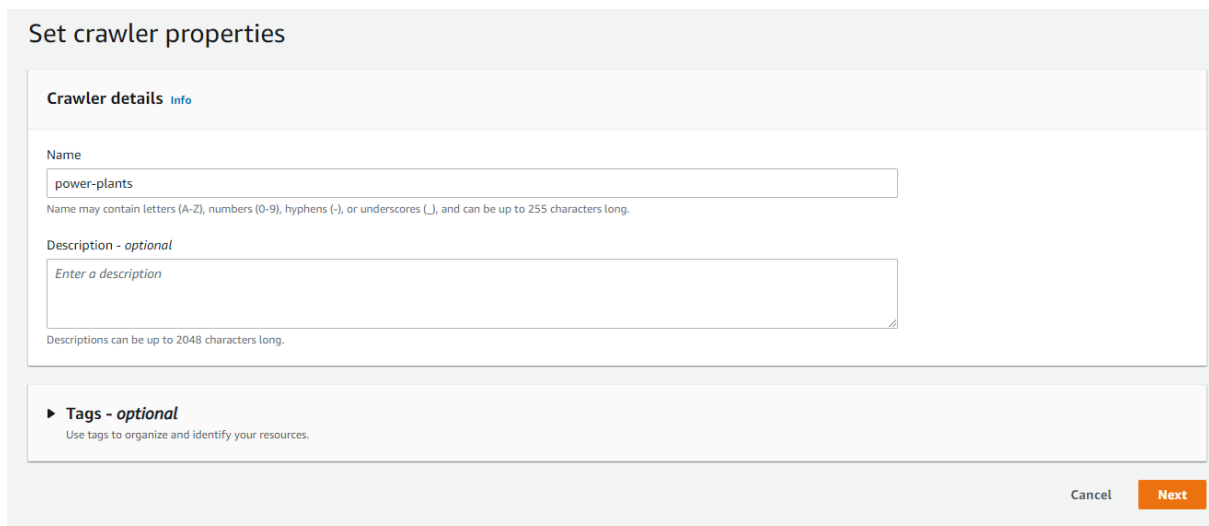


2. Define data format of the .csv file using Amazon Glue



In this step, we create a database and table in Amazon Glue that define the data format of the .csv file in Amazon S3 to allow it to be queried from Amazon Athena.

2.1. Create a crawler to populate the AWS Glue Data Catalog with tables.



2.1.1. Add your S3 bucket as a data source.

Choose data sources and classifiers

Data source configuration

Is your data already mapped to Glue tables?

Not yet
Select one or more data sources to be crawled.

Yes
Select existing tables from your Glue Data Catalog.

Data sources (0) [Info](#)

The list of data sources to be scanned by the crawler.

[Edit](#) [Remove](#) [Add a data source](#)

Type	Data source	Parameters
You don't have any data sources.		

[Add a data source](#)

2.1.2. Enable the crawler to crawl all sub-folders in the S3 data source.

Add data source ✕

Data source
Choose the source of data to be crawled.

S3

Network connection - optional
Optionally include a Network connection to use with this S3 target. Note that each crawler is limited to one Network connection so any other S3 targets will also use the same connection (or none, if left blank).

▼ [↻](#)

[Clear selection](#) [Add new connection](#)

Location of S3 data

In this account
 In a different account

S3 path
Browse for or enter an existing S3 path.

✕ [View](#) [Browse](#)

All folders and files contained in the S3 path are crawled. For example, type s3://MyBucket/MyFolder/ to crawl all objects in MyFolder within MyBucket.

Subsequent crawler runs
This field is a global field that affects all S3 data sources.

Crawl all sub-folders
Crawl all folders again with every subsequent crawl.

Crawl new sub-folders only
Only Amazon S3 folders that were added since the last crawl will be crawled. If the schemas are compatible, new partitions will be added to existing tables.

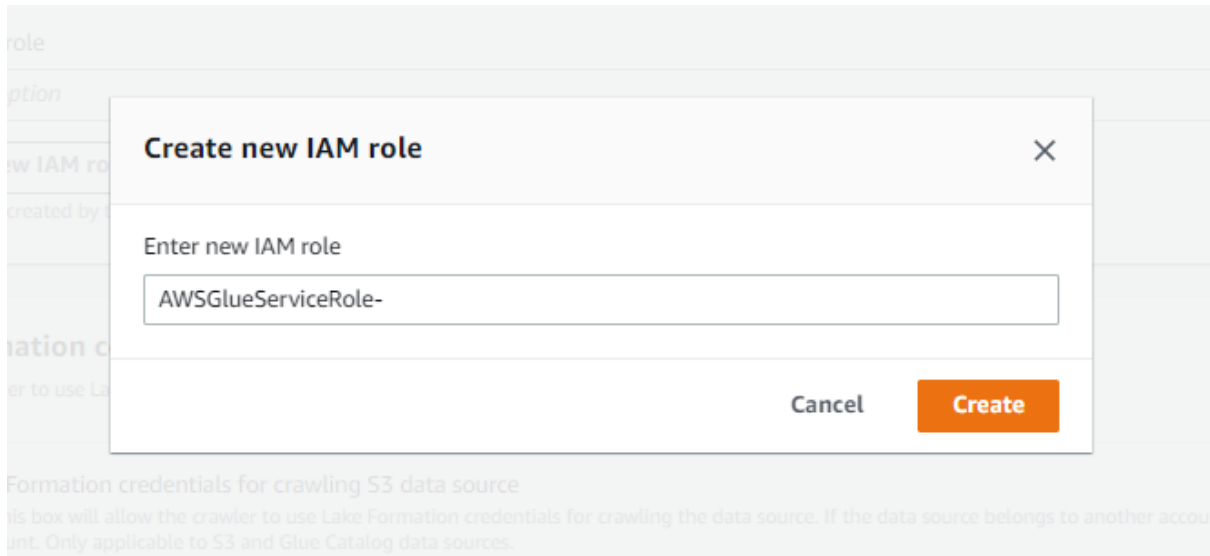
Crawl based on events
Rely on Amazon S3 events to control what folders to crawl.

Sample only a subset of files

Exclude files matching pattern

[Cancel](#) [Add an S3 data source](#)

2.1.3. Configure security settings by creating a new IAM role named "AWSGlueServiceRole-finalproject".



Configure security settings

IAM role [Info](#)

Existing IAM role

AWSGlueServiceRole-finalproject ▼



[View](#)

Create new IAM role

Update chosen IAM role

Only IAM roles created by the AWS Glue console and have the prefix "AWSGlueServiceRole-" can be updated.

Lake Formation configuration - *optional* [Preview](#)

Allow the crawler to use Lake Formation credentials for crawling the data source. [Learn more.](#)

Use Lake Formation credentials for crawling S3 data source

Checking this box will allow the crawler to use Lake Formation credentials for crawling the data source. If the data source belongs to another account, you must provide the registered account ID. Otherwise, the crawler will crawl only those data sources associated to the account. Only applicable to S3 and Glue Catalog data sources.

2.1.4. Add a database to set output and keep crawler schedule on demand.

Set output and scheduling

Output configuration [Info](#)

Target database
Choose a database

Table name prefix - optional
Type a prefix added to table names

Maximum table threshold - optional
This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.
Type a number greater than 0

▶ **Advanced options**

Crawler schedule
You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron [syntax](#). [Learn more](#) [↗](#)

Frequency

We will name our database "power-plants" for consistency.

Create a database
Create a database in the AWS Glue Data Catalog.

Database details

Name

Database name is required, in lowercase characters, and no longer than 255 characters.

Location - optional
Set the URI location for use by clients of the Data Catalog.

Description - optional
Enter text

Descriptions can be up to 2048 characters long.

Review and create.

Review and create

Step 1: Set crawler properties

Set crawler properties

Name power-plants	Description -	Tags -
----------------------	------------------	-----------

Step 2: Choose data sources and classifiers

Data sources (1) [Info](#)
The list of data sources to be scanned by the crawler.

Type	Data source	Parameters
s3	s3://power-plants	Recrawl all

Step 3: Configure security settings

Configure security settings

IAM role AWSGlueServiceRole-finalproject	Security configuration -	Lake Formation configuration -
---	-----------------------------	-----------------------------------

Step 4: Set output and scheduling

Set output and scheduling

Database power-plants	Table prefix - optional -	Maximum table threshold - optional -	Schedule On demand
--------------------------	------------------------------	---	-----------------------

2.2. Run the crawler and go to Amazon Athena to check that our crawler creates a table in the AWSDataCatalog for us.

The screenshot shows the Amazon Glue Crawlers console. At the top right, it says "Last updated: December 2, 2022 at 05:23:42 (UTC)". Below that, there are buttons for "Action", "Run", and "Create crawler". The "Run" button is highlighted with a red box. Below the buttons is a table with columns: Name, State, Schedule, Last run, Log, and Table changes from last run. One crawler named "power-plants" is listed with a state of "Ready" and a last run status of "Succeeded".

We can retrieve the dataset using the Athena query editor.

The screenshot shows the Amazon Athena Query Editor. The query editor contains the following SQL query: `SELECT * FROM "power-plants"."power-plants" limit 10;`. Below the query editor, there are buttons for "Run again", "Explain", "Cancel", "Clear", and "Create". The "Query results" tab is active, showing a "Completed" status and "Results (10)". The results are displayed in a table with the following columns: #, country, country_long, name, gppd_idnr, capacity_mw, latitude, longitude, primary_fuel, and other_fuel. The results show three rows of data for power plants in Afghanistan.

#	country	country_long	name	gppd_idnr	capacity_mw	latitude	longitude	primary_fuel	other_fuel
1	AFG	Afghanistan	Kajaki Hydroelectric Power Plant Afghanistan	GEODB0040538	33.0	32.322	65.119	Hydro	
2	AFG	Afghanistan	Kandahar DOG	WKS0070144	10.0	31.67	65.795	Solar	
3	AFG	Afghanistan	Kandahar JOL	WKS0071196	10.0	31.623	65.792	Solar	

This is a close-up screenshot of the Amazon Athena Query Editor. It shows the "Data source" dropdown menu set to "AwsDataCatalog" and the "Database" dropdown menu set to "power-plants". Both dropdown menus are highlighted with a red box.

2.3. Create a job to clean your database using Amazon Glue DataBrew.

In Amazon DataBrew, create a new project called "power-plants" and attach a new recipe "power-plants-recipe".

DataBrew > Projects > Create project

Create project [Info](#)

Project details

Project name
power-plants
The project name must contain 1-255 characters. Valid characters are alphanumeric (A-Z, a-z, 0-9), hyphen (-), period (.), and space.

Recipe details [Info](#)
Data cleaning steps in DataBrew are stored as a recipe. A recipe is connected to a project by default. An existing recipe with no associated project could also be applied to a project.

Attached recipe
Create new recipe ▼
Recipe name
power-plants-recipe
The recipe name must contain 1-255 characters. Valid characters are alphanumeric (A-Z, a-z, 0-9), hyphen (-), period (.), and space.

Import steps from recipe
Import recipe steps from an existing recipe into your project. The existing recipe that you chose will not be edited.

Select our existing dataset, the one located at "s3://power-plants/global_power_plant_database.csv" from S3.

Select a dataset
Select the dataset that you want to work on

My datasets
Your imported datasets

Sample files
Explore example files for your dataset

New dataset
Import new dataset

Find datasets

Dataset name	Data type	Source	Create date
power-plants	csv	S3	2 days ago November 30, 2022, 9:32:54 am

Create a new IAM role named 'AWSGlueDataBrewServiceRole-finalproject' to enable permissions and click create project.

Permissions [Info](#)
DataBrew needs permission to connect to data on your behalf. Use an IAM role with the [required policy](#) attached.

Role name
Choose the role that has access to connect to your data. Refresh to see the latest updates.
Create new IAM role ▼

New IAM role suffix
Your role will be prefixed with "AWSGlueDataBrewServiceRole-"
finalproject

By clicking "Create project" you are authorizing creation of this role.

As soon as you create a DataBrew project, the project opens and costs begin to accrue to your AWS account. [Pricing details](#)

Cancel **Create project**

Let your "power-plants" table load from the .csv file in your S3 bucket.

Created project "power-plants"

power-plants
Dataset: power-plants | Sample: First n samples (0 rows)

Viewing 0 rows

Your session will be ready soon!
Provisioning compute
0%

Your session will take about a minute to be ready. Once ready there will be no additional load time.

Add steps to your recipe to clean up your database as part of the new job:

1. Delete unnecessary columns.
2. Remove special characters, numbers, white spaces, quotation marks and punctuation from relevant columns when needed.
3. Format text columns (Capital case) and number columns (decimal precision).
4. Publish your recipe.

The screenshot shows the DataBrew interface. On the left, a table view of the 'power-plants' dataset is visible with columns like 'country_long', 'name', 'capacity_mw', 'latitude', 'longitude', 'primary_fuel', and 'commissioning_year'. On the right, a 'Recipe (8)' panel is open, showing a list of applied steps:

1. Delete columns: country, gpid, id, other_fuel1, other_fuel2, other_fuel3, source, url, geolocation, source, w_epp_id, generation_gwh_2013, generation_gwh_2014, generation_gwh_2015, generation_gwh_2016, generation_gwh_2017, generation_gwh_2018, generation_gwh_2019, generation_data_source, estimated_generation_gwh_2013, estimated_generation_gwh_2014, estimated_generation_gwh_2015, estimated_generation_gwh_2016, estimated_generation_gwh_2017, estimated_generation_note_2013, estimated_generation_note_2014, estimated_generation_note_2015, estimated_generation_note_2016, estimated_generation_note_2017, year_of_capacity_data
2. Remove special characters, numbers, white spaces, quotation marks, punctuation from country_long
3. Change format of name to Capital case
4. Remove special characters, white spaces, quotation marks, punctuation from name
5. Change format of capacity_mw to decimal precision
6. Remove special characters, numbers, white spaces, quotation marks, punctuation from primary_fuel
7. Change format of commissioning_year to decimal precision
8. Remove special characters, white spaces, quotation marks, punctuation from owner

Go to jobs and create a new job named "power-plants-clean". Then, your 'power-plants' dataset and "power-plants-recipe" recipe.

The screenshot shows the 'Create job' form in DataBrew. The 'Job name' field is filled with 'power-plants-clean'. Under 'Job type', 'Create a recipe job' is selected. In the 'Job input' section, 'Run on' is set to 'Dataset'. The 'Choose dataset' dropdown is set to 'power-plants', and the 'Select a recipe' dropdown is set to 'power-plants-recipe' (Version 2.0).

For simplicity, the job output will go under our existing "power-plants" S3 bucket.

Job output settings [Info](#)
Running a job generates output files at specified file destinations.

Output 1 [Settings](#)

Output to Output location	File type Output format	Delimiter CSV separator	Compression Available types
<input type="text" value="Amazon S3"/>	<input type="text" value="CSV"/>	<input type="text" value="Comma (,)"/>	<input type="text" value="None"/>

S3 bucket owner's AWS account

Current AWS account
304024973019

Another AWS account

S3 location
Format is: s3://bucket/folder/

Setting summary [Info](#)
File output storage
Create a new folder for each job run
File output
Autogenerate files
Custom partition by column values
Disabled

Output path preview
s3://power-plants/power-plants-clean_02Dec2022_
timestamp_part00000.csv

Create a new IAM role, "**AWSGlueDataBrewServiceRole-cleaning**", and click create and run a job.

Permissions [Info](#)
DataBrew needs permission to connect to data on your behalf. Use an IAM role with the [required policy](#) attached.

Role name
Choose the role that has access to connect to your data. Refresh to see the latest updates.

New IAM role suffix
Your role will be prefixed with "AWSGlueDataBrewServiceRole-"

By clicking "Create job" you are authorizing creation of this role.

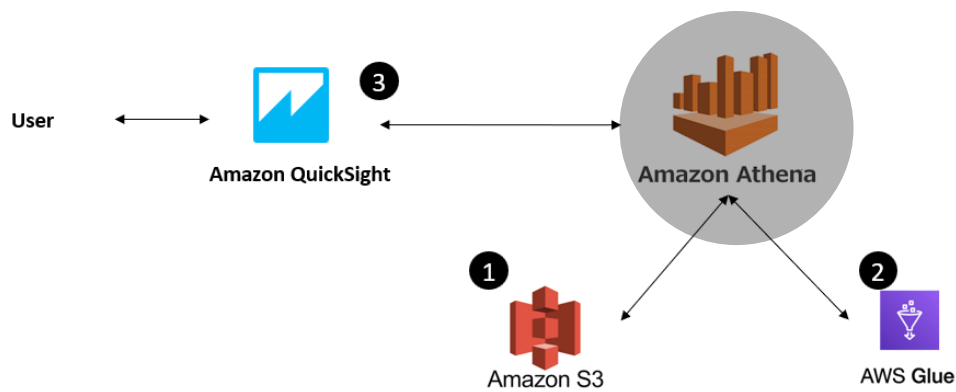
Run your crawler again to populate your clean table in the AWSDatacatalog.

Crawlers
A crawler connects to a data store, progresses through a prioritized list of classifiers to determine the schema for your data, and then creates metadata tables in your data catalog.

Crawlers (1/1) [Info](#)
View and manage all available crawlers.

<input checked="" type="checkbox"/>	Name	State	Schedule	Last run
<input checked="" type="checkbox"/>	power-plants	Running		✔ Succeeded

3. Query your table in Amazon Athena



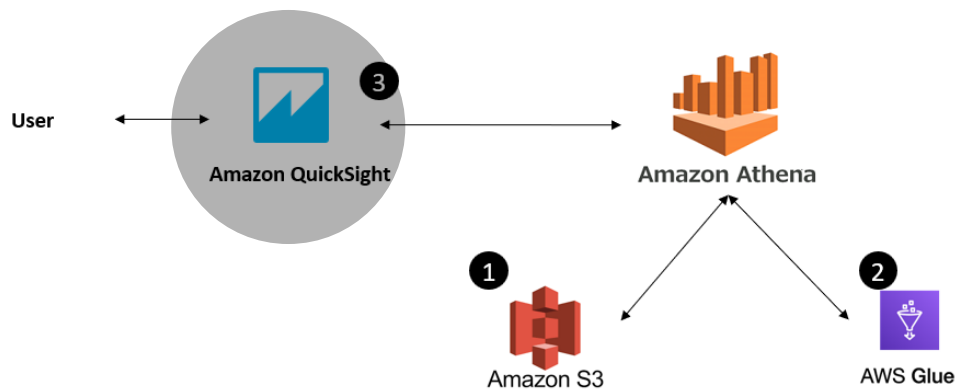
Now we can go to Amazon Athena Query Editor to view our AWSDataCatalog database and query our tables. Our DataBrew job has created a new table named `power_plants_clean_date`.

Table Name	Schema
power_plants	
power_plants_clean_02dec2022_16699	
71056056	
country	string
name	string
capacity_mw	double
latitude	double
longitude	double
primary_fuel	string
commissioning_year	bigint
owner	string
generation_gwh_2019	string

We can run a query to prepare our data and view our new table with the following command line: **SELECT * FROM "power-plants"."power_plants_clean_date";**

```
SQL Ln 1, Col 1
Run again Explain Cancel Clear Create
Query results Query stats
Completed Time in queue: 167 ms Run time: 719 ms Data scanned: 701.27
Results (10) Copy Download results
Search rows
# country name capacity_mw latitude longitude primary_fuel commissioning_year owner generation_gwh_2019
1 Afghanistan "Kajaki Hydroelectric Power Plant Afghanistan" 33.0 32.322 65.119 Hydro
2 Afghanistan "Kandahar Dog" 10.0 31.67 65.795 Solar
3 Afghanistan "Kandahar Jol" 10.0 31.623 65.792 Solar
4 Afghanistan "Mahipar Hydroelectric Power Plant Afghanistan" 66.0 34.556 69.4787 Hydro
5 Afghanistan "Naghlu Dam Hydroelectric Power Plant Afghanistan" 100.0 34.641 69.717 Hydro
6 Afghanistan "Nangarhar Darunta Hydroelectric Power Plant Afghanistan" 11.55 34.4847 70.3653 Hydro
```

4. Visualize data in Amazon QuickSight



Now that the database looks clean, we can visualize our dataset in Amazon QuickSight.

4.1. Security and Permissions

Sign into your Amazon QuickSight account in the same region as your S3 and Athena services.

Under security and permissions, select the 'power-plants' bucket and the Athena service.

QuickSight access to AWS services

Make your existing AWS data and users available in QuickSight. [Learn more](#)

IAM Role

- Use QuickSight-managed role (default)
- Use an existing role

Allow access and autodiscovery for these resources

- Amazon Redshift
- Amazon RDS

This policy used by QuickSight for AWS resource access was modified outside of QuickSight, so you can no longer edit this policy to provide AWS resource permission to QuickSight. To edit this policy permissions, go to IAM console and delete this policy permission with policy arn:arn:aws:iam::304024973019:policy/service-role/AWSQuickSightRDSPolicy.

- IAM

- Amazon S3 (1 buckets selected)
[Select S3 buckets](#)
- Amazon Athena
Make sure you've chosen the right Amazon S3 buckets for QuickSight access

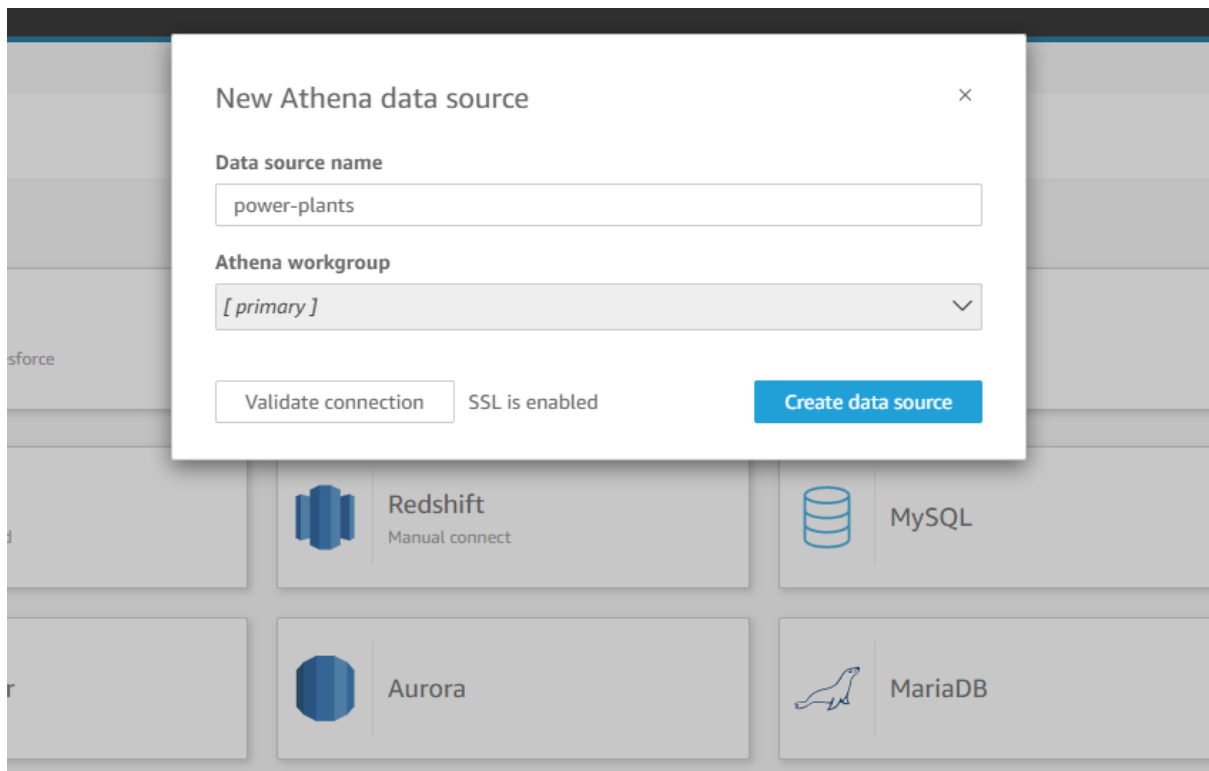
- Amazon S3 Storage Analytics
- AWS IoT Analytics
- Amazon OpenSearch Service
- Amazon SageMaker
- Amazon Timestream
- AWS SecretsManager
[Select secrets](#)

Save

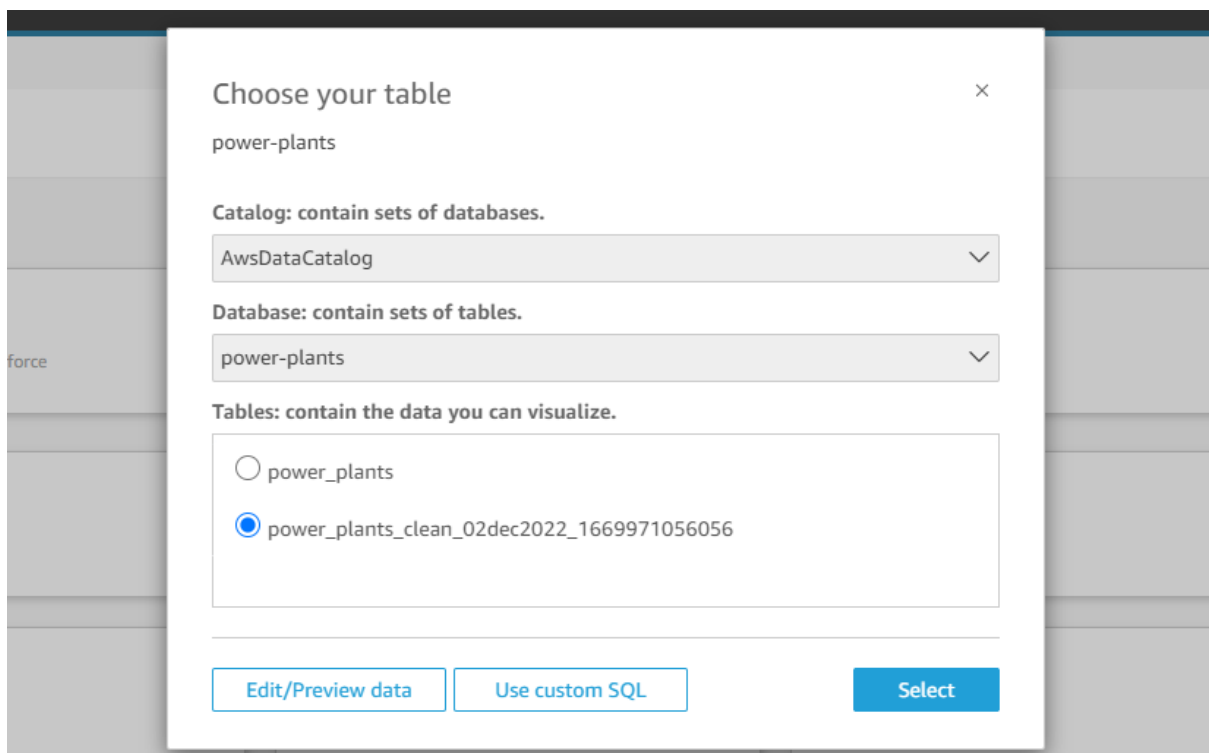
Cancel

4.2. Create a dataset

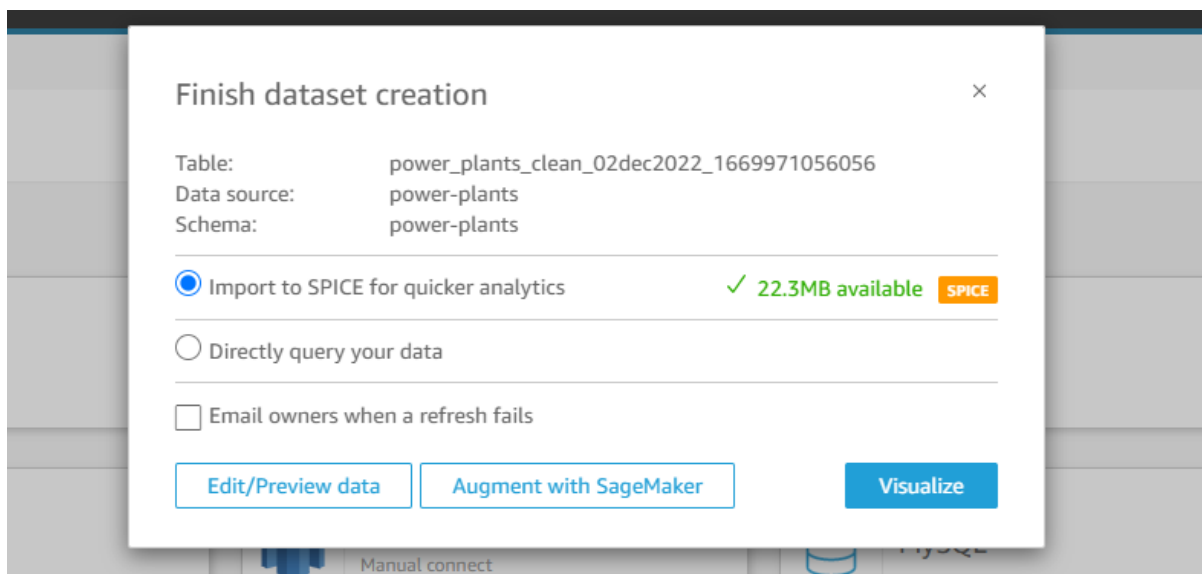
Under dataset, select Athena and add "power-plants" as a new Athena data source.



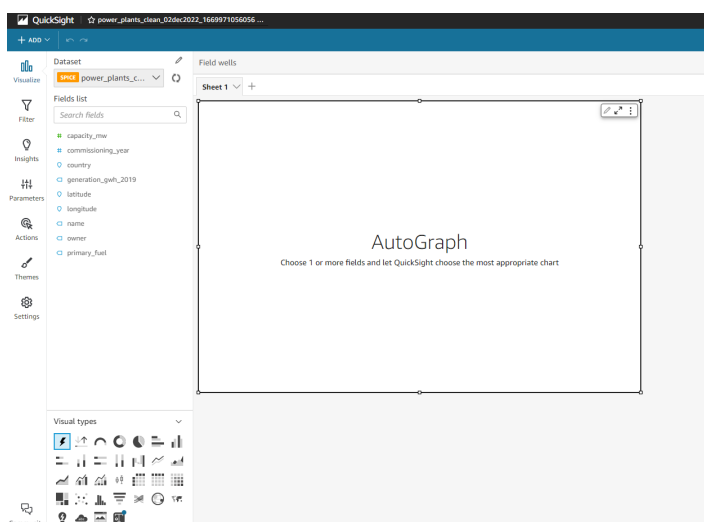
Choose your clean table.



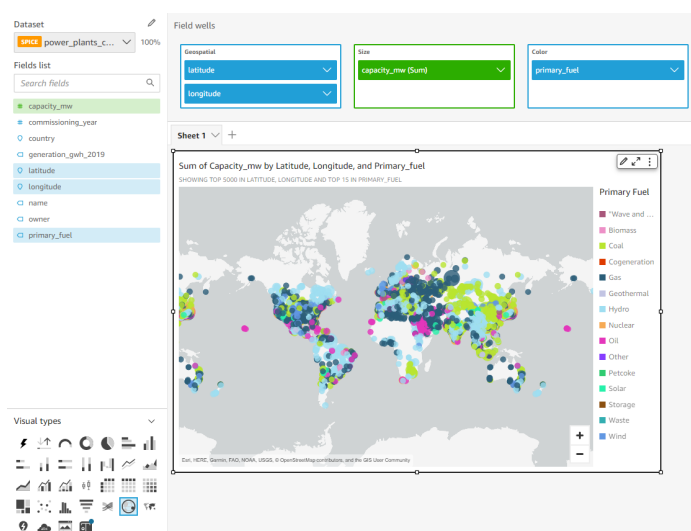
Import to Spice for quicker analytics and click visualize.



We can now visualize the fields to populate the interactive map.



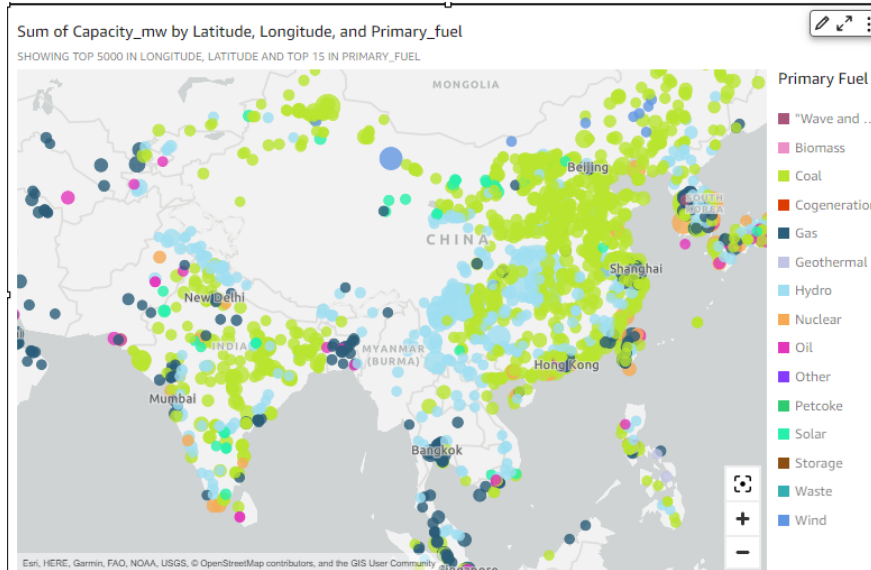
Select the "points on map" visual type to generate the map and add your fields to display the relevant information.



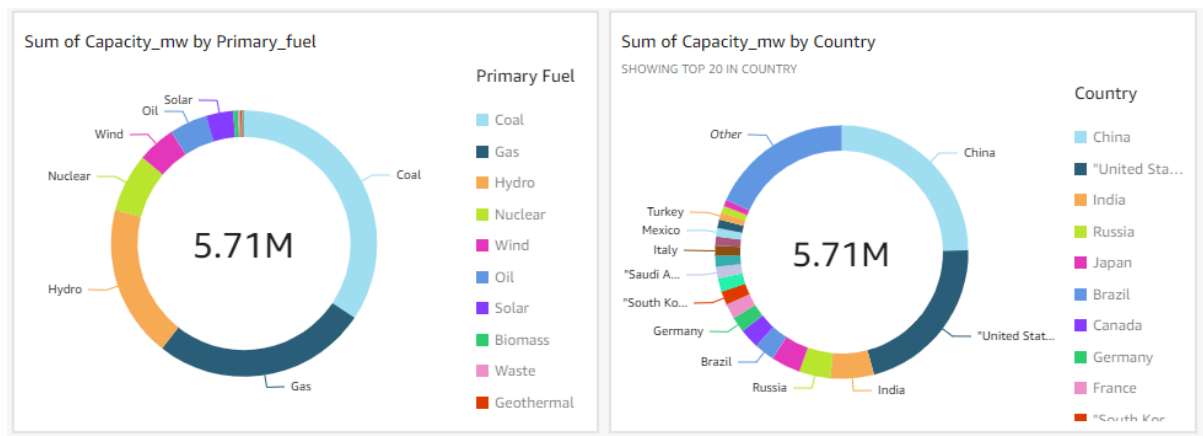
5. Results

We now have a real-world use case for data visualization based on a meaningful interactive map:

Not surprisingly, China and India rely mainly on coal to fuel power plants, which have significant impact on the environment.



Coal and Gas are the most significant fuel sources, and China and the US are the top electricity producers based on capacity.



We can publish it as a dashboard for further sharing and dissemination.

