



AWS Data Preparation Helper Tools



Brock Tubre

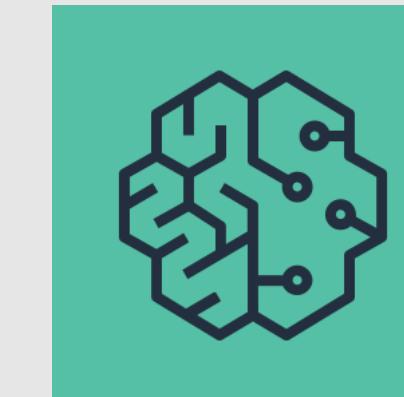
INSTRUCTOR

Data Preparation Helper Tools

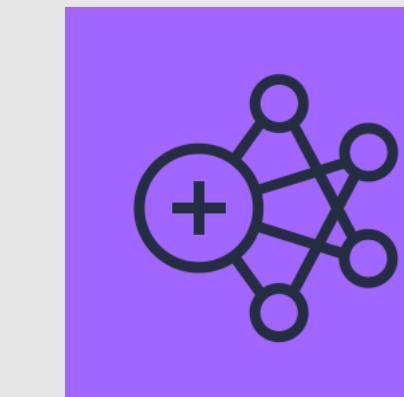
Data Preparation



AWS Glue



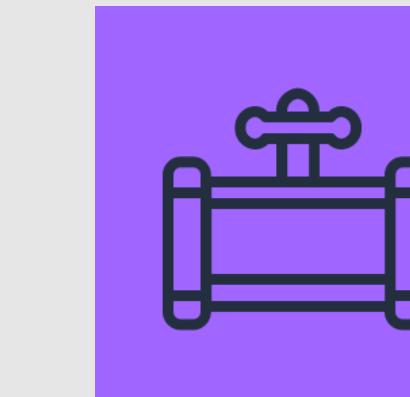
SageMaker



EMR



Athena

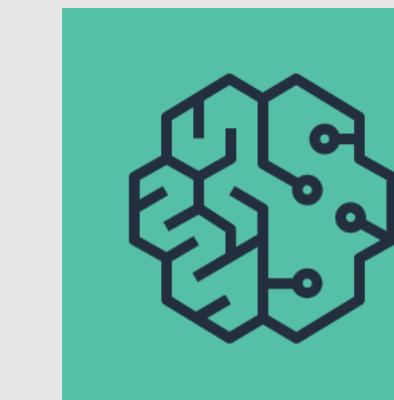


Data Pipeline

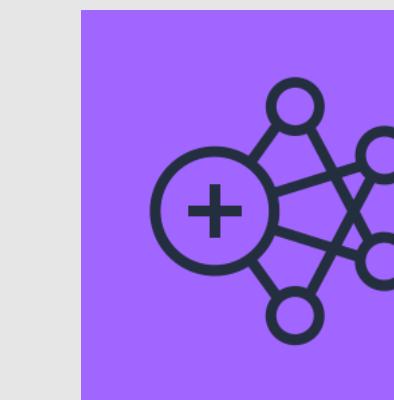
Data Preparation



AWS Glue



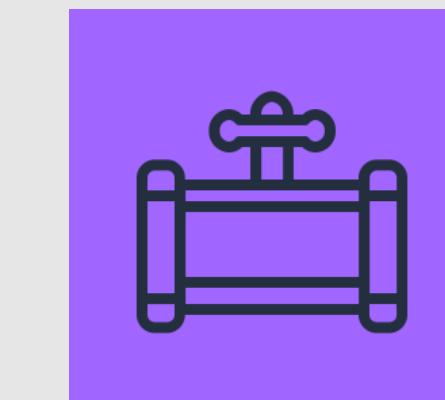
SageMaker



EMR

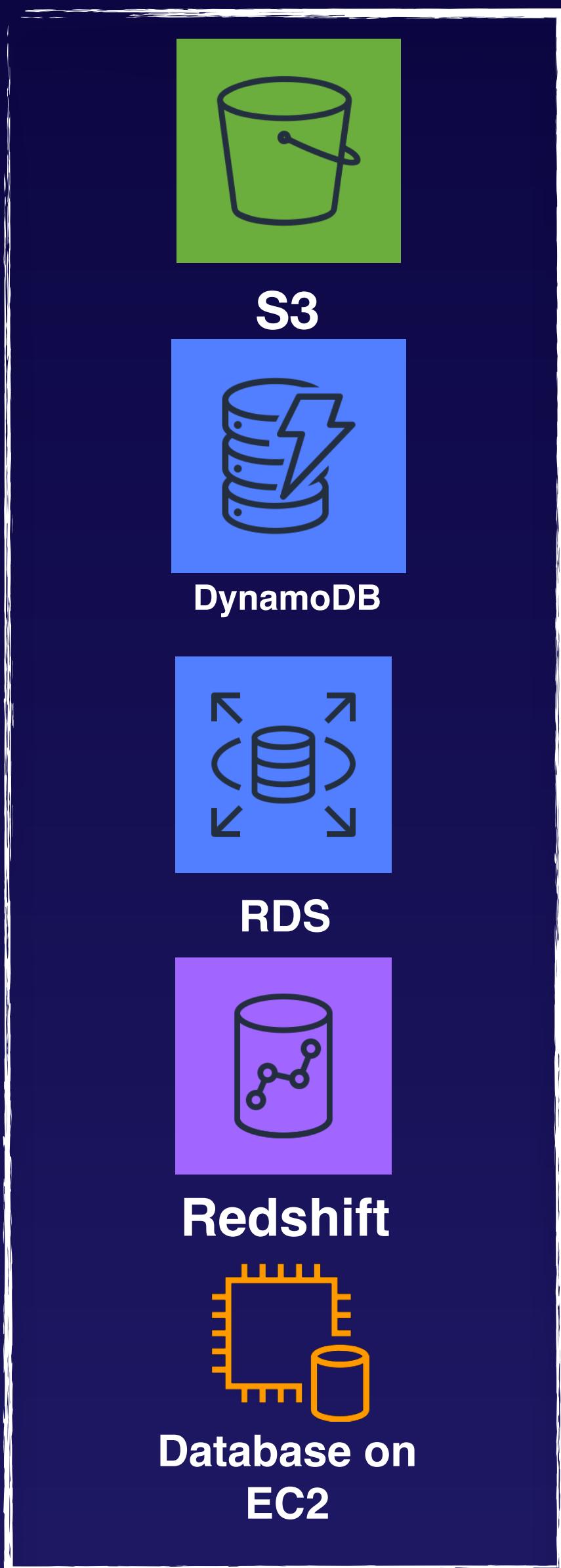


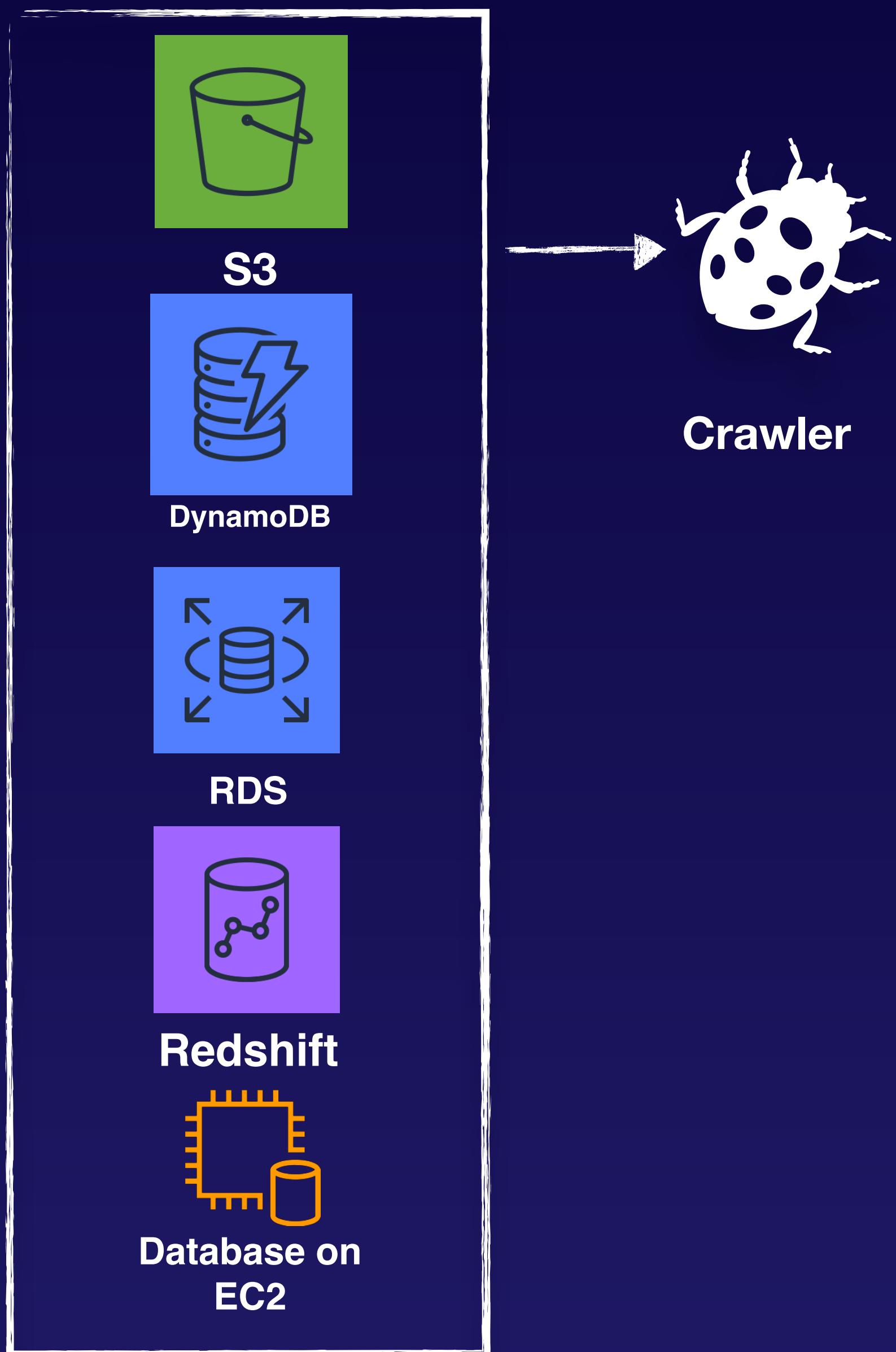
Athena

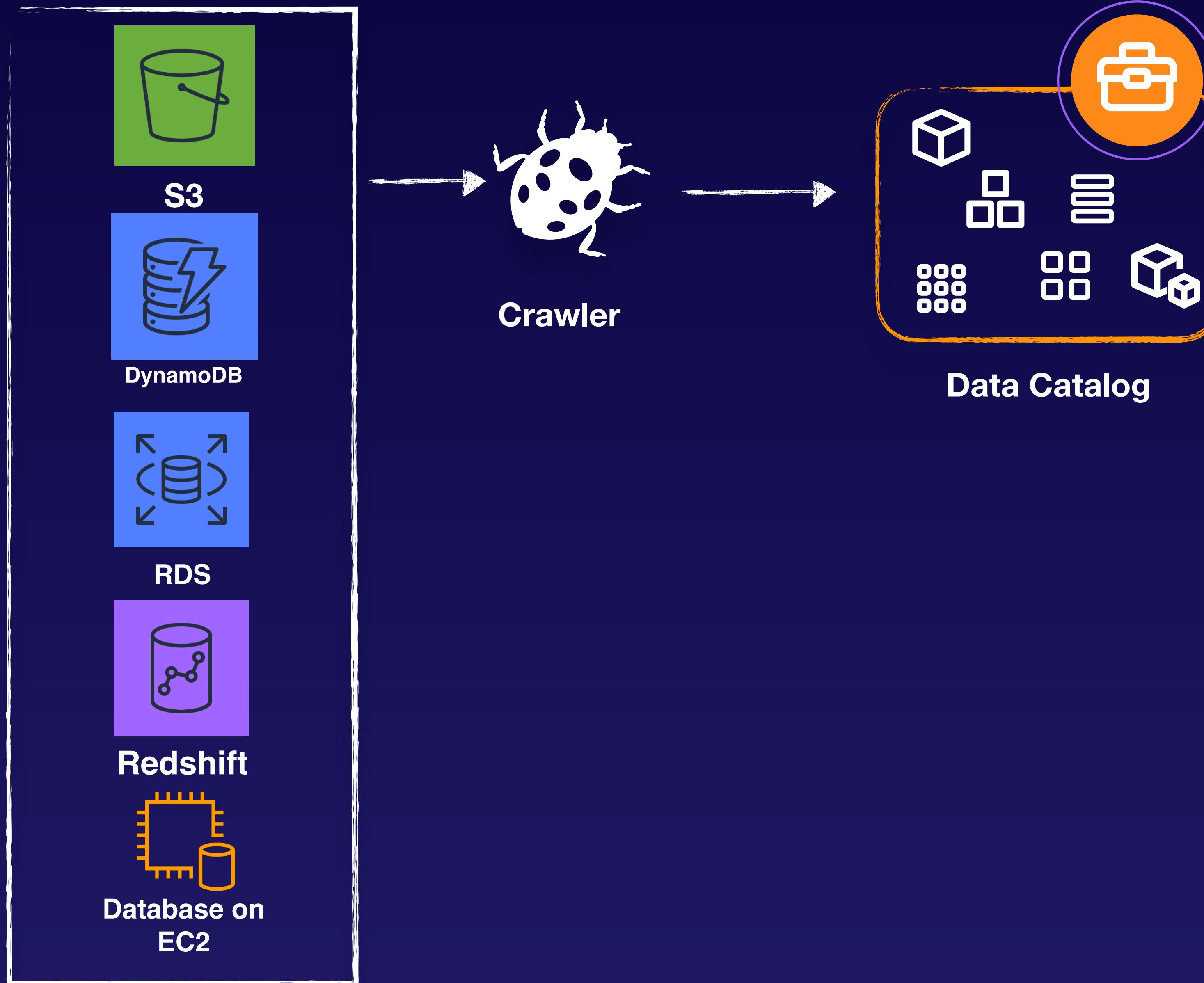


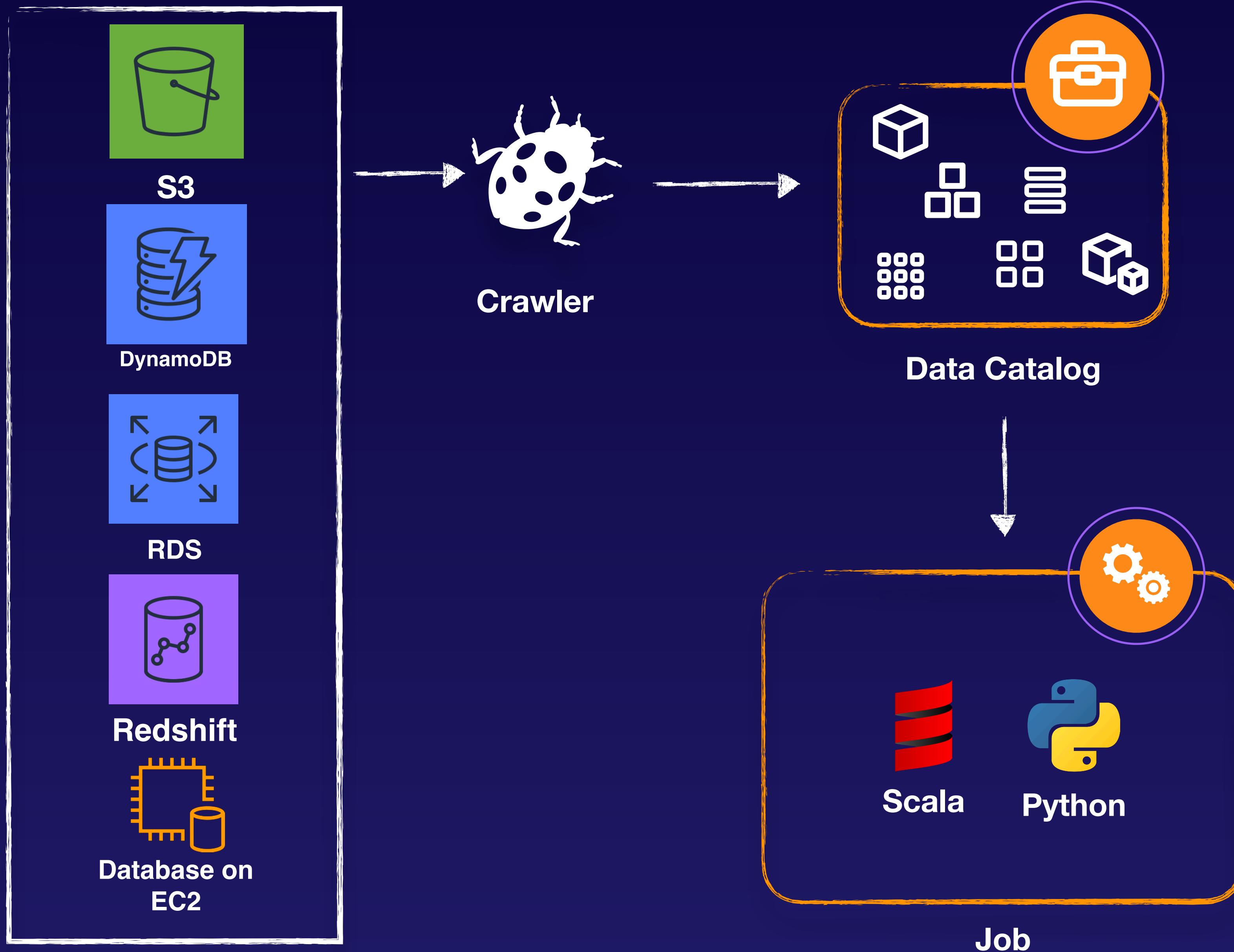
Data Pipeline



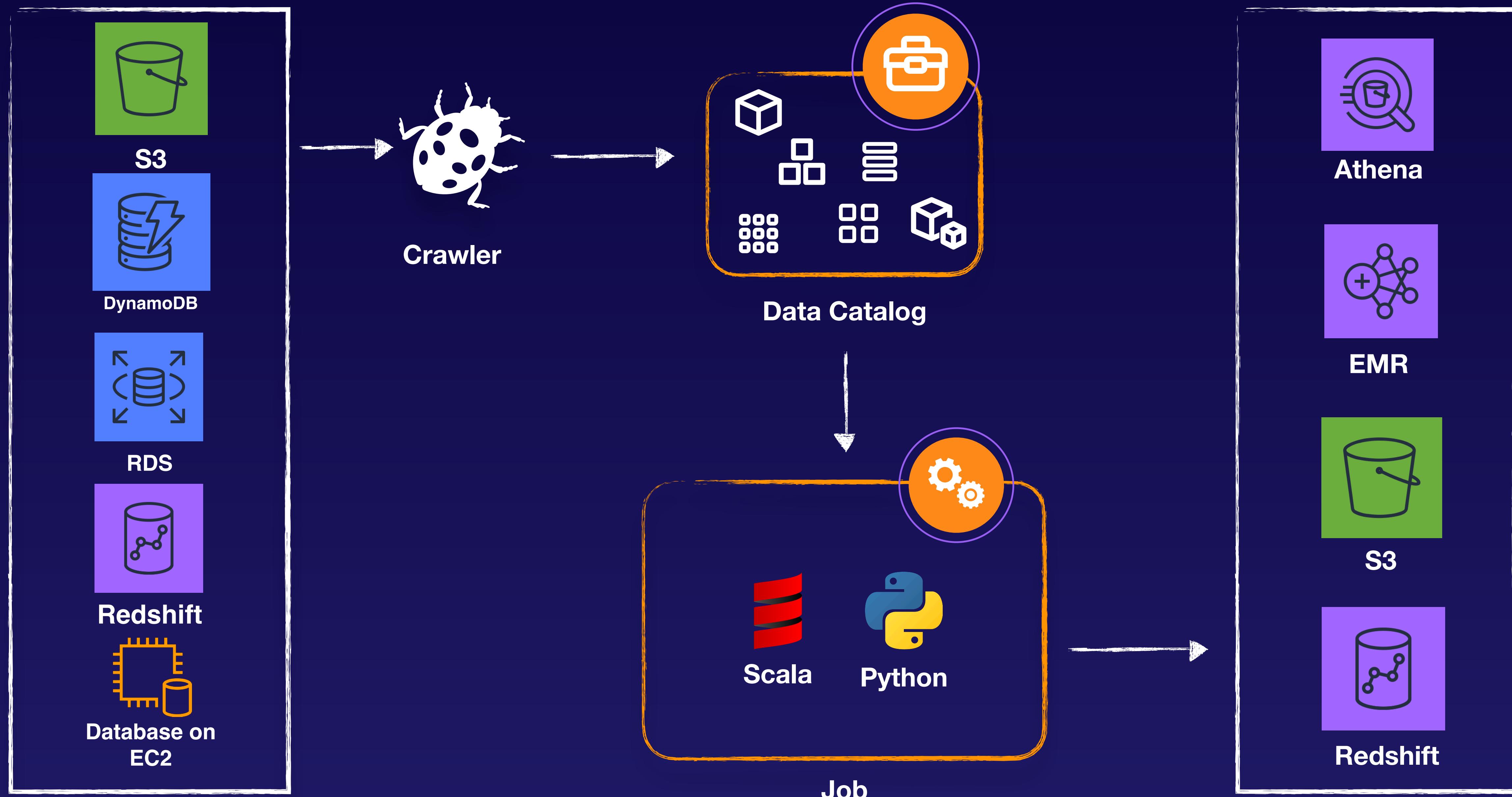


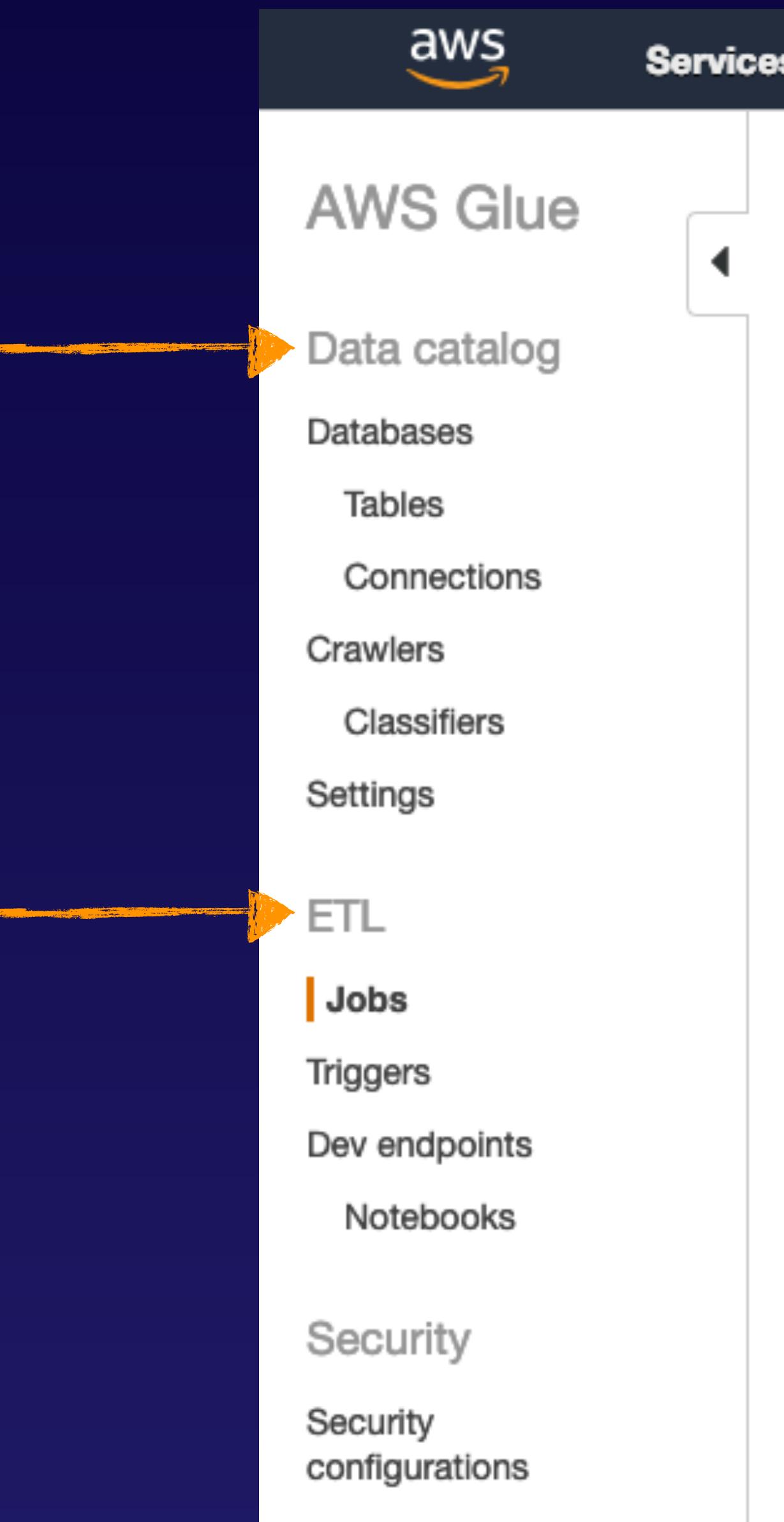


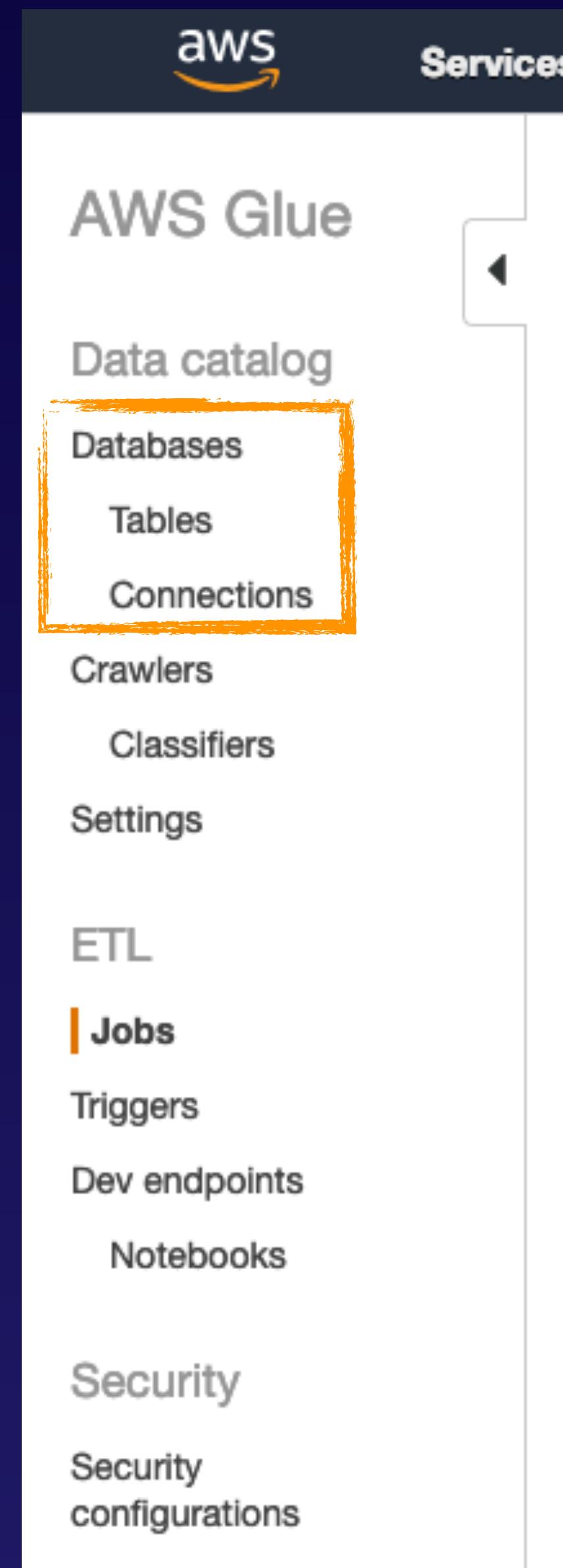


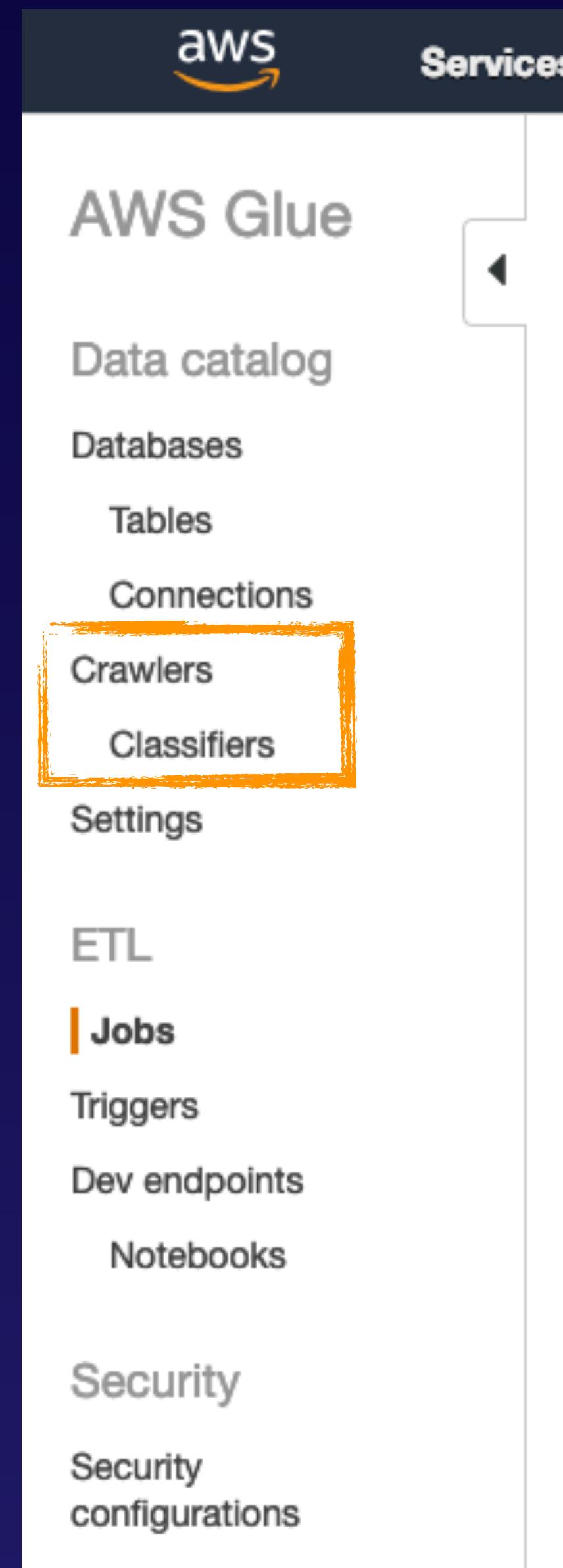


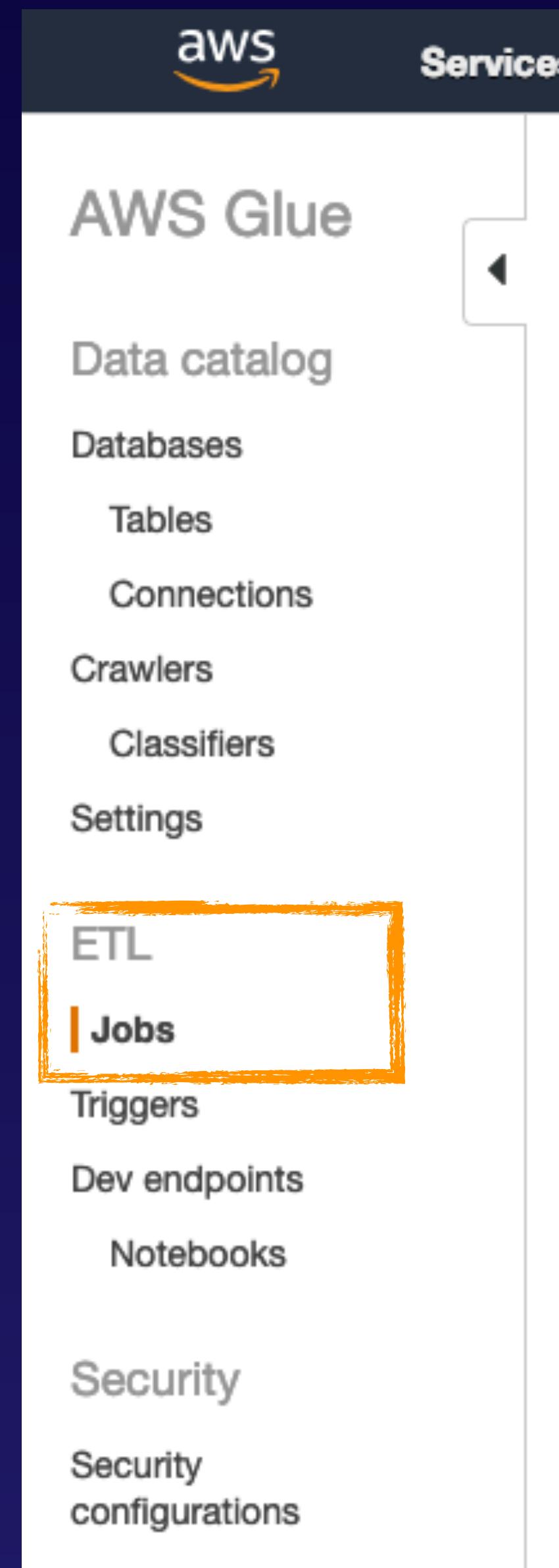
AWS Glue











aws Services Resource Groups ⚙️ 🔍 Brock Tuber N. Virginia Support X

Add job

Configure the job properties

Job properties

Data source

Data target

Schema

Name

IAM role [AWSGlueServiceRole-General-Role](#) 

Ensure that this role has permission to your Amazon S3 sources, targets, temporary directory, scripts, and any libraries used by the job. [Create IAM role](#).

Type

This job runs

A proposed script generated by AWS Glue [i](#)

An existing script that you provide

A new script to be authored by you

ETL language Python Scala

Script file name

S3 path where the script is stored 

Encrypt script using SSE-S3

Temporary directory [i](#) 

 [Feedback](#)  [English \(US\)](#)

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)



The screenshot shows the AWS Glue "Add job" configuration interface. On the left, there's a sidebar with tabs: "Job properties" (selected), "Data source", "Data target", and "Schema". The main area is titled "Configure the job properties".
Name: (Input field)
IAM role: AWSGlueServiceRole-General-Role (Dropdown menu)
Type: Spark (Dropdown menu, highlighted with an orange border)
This job runs: A proposed script generated by AWS Glue (Radio button selected)
ETL language: Python (Radio button selected) Scala (Radio button)
Script file name: scala-job-car-data (Input field)
S3 path where the script is stored: s3://aws-glue-scripts-us-east-1/root (Input field)
Temporary directory: s3://aws-glue-temporar... (Input field)

At the bottom, there are footer links: Feedback, English (US), © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved., Privacy Policy, and Terms of Use.

AWS Glue Jobs



aws Services Resource Groups ⚙️ 🔔 Brock Tuber N. Virginia Support X

Add job

Configure the job properties

Job properties

Job properties

Data source

Data target

Schema

Name

IAM role ⓘ 

Ensure that this role has permission to your Amazon S3 sources, targets, temporary directory, scripts, and any libraries used by the job. [Create IAM role](#).

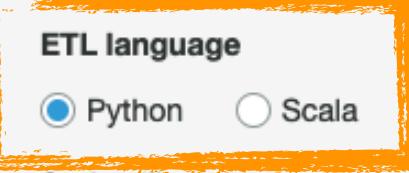
Type ⓘ

This job runs

A proposed script generated by AWS Glue ⓘ

An existing script that you provide

A new script to be authored by you

ETL language 

Python Scala

Script file name

S3 path where the script is stored 

Encrypt script using SSE-S3

Temporary directory ⓘ 

Feedback English (US) © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Glue Jobs



Add job

Configure the job properties

Job properties

Data source

Data target

Schema

Name

IAM role 

Ensure that this role has permission to your Amazon S3 sources, targets, temporary directory, scripts, and any libraries used by the job. [Create IAM role](#).

Type 

This job runs

A proposed script generated by AWS Glue 

An existing script that you provide

A new script to be authored by you

ETL language Python Scala

Script file name

S3 path where the script is stored 

Encrypt script using SSE-S3

Temporary directory 

 [Feedback](#)  [English \(US\)](#)

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)

AWS Glue Jobs

aws Services Resource Groups ⚡

Job: ml-specialty-course-job

Action Save Run job Generate diagram

Insert template at cursor ⓘ Source Target Target Location Transform Spigot X ?

Database Name adult-data-database
Table Name ml_sandbox_demo

Transform Name ApplyMapping

Path s3://ml-python-code-bucket

```
1 import sys
2 from awsglue.transforms import *
3 from awsglue.utils import getResolvedOptions
4 from pyspark.context import SparkContext
5 from awsglue.context import GlueContext
6 from awsglue.job import Job
7
8 ## @params: [JOB_NAME]
9 args = getResolvedOptions(sys.argv, ['JOB_NAME'])
10
11 sc = SparkContext()
12 glueContext = GlueContext(sc)
13 spark = glueContext.spark_session
14 job = Job(glueContext)
15 job.init(args['JOB_NAME'], args)
16 ## @type: DataSource
17 ## @args: [database = "adult-data-database", table_name = "ml_sandbox_demo", transformation_ctx = "datasource0"]
18 ## @return: datasource0
19 ## @inputs: []
20 datasource0 = glueContext.create_dynamic_frame.from_catalog(database = "adult-data-database", table_name = "ml_sandbox_demo", trans-
21 ## @type: ApplyMapping
22 ## @args: [mapping = [("age", "long", "age", "long"), ("workclass", "string", "workclass", "string"), ("fnlwgt", "long", "fnlwgt",
23 ## @return: applymapping1
24 ## @inputs: [frame = datasource0]
25 applymapping1 = ApplyMapping.apply(frame = datasource0, mappings = [("age", "long", "age", "long"), ("workclass", "string", "workcl
26 ## @type: DataSink
27 ## @args: [connection_type = "s3", connection_options = {"path": "s3://ml-python-code-bucket"}, format = "csv", transformation_ctx
28 ## @return: datasink2
29 ## @inputs: [frame = applymapping1]
30 datasink2 = glueContext.write_dynamic_frame.from_options(frame = applymapping1, connection_type = "s3", connection_options = {"path
31 job.commit()
```

Feedback English (US)

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Glue Jobs

aws Services Resource Groups ⚡

Job: ml-specialty-course-job

Action Save Run job Generate diagram Insert template at cursor ⓘ Source Target Target Location Transform Spigot X ?

Database Name adult-data-database
Table Name ml_sandbox_demo

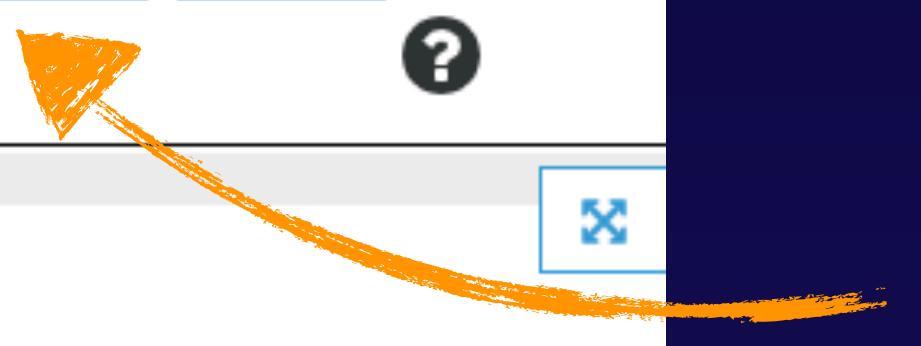
Transform Name ApplyMapping

Path s3://ml-python-code-bucket

```
1 import sys
2 from awsglue.transforms import *
3 from awsglue.utils import getResolvedOptions
4 from pyspark.context import SparkContext
5 from awsglue.context import GlueContext
6 from awsglue.job import Job
7
8 ## @params: [JOB_NAME]
9 args = getResolvedOptions(sys.argv, ['JOB_NAME'])
10
11 sc = SparkContext()
12 glueContext = GlueContext(sc)
13 spark = glueContext.spark_session
14 job = Job(glueContext)
15 job.init(args['JOB_NAME'], args)
16 ## @type: DataSource
17 ## @args: [database = "adult-data-database", table_name = "ml_sandbox_demo", transformation_ctx = "datasource0"]
18 ## @return: datasource0
19 ## @inputs: []
20 datasource0 = glueContext.create_dynamic_frame.from_catalog(database = "adult-data-database", table_name = "ml_sandbox_demo", trans
21 ## @type: ApplyMapping
22 ## @args: [mapping = [("age", "long", "age", "long"), ("workclass", "string", "workclass", "string"), ("fnlwgt", "long", "fnlwgt",
23 ## @return: applymapping1
24 ## @inputs: [frame = datasource0]
25 applymapping1 = ApplyMapping.apply(frame = datasource0, mappings = [("age", "long", "age", "long"), ("workclass", "string", "workcl
26 ## @type: DataSink
27 ## @args: [connection_type = "s3", connection_options = {"path": "s3://ml-python-code-bucket"}, format = "csv", transformation_ctx
28 ## @return: datasink2
29 ## @inputs: [frame = applymapping1]
30 datasink2 = glueContext.write_dynamic_frame.from_options(frame = applymapping1, connection_type = "s3", connection_options = {"path
31 job.commit()
```

Feedback English (US)

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use



AWS Glue PySpark Transforms Reference

AWS Glue has created the following transform Classes to use in PySpark ETL operations.

- [GlueTransform Base Class](#)
- [ApplyMapping Class](#)
- [DropFields Class](#)
- [DropNullFields Class](#)
- [ErrorsAsDynamicFrame Class](#)
- [Filter Class](#)
- [Join Class](#)
- [Map Class](#)
- [MapToCollection Class](#)
- [Relationalize Class](#)
- [RenameField Class](#)
- [ResolveChoice Class](#)
- [SelectFields Class](#)
- [SelectFromCollection Class](#)
- [Spigot Class](#)
- [SplitFields Class](#)
- [SplitRows Class](#)
- [Unbox Class](#)
- [UnnestFrame Class](#)

Output Formats

- | [avro](#)
- [csv](#)
- [ion](#)
- [grokLog](#)
- [json](#)
- [orc](#)
- [parquet](#)
- [xml](#)

APIs in the AWS Glue Scala Library

com.amazonaws.services.glue

The `com.amazonaws.services.glue` package in the AWS Glue Scala library contains the following APIs:

- [ChoiceOption](#)
- [DataSink](#)
- [DataSource trait](#)
- [DynamicFrame](#)
- [DynamicRecord](#)
- [GlueContext](#)
- [MappingSpec](#)
- [ResolveSpec](#)

APIs in the AWS Glue Scala Library

com.amazonaws.services.glue.types

The `com.amazonaws.services.glue.types` package in the AWS Glue Scala library contains the following APIs:

- [ArrayNode](#)
- [BinaryNode](#)
- [BooleanNode](#)
- [ByteNode](#)
- [DateNode](#)
- [DecimalNode](#)
- [DoubleNode](#)
- [DynamicNode](#)
- [FloatNode](#)
- [IntegerNode](#)
- [LongNode](#)
- [MapLikeNode](#)
- [MapNode](#)
- [NullNode](#)
- [ObjectNode](#)
- [ScalarNode](#)
- [ShortNode](#)
- [StringNode](#)
- [TimestampNode](#)

AWS Glue Jobs



aws Services Resource Groups ⚡

Brock Tuber N. Virginia Support

Add job

Job properties

Connections

Configure the job properties

Name

IAM role [AWSGlueServiceRole-General-Role](#)

Ensure that this role has permission to your Amazon S3 sources, targets, temporary directory, scripts, and any libraries used by the job. [Create IAM role](#).

Type Python shell

This job runs

An existing script that you provide
 A new script to be authored by you

S3 path where the script is stored s3://aws-glue-scripts-us-east-1/root

Encrypt script using SSE-S3

► Security configuration, script libraries, and job parameters (optional)

Next

Feedback English (US) © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Glue Jobs



aws Services Resource Groups ⚡

Brock Tuber N. Virginia Support

Add job

Configure the job properties

Job properties

Connections

Name

IAM role [Create IAM role](#)

Ensure that this role has permission to your Amazon S3 sources, targets, temporary directory, scripts, and any libraries used by the job.

Type Python shell

This job runs

An existing script that you provide

A new script to be authored by you

S3 path where the script is stored [Edit](#)

Encrypt script using SSE-S3

► Security configuration, script libraries, and job parameters (optional)

[Next](#)

Feedback English (US) © 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AWS Glue Jobs

AWS Glue Jobs

A job is your business logic required to perform extract, transform and load (ETL) work. Job runs are initiated by triggers which can be scheduled or driven by events.

Add job Action Filter by attributes User preferences Showing: 1 - 1

Name	Type	ETL language	Script location	Last modified	Job bookmark
new-test-job-python-s...	Python shell		s3://aws-glue-scripts-...	6 March 2019 5:43 PM...	Disable

ETL

Jobs

Triggers Dev endpoints Notebooks

Security Security configurations

Tutorials Add crawler Explore table Add job

History Details Script

```
1 import boto3
2 import pandas as pd
3 import io
4 from sklearn.preprocessing import MinMaxScaler
5
6 bucket='ml-sandbox-demo'
7 data_key = 'car_data.csv'
8
9 s3 = boto3.client('s3')
10 obj = s3.get_object(Bucket=bucket, Key=data_key)
11 df = pd.read_csv(io.BytesIO(obj['Body'].read()))
12
13 df.replace({'Make' : {'BMW':'Corvette'}}, inplace=True)
14 df.replace({'Make' : {'Ford':'Mustang'}}, inplace=True)
15 df.replace({'Make' : {'Toyota':'Camero'}}, inplace=True)
16
17 # Clean up any missing data
18 df.drop(columns=['Market Category'], inplace=True)
19 engine_hp_mean = df['Engine HP'].mean()
20
```

Edit script

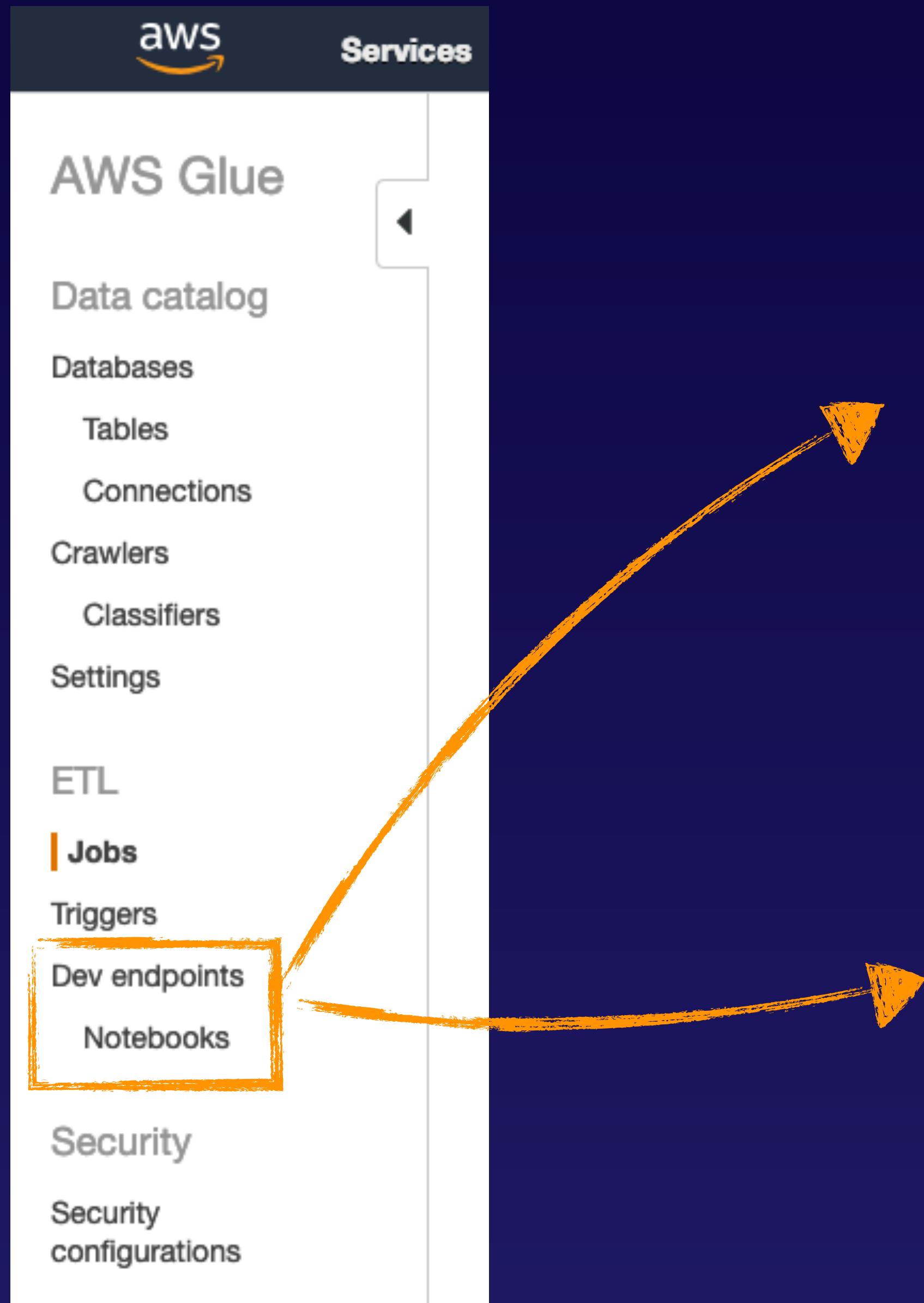
Supported Libraries for Python Shell Jobs

The environment for running a Python shell job supports the following libraries:

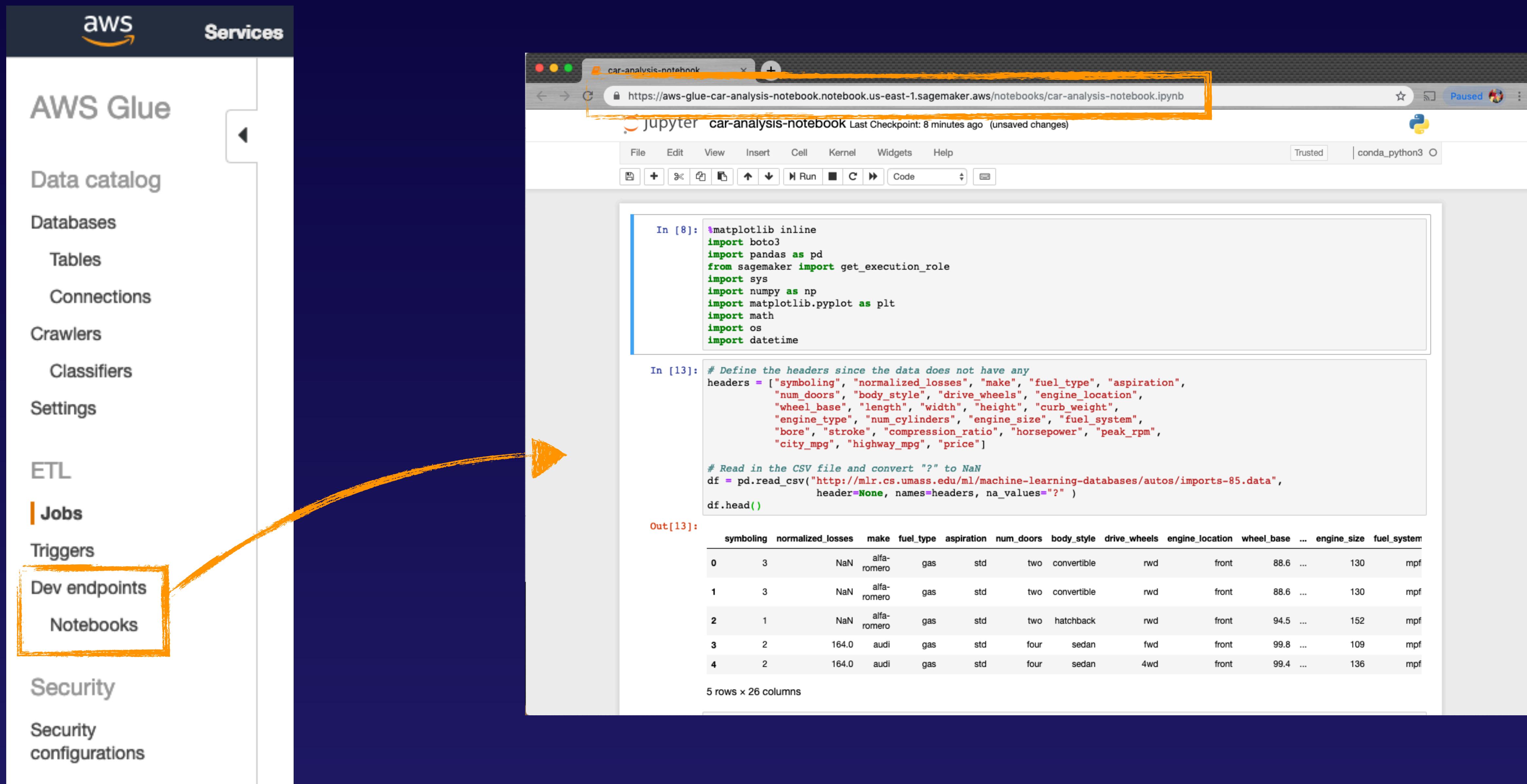
- Boto3
- collections
- CSV
- gzip
- multiprocessing
- NumPy
- pandas
- pickle
- re
- SciPy
- sklearn
- sklearn.feature_extraction
- sklearn.preprocessing
- xml.etree.ElementTree
- zipfile



AWS Glue Notebooks



AWS Glue Notebooks



The diagram illustrates the integration between AWS Glue Jobs and AWS Glue Notebooks. On the left, the AWS Glue console interface is shown, specifically the 'Jobs' section. A red box highlights the 'Dev endpoints' and 'Notebooks' options under the 'Jobs' menu. An orange arrow points from this highlighted area to the right side of the screen, where a Jupyter notebook is displayed.

The Jupyter notebook interface shows a session titled 'car-analysis-notebook.ipynb'. The URL in the browser bar is <https://aws-glue-car-analysis-notebook.notebook.us-east-1.sagemaker.aws/notebooks/car-analysis-notebook.ipynb>. The notebook contains Python code for data analysis:

```
In [8]: %matplotlib inline
import boto3
import pandas as pd
from sagemaker import get_execution_role
import sys
import numpy as np
import matplotlib.pyplot as plt
import math
import os
import datetime

In [13]: # Define the headers since the data does not have any
headers = ["symboling", "normalized_losses", "make", "fuel_type", "aspiration",
           "num_doors", "body_style", "drive_wheels", "engine_location",
           "wheel_base", "length", "width", "height", "curb_weight",
           "engine_type", "num_cylinders", "engine_size", "fuel_system",
           "bore", "stroke", "compression_ratio", "horsepower", "peak_rpm",
           "city_mpg", "highway_mpg", "price"]

# Read in the CSV file and convert "?" to NaN
df = pd.read_csv("http://mlr.cs.umass.edu/ml/machine-learning-databases/autos/imports-85.data",
                 header=None, names=headers, na_values='?')
df.head()

Out[13]:
```

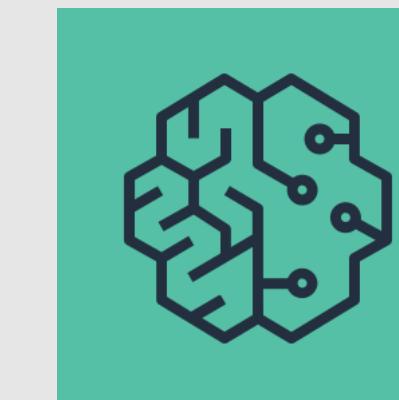
	symboling	normalized_losses	make	fuel_type	aspiration	num_doors	body_style	drive_wheels	engine_location	wheel_base	...	engine_size	fuel_system
0	3	NaN	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpf
1	3	NaN	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpf
2	1	NaN	alfa-romero	gas	std	two	hatchback	rwd	front	94.5	...	152	mpf
3	2	164.0	audi	gas	std	four	sedan	fwd	front	99.8	...	109	mpf
4	2	164.0	audi	gas	std	four	sedan	4wd	front	99.4	...	136	mpf

5 rows x 26 columns

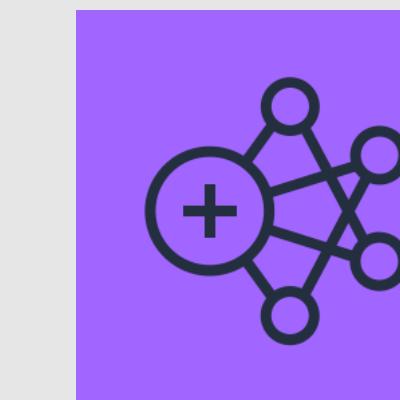
Data Preparation



AWS Glue



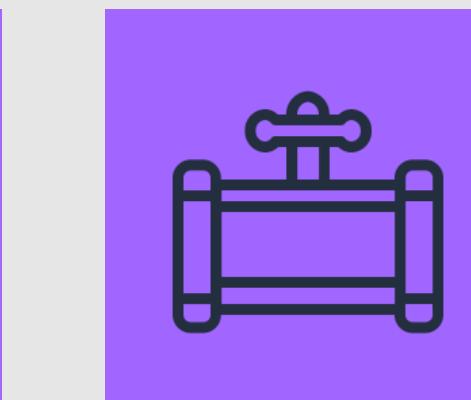
SageMaker



EMR



Athena



Data Pipeline



SageMaker Jupyter Notebooks

AWS Services Resource Groups ⚡

Brock Tubre N. Virginia Support

Amazon SageMaker

Dashboard Search Beta

Ground Truth

- Labeling jobs
- Labeling datasets
- Labeling workforces

Notebook

- Notebook instances**
- Lifecycle configurations
- Git repositories

Training

- Algorithms
- Training jobs
- Hyperparameter tuning jobs

Inference

- Compilation jobs
- Model packages
- Models
- Endpoint configurations
- Endpoints
- Batch transform jobs

AWS Marketplace

Overview

Ground Truth

Set up and manage labeling jobs for highly accurate training datasets using active learning and human labeling.

Labeling Jobs

Notebook

Availability of AWS and SageMaker SDKs and sample notebooks to create training Jobs and deploy models.

Notebook instances

Training

Train and tune models at any scale. Leverage high performance AWS algorithms or bring your own.

Training jobs

Hyperparameter tuning jobs

Inference

Create models from training jobs or import external models for hosting to run inferences on new data.

Models

Endpoints

Batch transform jobs

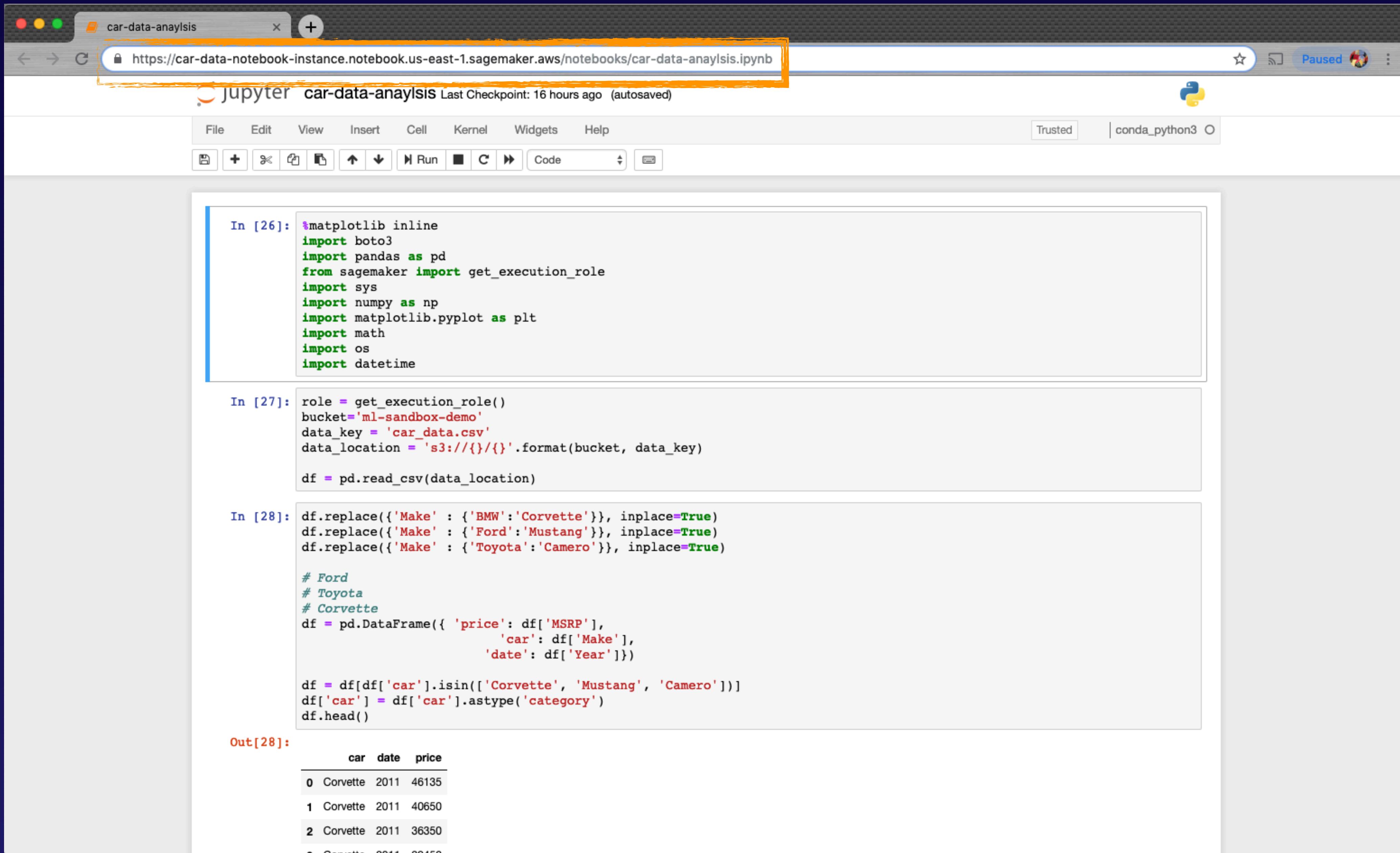
No recent activity.

Recent activity within the Last 7 days

Feedback English (US)

© 2008 - 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

SageMaker Jupyter Notebooks



The screenshot shows a Jupyter Notebook interface running in a web browser. The title bar indicates the notebook is titled "car-data-analysis.ipynb". The browser address bar shows the URL: <https://car-data-notebook-instance.notebook.us-east-1.sagemaker.aws/notebooks/car-data-analysis.ipynb>. The notebook has three cells:

- In [26]:**

```
%matplotlib inline
import boto3
import pandas as pd
from sagemaker import get_execution_role
import sys
import numpy as np
import matplotlib.pyplot as plt
import math
import os
import datetime
```
- In [27]:**

```
role = get_execution_role()
bucket='ml-sandbox-demo'
data_key = 'car_data.csv'
data_location = 's3://{}{}'.format(bucket, data_key)

df = pd.read_csv(data_location)
```
- In [28]:**

```
df.replace({'Make' : {'BMW':'Corvette'}}, inplace=True)
df.replace({'Make' : {'Ford':'Mustang'}}, inplace=True)
df.replace({'Make' : {'Toyota':'Camaro'}}, inplace=True)

# Ford
# Toyota
# Corvette
df = pd.DataFrame({ 'price': df['MSRP'],
                     'car': df['Make'],
                     'date': df['Year']})

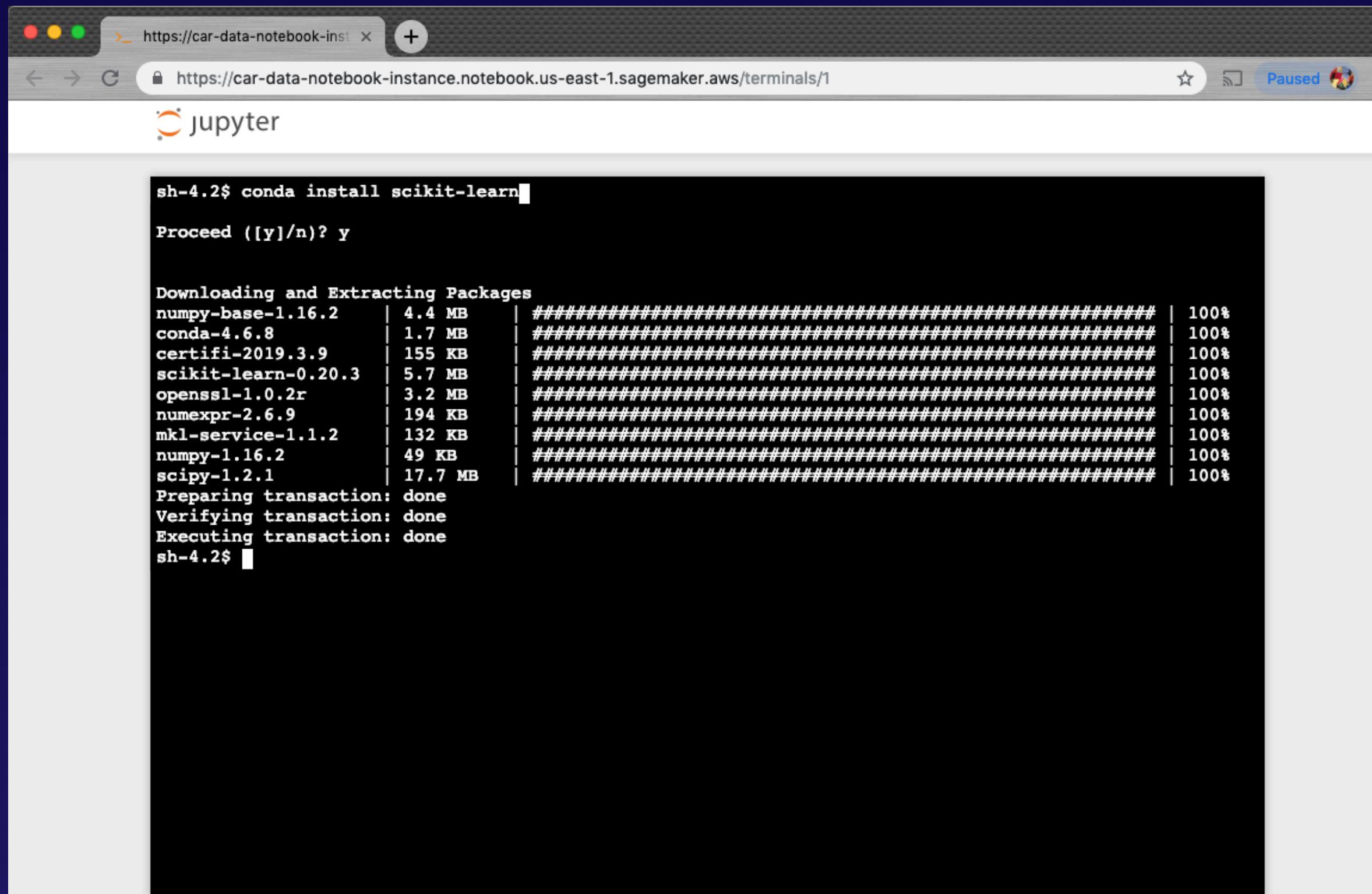
df = df[df['car'].isin(['Corvette', 'Mustang', 'Camaro'])]
df['car'] = df['car'].astype('category')
df.head()
```

Out[28]:

	car	date	price
0	Corvette	2011	46135
1	Corvette	2011	40650
2	Corvette	2011	36350
3	Corvette	2011	29450

SageMaker Jupyter Notebooks

conda

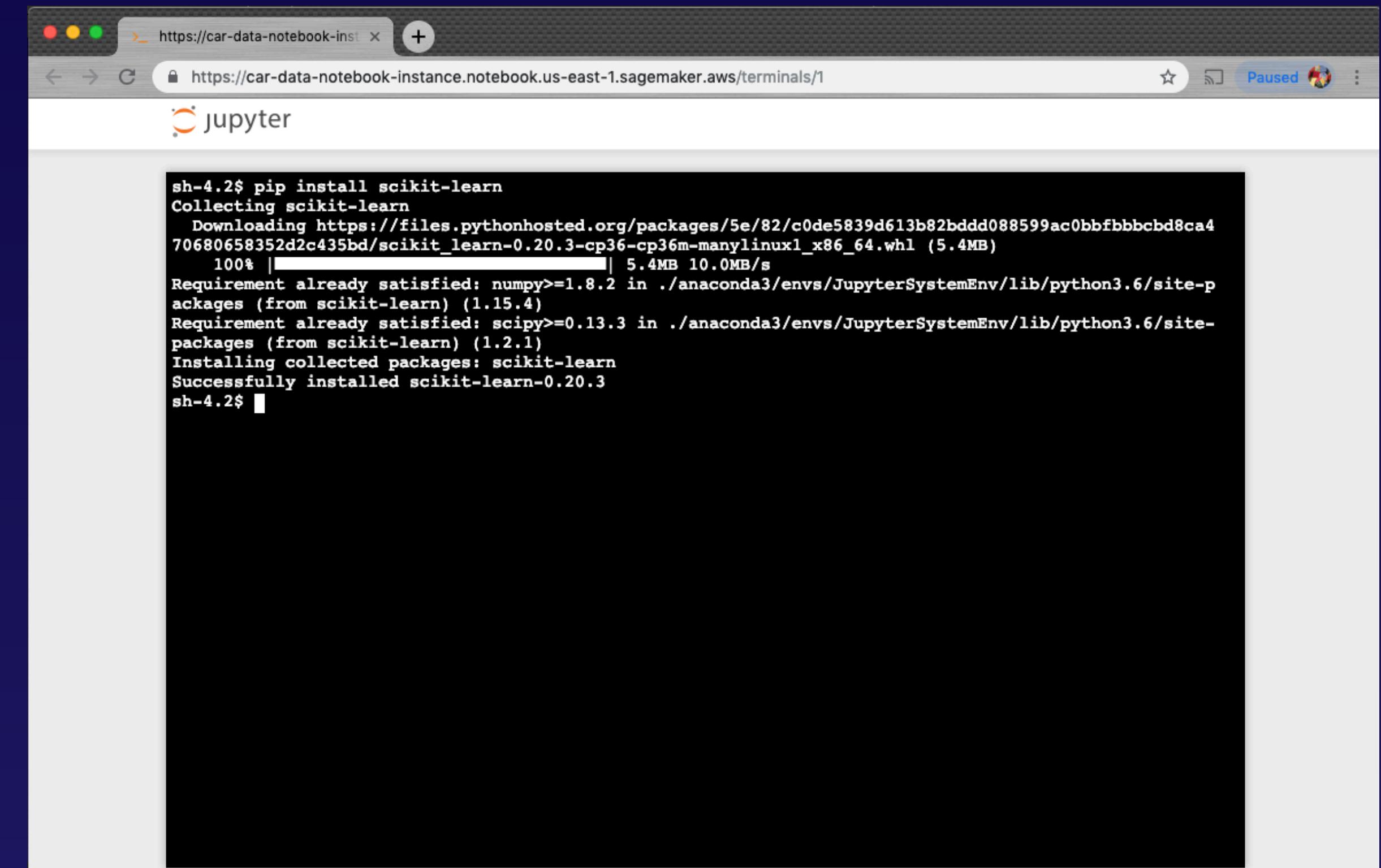


```
sh-4.2$ conda install scikit-learn
Proceed ([y]/n)? y

Downloading and Extracting Packages
numpy-base-1.16.2      | 4.4 MB    | #####| 100%
conda-4.8               | 1.7 MB    | #####| 100%
certifi-2019.3.9        | 155 KB   | #####| 100%
scikit-learn-0.20.3     | 5.7 MB    | #####| 100%
openssl-1.0.2r          | 3.2 MB    | #####| 100%
numexpr-2.6.9            | 194 KB   | #####| 100%
mkl-service-1.1.2       | 132 KB   | #####| 100%
numpy-1.16.2             | 49 KB    | #####| 100%
scipy-1.2.1              | 17.7 MB   | #####| 100%

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
sh-4.2$
```

pip

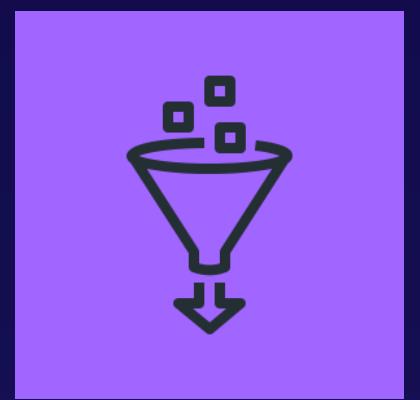


```
sh-4.2$ pip install scikit-learn
Collecting scikit-learn
  Downloading https://files.pythonhosted.org/packages/5e/82/c0de5839d613b82bdd088599ac0bbfbcb8ca470680658352d2c435bd/scikit_learn-0.20.3-cp36-cp36m-manylinux1_x86_64.whl (5.4MB)
    100% |████████████████████████████████| 5.4MB 10.0MB/s
Requirement already satisfied: numpy>=1.8.2 in ./anaconda3/envs/JupyterSystemEnv/lib/python3.6/site-packages (from scikit-learn) (1.15.4)
Requirement already satisfied: scipy>=0.13.3 in ./anaconda3/envs/JupyterSystemEnv/lib/python3.6/site-packages (from scikit-learn) (1.2.1)
Installing collected packages: scikit-learn
Successfully installed scikit-learn-0.20.3
sh-4.2$
```

Using AWS Glue or SageMaker



SageMaker & Jupyter
Notebooks



ETL jobs in AWS Glue

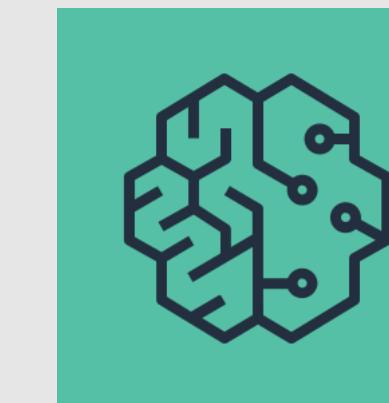


Elastic Map Reduce (EMR)

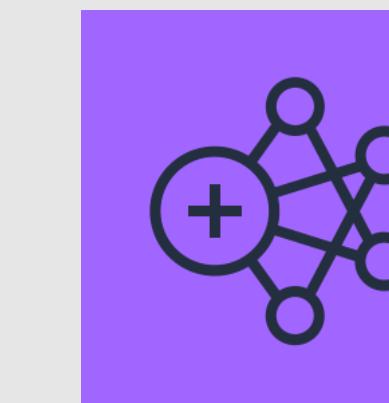
Data Preparation



AWS Glue



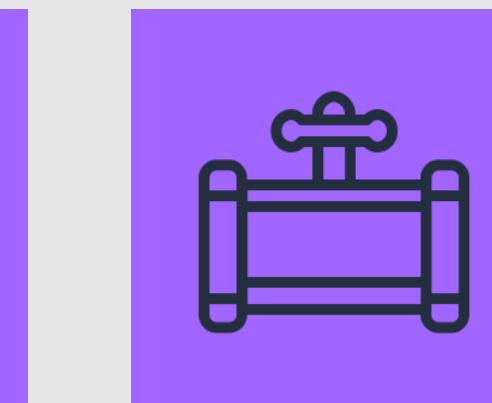
SageMaker



EMR



Athena



Data Pipeline



Elastic Map Reduce (EMR)



APACHE
PIG

presto 

mxnet



MAHOUT



APACHE
HBASE 



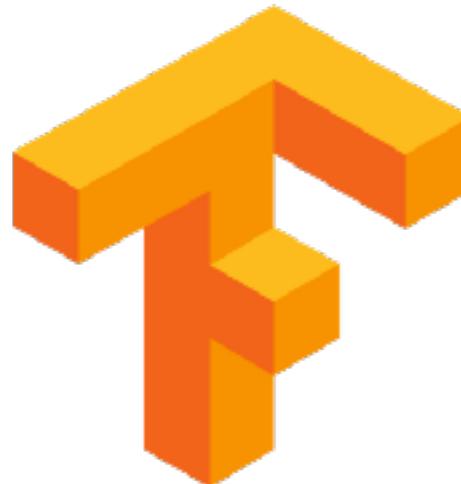
APACHE
Spark 



jupyter



APACHE
Zookeeper



Elastic Map Reduce (EMR)



Spark ML and MLib
ETL and Machine
Learning Library



Presto
SQL Query
Engine



MAHOUT

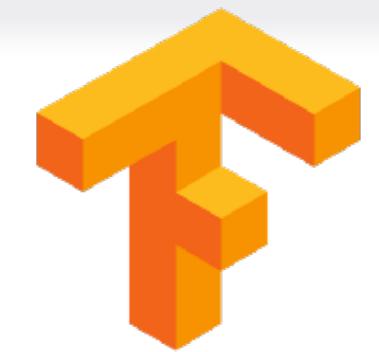
Mahout
Machine Learning
Framework



Hive
ETL Service



Jupyter Notebooks
Code Sharing



TensorFlow
Machine Learning
Framework



Hadoop Distributed File
System
Persistant Datastore



MXNet
Machine Learning
Framework

Elastic Map Reduce (EMR)



Spark ML and MLib
ETL and Machine
Learning Library



Presto
SQL Query
Engine



MAHOUT

Mahout
Machine Learning
Framework



Hive
ETL Service



Jupyter Notebooks
Code Sharing



TensorFlow
Machine Learning
Framework



Hadoop Distributed File
System
Persistant Datastore



MXNet
Machine Learning
Framework

Elastic Map Reduce (EMR)



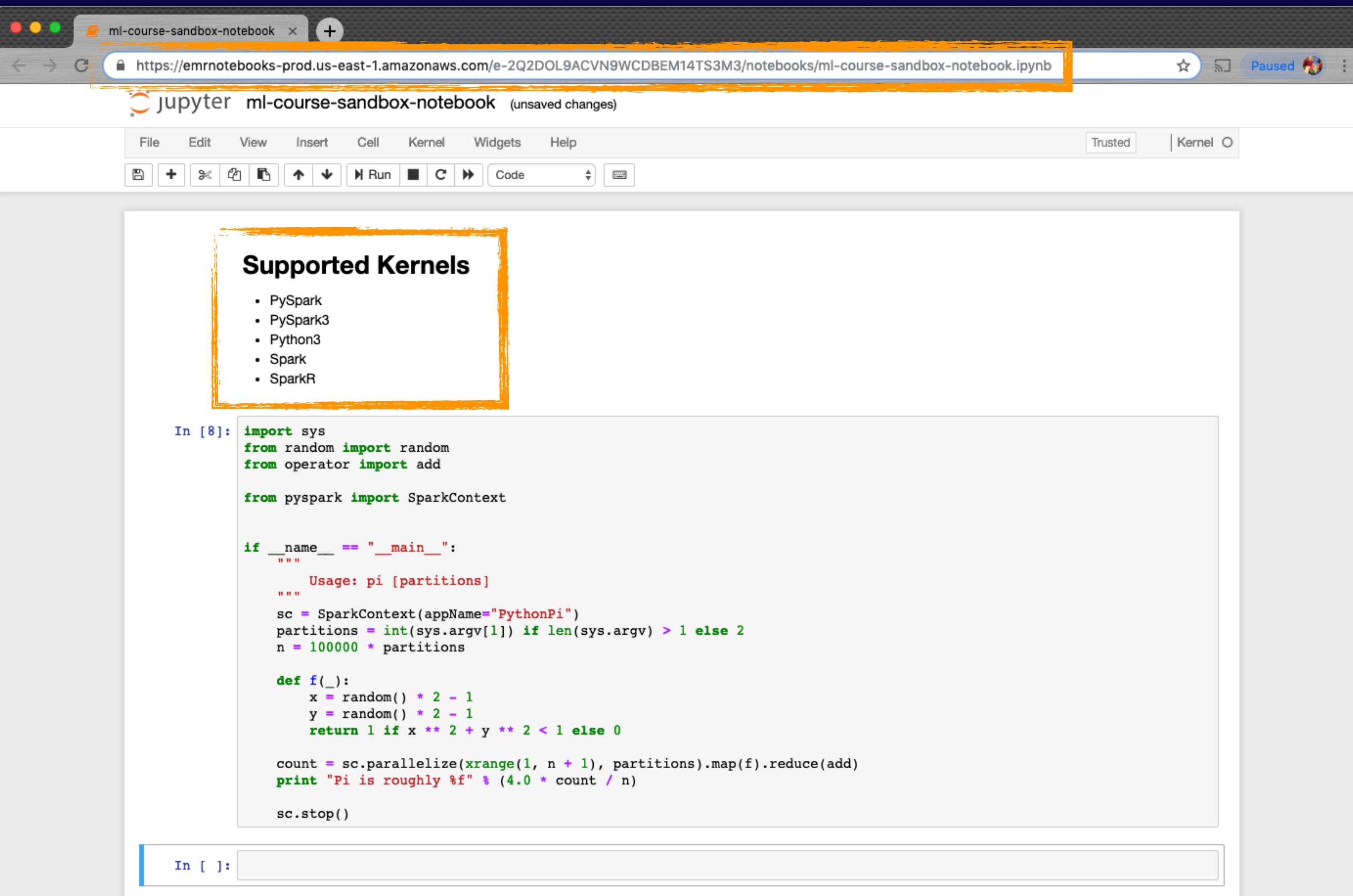
Create Cluster - Advanced Options

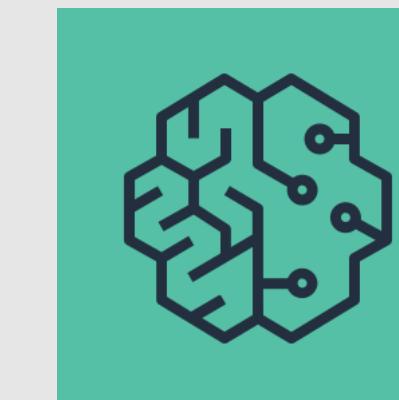
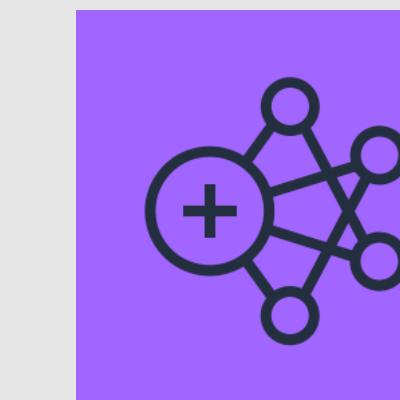
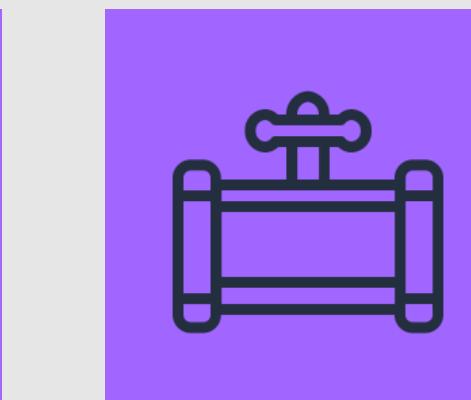
Software Configuration

Release emr-5.20.0

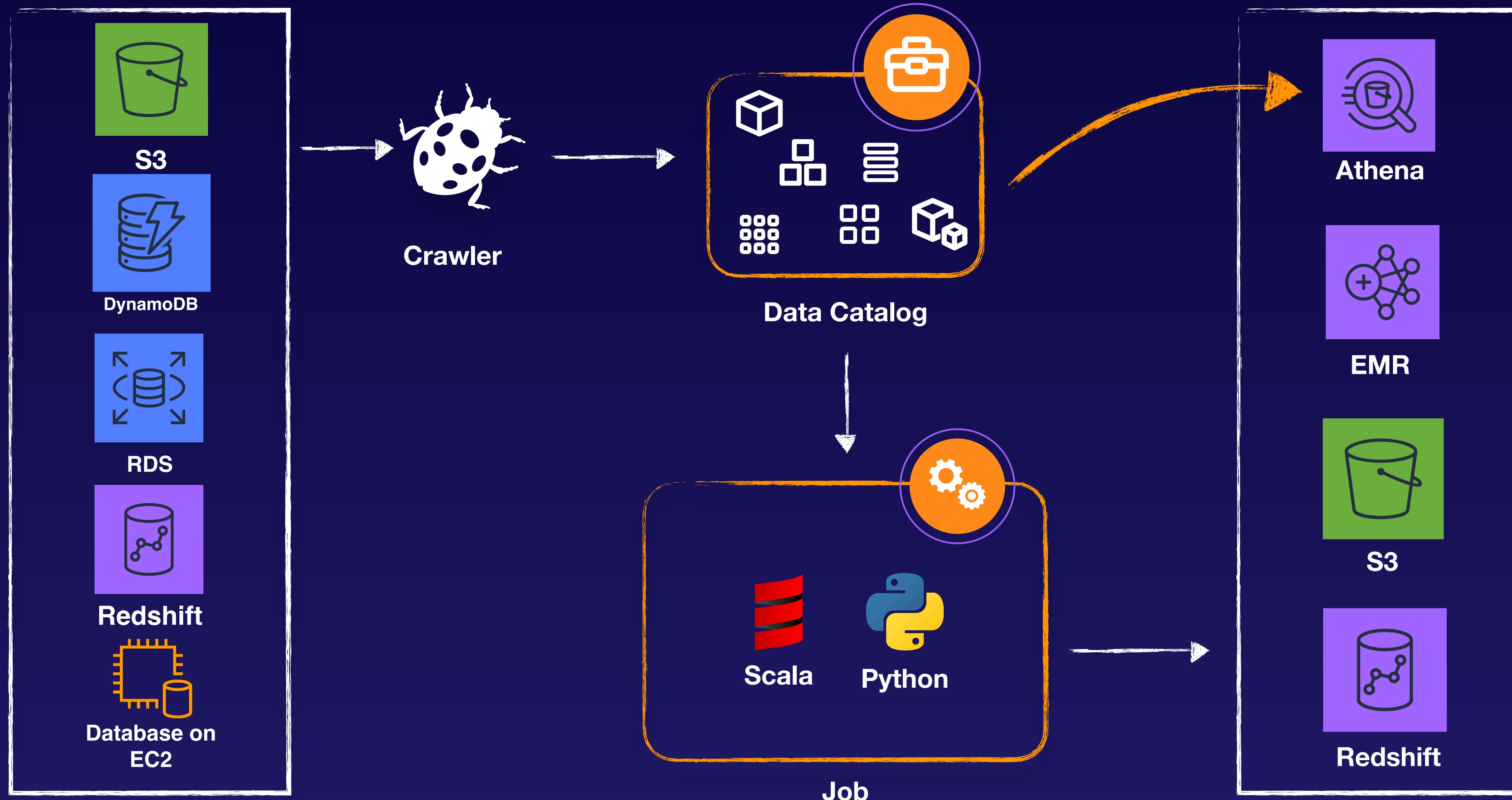
- | | | |
|--|--|---|
| <input checked="" type="checkbox"/> Hadoop 2.8.5 | <input type="checkbox"/> Zeppelin 0.8.0 | <input type="checkbox"/> Livy 0.5.0 |
| <input checked="" type="checkbox"/> JupyterHub 0.9.4 | <input type="checkbox"/> Tez 0.9.1 | <input type="checkbox"/> Flink 1.6.2 |
| <input type="checkbox"/> Ganglia 3.7.2 | <input type="checkbox"/> HBase 1.4.8 | <input type="checkbox"/> Pig 0.17.0 |
| <input checked="" type="checkbox"/> Hive 2.3.4 | <input checked="" type="checkbox"/> Presto 0.214 | <input type="checkbox"/> ZooKeeper 3.4.13 |
| <input checked="" type="checkbox"/> MXNet 1.3.1 | <input type="checkbox"/> Sqoop 1.4.7 | <input checked="" type="checkbox"/> Mahout 0.13.0 |
| <input type="checkbox"/> Hue 4.3.0 | <input type="checkbox"/> Phoenix 4.14.0 | <input type="checkbox"/> Oozie 5.0.0 |
| <input checked="" type="checkbox"/> Spark 2.4.0 | <input type="checkbox"/> HCatalog 2.3.4 | <input checked="" type="checkbox"/> TensorFlow 1.12.0 |

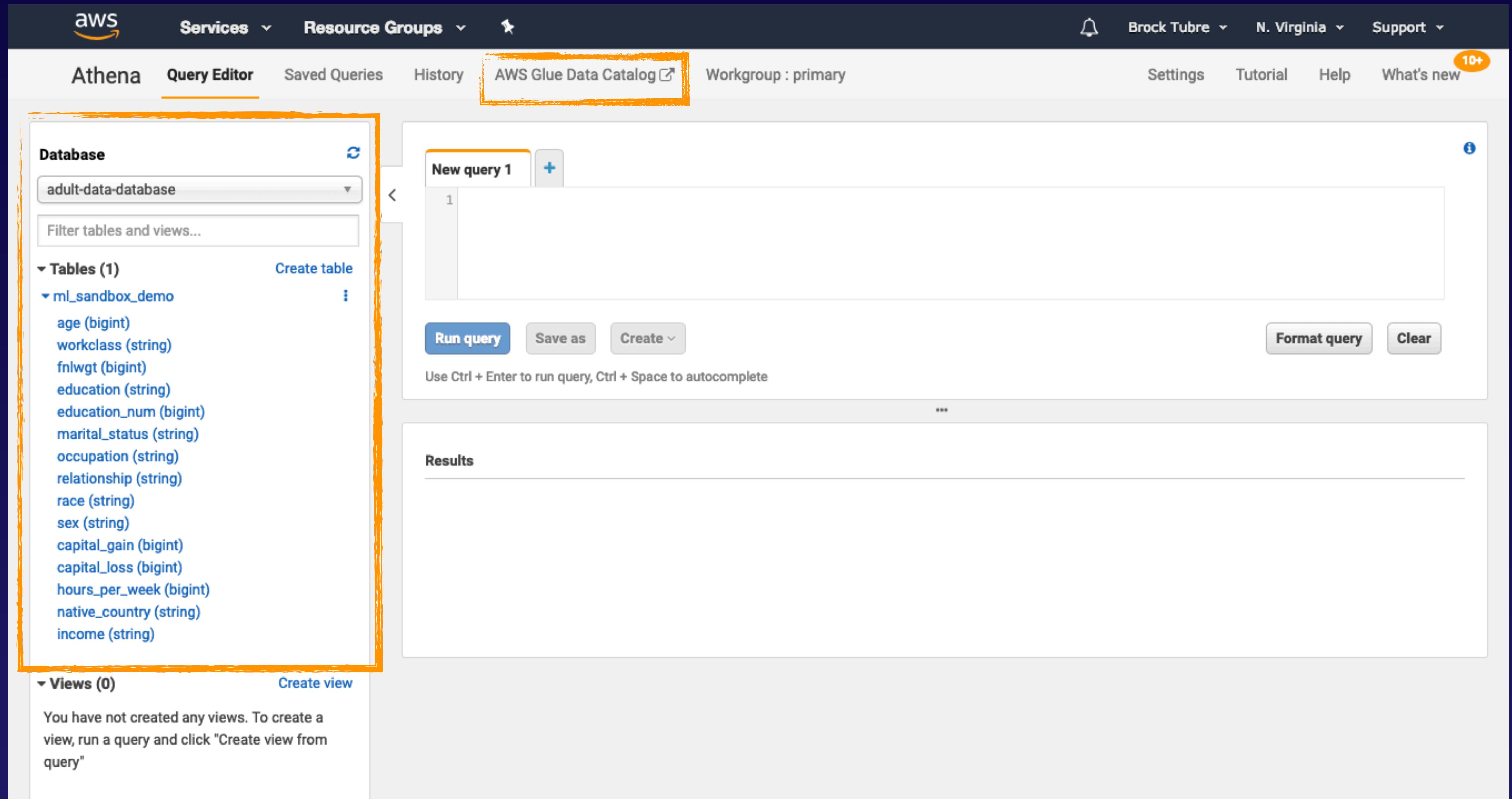
Elastic Map Reduce (EMR) Notebooks



Data Preparation**AWS Glue****SageMaker****EMR****Athena****Data Pipeline**

AWS Glue





The screenshot displays the AWS Athena Query Editor interface. On the left, a sidebar provides access to the database and table structures:

- Database:** adult-data-database (selected)
- Tables (1):** ml_sandbox_demo (expanded)
 - age (bigint)
 - workclass (string)
 - fnlwgt (bigint)
 - education (string)
 - education_num (bigint)
 - marital_status (string)
 - occupation (string)
 - relationship (string)
 - race (string)
 - sex (string)
 - capital_gain (bigint)
 - capital_loss (bigint)
 - hours_per_week (bigint)
 - native_country (string)
 - income (string)
- Views (0):** You have not created any views. To create a view, run a query and click "Create view from query".

The main workspace contains a query editor titled "New query 1" with the following content:

```
New query 1
1
```

Buttons at the bottom of the editor include: Run query, Save as, Create, Format query, and Clear.

At the bottom of the interface, a note states: "Use Ctrl + Enter to run query, Ctrl + Space to autocomplete".

Screenshot of the AWS Athena Query Editor interface.

The interface includes the following components:

- Top Bar:** AWS logo, Services dropdown, Resource Groups dropdown, Notifications icon, Brock Tuber (User), N. Virginia (Region), Support dropdown, and a notification badge (10+).
- Left Sidebar (Database Panel):**
 - Database:** adult-data-database (selected).
 - Filter tables and views...
 - Tables (1):** ml_sandbox_demo
 - age (bigint)
 - workclass (string)
 - fnlwgt (bigint)
 - education (string)
 - education_num (bigint)
 - marital_status (string)
 - occupation (string)
 - relationship (string)
 - race (string)
 - sex (string)
 - capital_gain (bigint)
 - capital_loss (bigint)
 - hours_per_week (bigint)
 - native_country (string)
 - income (string)
 - Views (0):** Create view
- Central Area:**
 - New query 1** (highlighted with a yellow arrow pointing from the top left).
 - Run query
 - Save as
 - Create
 - Results** (empty area).
- Bottom Bar:** Use Ctrl + Enter to run query, Ctrl + Space to autocomplete.

AWS Services Resource Groups ⚡ Brock Tuber N. Virginia Support 10+

Athena Query Editor Saved Queries History AWS Glue Data Catalog Workgroup : primary Settings Tutorial Help What's new

Database
adult-data-database
Filter tables and views...

Tables (1)
ml_sandbox_demo
age (bigint)
workclass (string)
fnlwgt (bigint)
education (string)
education_num (bigint)
marital_status (string)
occupation (string)
relationship (string)
race (string)
sex (string)
capital_gain (bigint)
capital_loss (bigint)
hours_per_week (bigint)
native_country (string)
income (string)

Create table ⋮

New query 1 +
1 SELECT * FROM ml_sandbox_demo
2 WHERE sex = 'Male'
3 AND age > 39 limit 10;

Run query **Save as** **Create** (Run time: 1.68 seconds, Data scanned: 1.12 MB) **Format query** **Clear**

Use Ctrl + Enter to run query, Ctrl + Space to autocomplete

Results ⌂ ↗

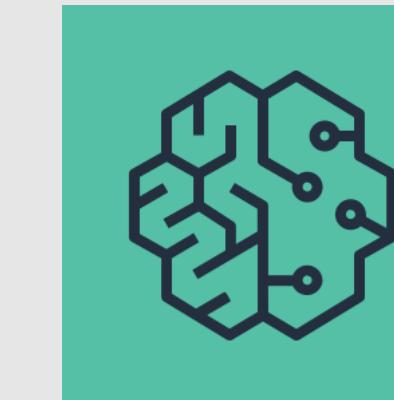
	age	workclass	fnlwgt	education	education_num	marital_status	occupation	relationship	race	sex	capital
1	50	Self-emp-not-inc		Bachelors		Married-civ-spouse	Exec-managerial	Husband	White		Male
2	53	Private		11th		Married-civ-spouse	Handlers-cleaners	Husband	Black		Male
3	52	Self-emp-not-inc		HS-grad		Married-civ-spouse	Exec-managerial	Husband	White		Male
4	42	Private		Bachelors		Married-civ-spouse	Exec-managerial	Husband	White		Male
5	40	Private		Assoc-voc		Married-civ-spouse	Craft-repair	Husband	Asian-Pac-Islander		Male
6	40	Private		Doctorate		Married-civ-spouse	Prof-specialty	Husband	White		Male
7	43	Private		11th		Married-civ-spouse	Transport-moving	Husband	White		Male
8	56	Local-gov		Bachelors		Married-civ-spouse	Tech-support	Husband	White		Male
9	54	?		Some-college		Married-civ-spouse	?	Husband	Asian-Pac-Islander		Male

Data Pipeline

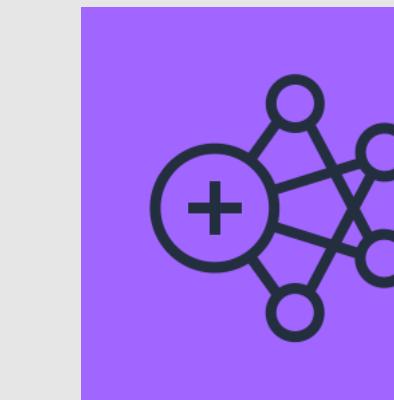
Data Preparation



AWS Glue



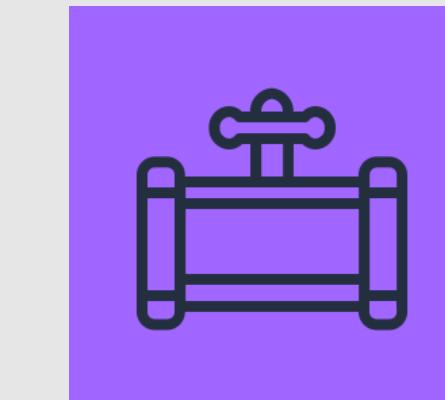
SageMaker



EMR



Athena



Data Pipeline





DynamoDB

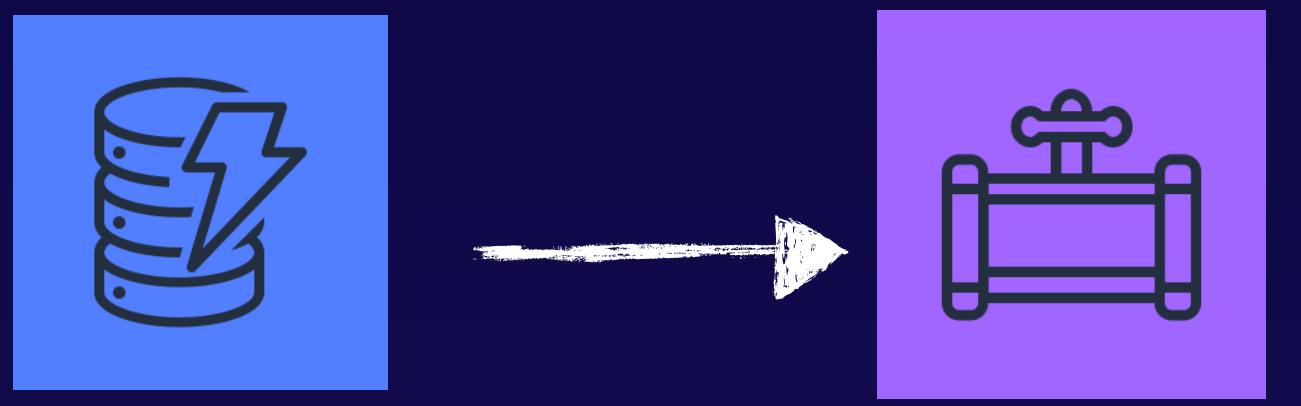


RDS



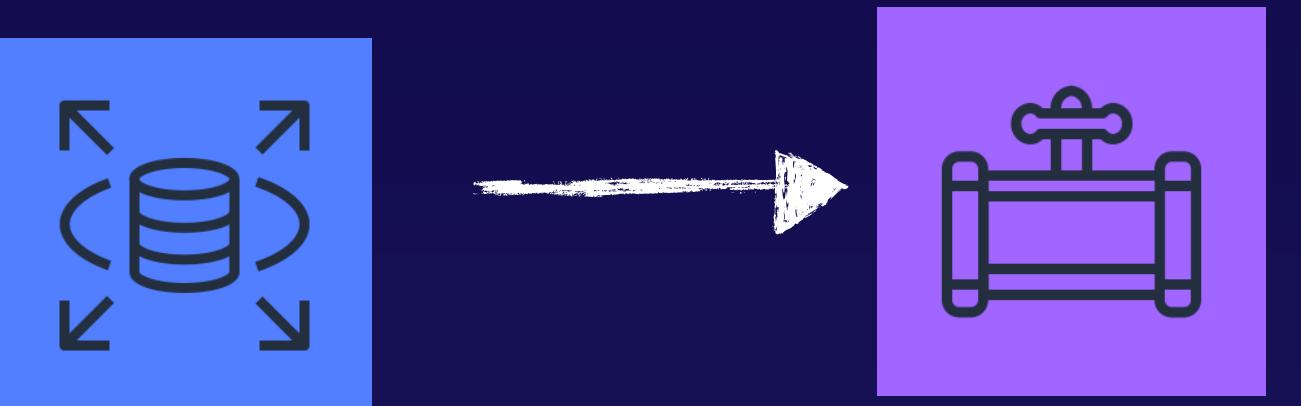
Redshift

Data Pipeline



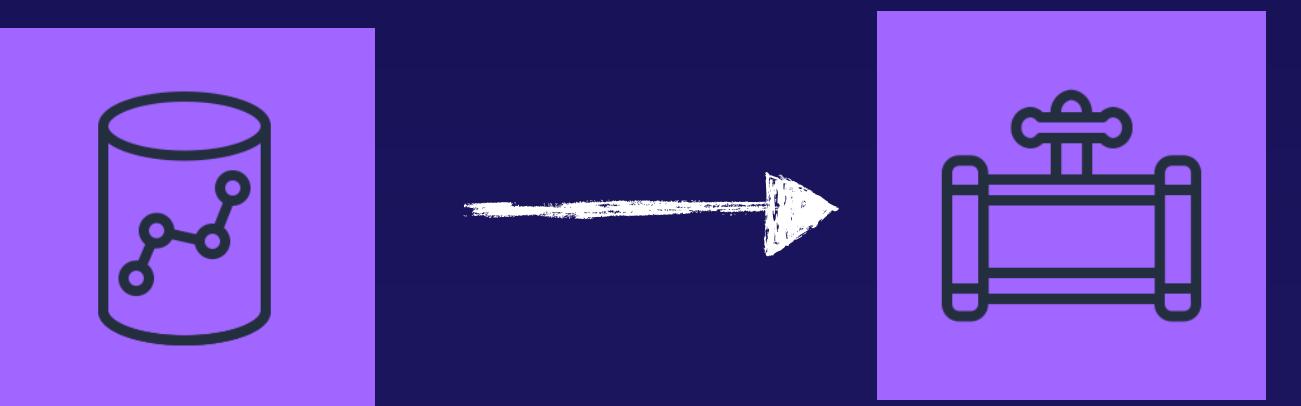
DynamoDB

Data Pipeline



RDS

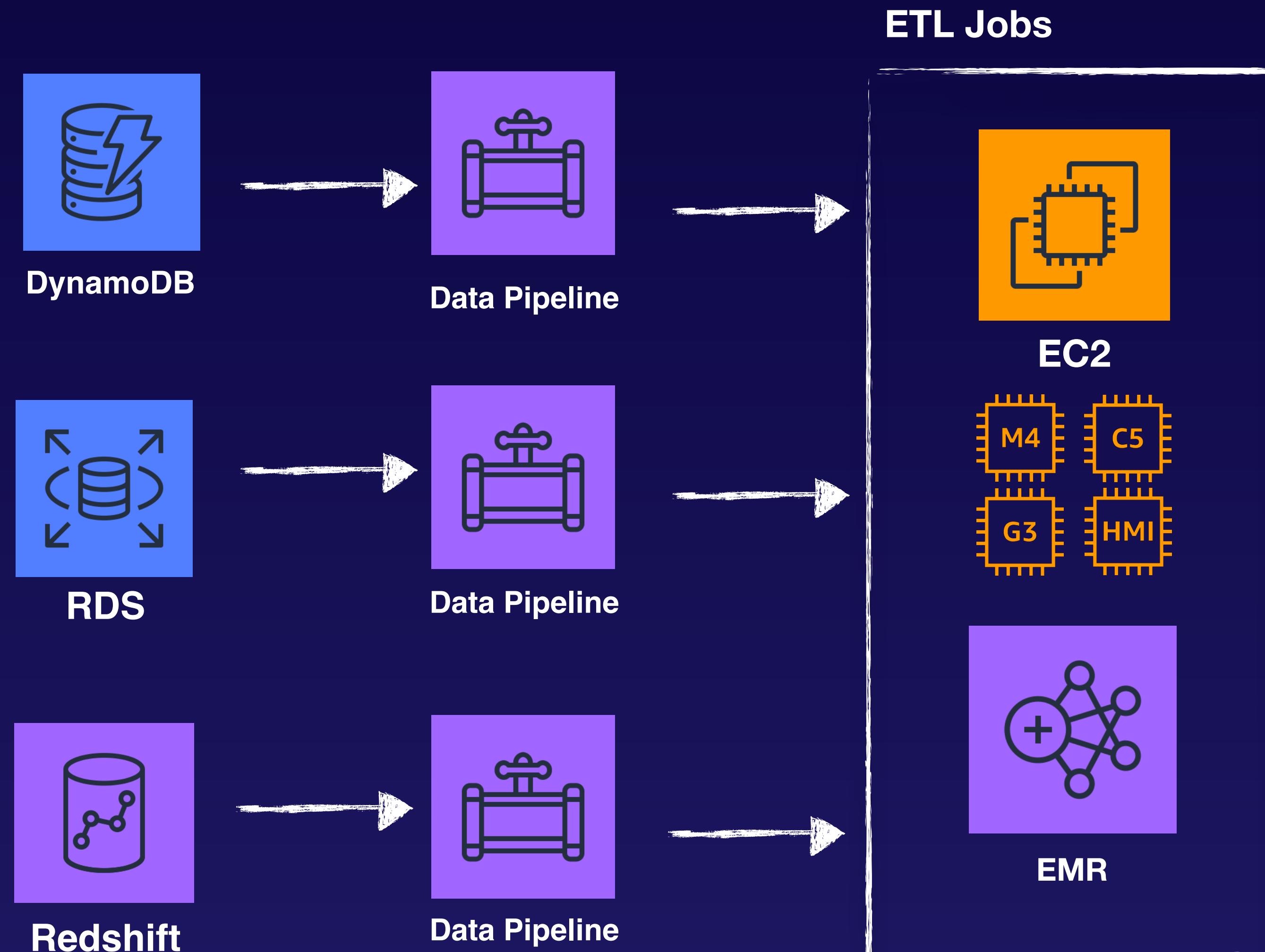
Data Pipeline



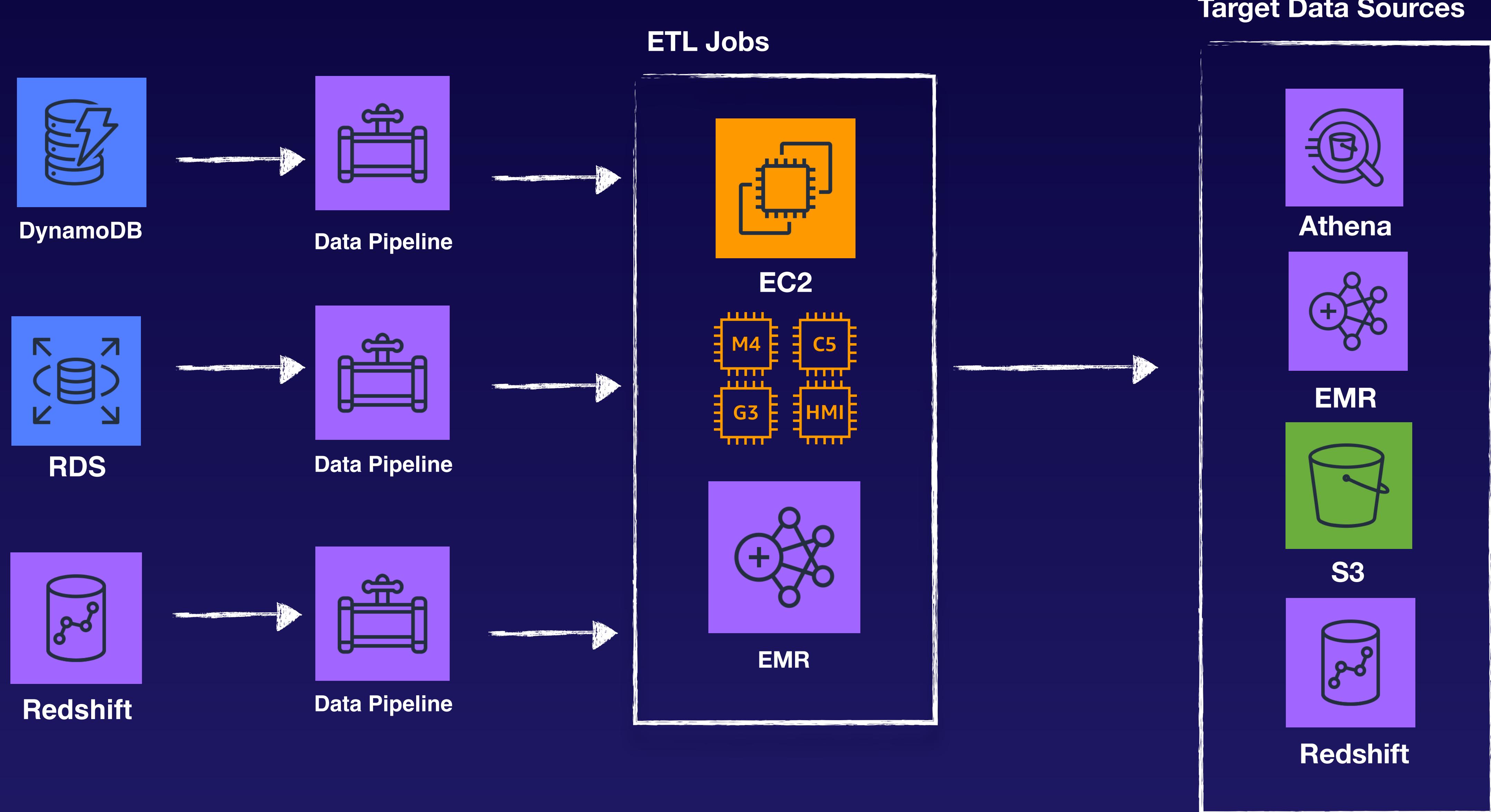
Redshift

Data Pipeline

Data Pipeline



Data Pipeline



AWS Services Resource Groups ⚡

Bell icon Brock Tubre N. Virginia Support

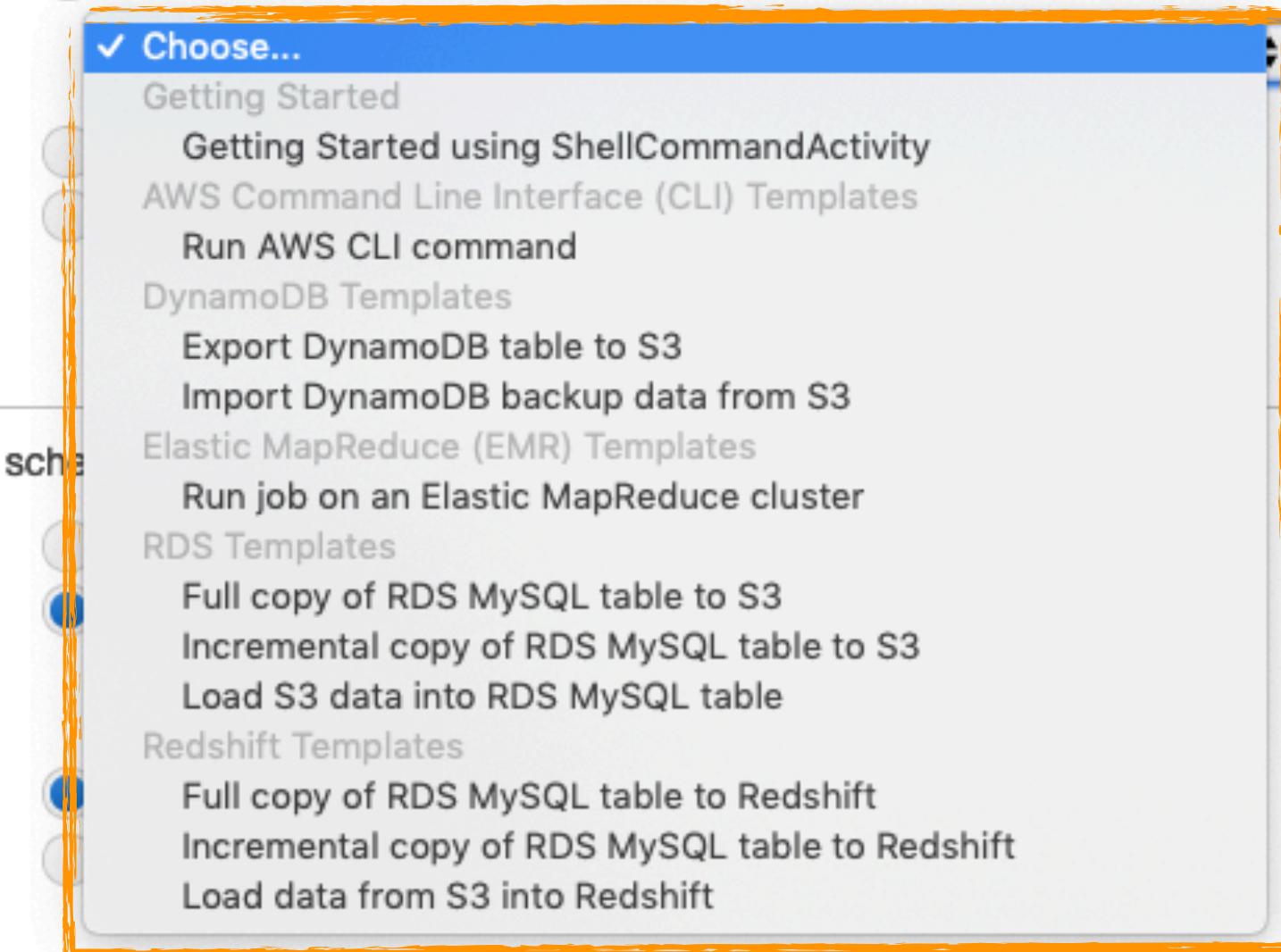
Data Pipeline Create Pipeline

Create Pipeline

Info You can create pipeline using a template or build one using the Architect page.

Name ml-specialty-cert-pipeline

Description (optional)

Source Build using a template Choose... 

- ✓ Choose...
- Getting Started
- Getting Started using ShellCommandActivity
- AWS Command Line Interface (CLI) Templates
- Run AWS CLI command
- DynamoDB Templates
- Export DynamoDB table to S3
- Import DynamoDB backup data from S3
- Elastic MapReduce (EMR) Templates
- Run job on an Elastic MapReduce cluster
- RDS Templates
- Full copy of RDS MySQL table to S3
- Incremental copy of RDS MySQL table to S3
- Load S3 data into RDS MySQL table
- Redshift Templates
- Full copy of RDS MySQL table to Redshift
- Incremental copy of RDS MySQL table to Redshift
- Load data from S3 into Redshift

Schedule

Info You can run your pipeline once or specify a schedule.

Run Once Every Never

Run every 1 minute 5 minutes 1 hour 1 day 1 week 1 month 1 year

Starting 2019-03-04 20:33 UTC Now

Ending never after occurrence(s) on at UTC (Current time is 20:34 UTC)

Which service to use?

Datasource	Data Preparation Tool	Why
S3, Redshift, RDS, DynamoDB, On Premise DB	AWS Glue	Use Python or Scala to transform data and output data into S3
S3	Athena	Query data and output results into S3
EMR	PySpark/Hive in EMR	Transform petabytes of distributed data and output data into S3
RDS, EMR, DynamoDB, Redshift	Data Pipeline	Setup EC2 instances to transform data and output data into S3