Forest Fire Predictive Analytics with AWS Cloud

Team 9: Anna Chow, Hemang Behl, Jason Gonsalves, Kevin Chuang, Richita Das

1. What is the project idea /What the application does?

Wildfires can cause devastating destruction and cost millions of dollars in damage, especially with increasing human development near wilderness or rural areas. The ability to predict wildfires would be a preemptive measure to prevent and/or manage wildfires. Factors affecting wildfires would be explored and used to gauge the probability of a wildfire occurring. Based on the location and the corresponding land type, a machine learning model will be trained to identify the major contributing factors of the fire. The whole process will be carried out in cloud where the model can directly consume the source file.

2. What are the technologies used?

- Python + libraries (numpy, sklearn, scipy, matplotlib, Flask, boto3, sagemaker) + IDE (Jupyter Notebook/Spyder)
- SQL
- AWS S3
- AWS Glacier
- AWS Glue
- AWS Athena
- AWS Redshift
- AWS QuickSight
- AWS SageMaker
- AWS CloudWatch Logs

3. Feature list of the dataset:

- FOD_ID = Global unique identifier.
- FPA_ID = Unique identifier that contains information necessary to track back to the original record in the source dataset.
- SOURCE_SYSTEM_TYPE = Type of source database or system that the record was drawn from (federal, nonfederal, or interagency).
- SOURCE_SYSTEM = Name of or other identifier for source database or system that the record was drawn from.
- NWCG_REPORTING_AGENCY = Active National Wildlife Coordinating Group (NWCG) Unit Identifier for the agency preparing the fire report
- NWCG_REPORTING_UNIT_ID = Active NWCG Unit Identifier for the unit preparing the fire report.

- NWCG_REPORTING_UNIT_NAME = Active NWCG Unit Name for the unit preparing the fire report.
- SOURCE_REPORTING_UNIT = Code for the agency unit preparing the fire report, based on code/name in the source dataset.
- SOURCE_REPORTING_UNIT_NAME = Name of reporting agency unit preparing the fire report, based on code/name in the source dataset.
- LOCAL_FIRE_REPORT_ID = Number or code that uniquely identifies an incident report for a particular reporting unit and a particular calendar year.
- LOCAL_INCIDENT_ID = Number or code that uniquely identifies an incident for a particular local fire management organization within a particular calendar year.
- FIRE_CODE = Code used within the interagency wildland fire community to track and compile cost information for emergency fire suppression
- FIRE_NAME = Name of the incident, from the fire report (primary) or ICS-209 report (secondary).
- ICS_209_INCIDENT_NUMBER = Incident (event) identifier, from the ICS-209 report.
- ICS_209_NAME = Name of the incident, from the ICS-209 report.
- MTBS_ID = Incident identifier, from the MTBS perimeter dataset.
- MTBS_FIRE_NAME = Name of the incident, from the MTBS perimeter dataset.
- COMPLEX_NAME = Name of the complex under which the fire was ultimately managed, when discernible.
- FIRE_YEAR = Calendar year in which the fire was discovered or confirmed to exist.
- DISCOVERY_DATE = Date on which the fire was discovered or confirmed to exist.
- DISCOVERY_DOY = Day of year on which the fire was discovered or confirmed to exist.
- DISCOVERY_TIME = Time of day that the fire was discovered or confirmed to exist.
- STAT_CAUSE_CODE = Code for the (statistical) cause of the fire.
- STAT_CAUSE_DESCR = Description of the (statistical) cause of the fire.
- CONT_DATE = Date on which the fire was declared contained or otherwise controlled (mm/dd/yyyy)
- CONT_DOY = Day of year on which the fire was declared contained or otherwise controlled.
- CONT_TIME = Time of day that the fire was declared contained or otherwise controlled (hhmm where hh=hour, mm=minutes).
- FIRE_SIZE = Estimate of acres within the final perimeter of the fire.
- FIRE_SIZE_CLASS = Code for fire size based on the number of acres within the final fire perimeter expenditures (A=greater than 0 but less than or equal to 0.25 acres, B=0.26-9.9 acres, C=10.0-99.9 acres, D=100-299 acres, E=300 to 999 acres, F=1000 to 4999 acres, and G=5000+ acres).
- LATITUDE = Latitude (NAD83) for point location of the fire (decimal degrees).
- LONGITUDE = Longitude (NAD83) for point location of the fire (decimal degrees).

- OWNER_CODE = Code for primary owner or entity responsible for managing the land at the point of origin of the fire at the time of the incident.
- OWNER_DESCR = Name of primary owner or entity responsible for managing the land at the point of origin of the fire at the time of the incident.
- STATE = Two-letter alphabetic code for the state in which the fire burned (or originated), based on the nominal designation in the fire report.
- COUNTY = County, or equivalent, in which the fire burned (or originated), based on nominal designation in the fire report.
- FIPS_CODE = Three-digit code from the Federal Information Process Standards (FIPS) publication 6-4 for representation of counties and equivalent entities.
- FIPS_NAME = County name from the FIPS publication 6-4 for representation of counties and equivalent entities.

Features post feature engineering:

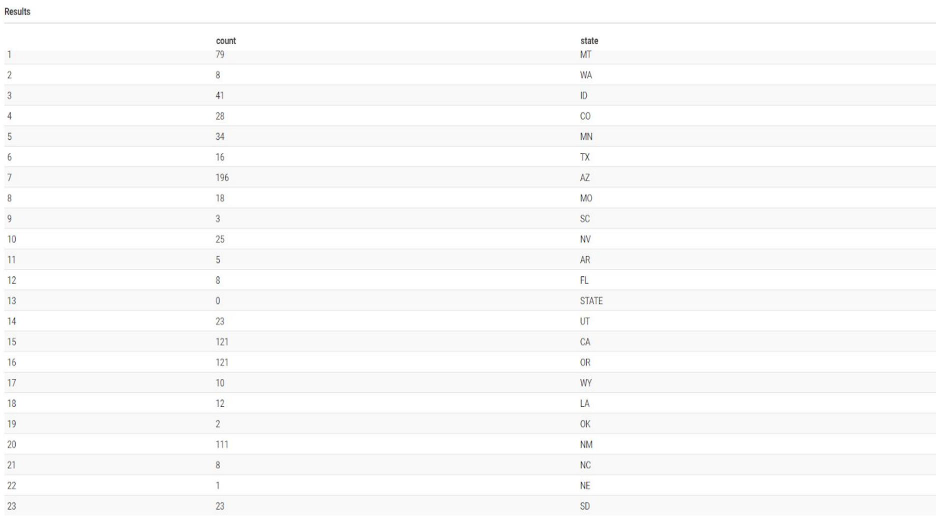
- FIRE_YEAR (long)
- STAT_CAUSE_DESCR (string)
- FIRE_SIZE (double)
- FIRE_SIZE_CLASS (string)
- LATITUDE (double)
- LONGITUDE (double)
- STATE (string)
- COUNTY (long)
- DISCOVERY_DATE (double)
- CNT_DATE (double)

4. Sample Demo Screenshots

AWS Athena

SQL Queries

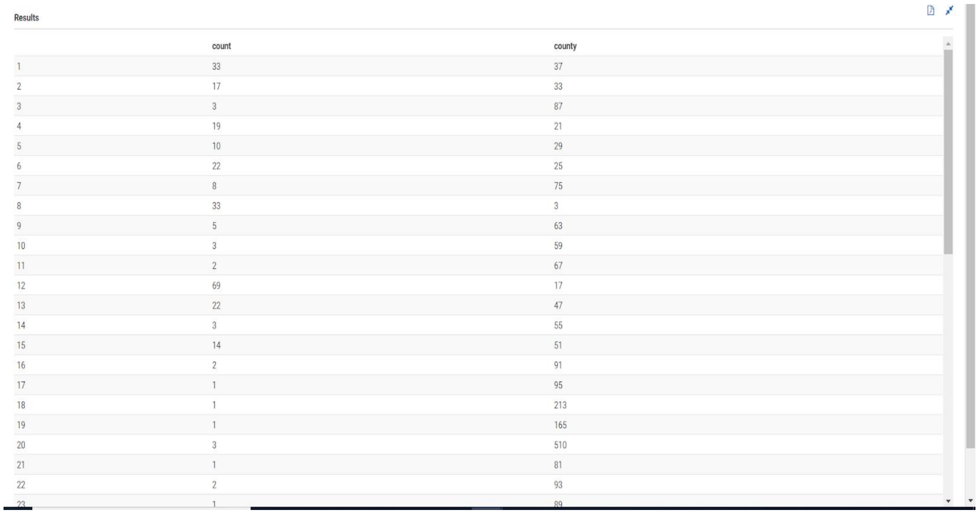
SELECT count(fod_id) as count, state FROM sampledb.test_266 GROUP BY state;



Results

	count	state
1	79	MT
2	8	WA
3	41	ID
4	28	CO
5	34	MN
6	16	TX
7	196	AZ
8	18	MO
9	3	SC
10	25	NV
11	5	AR
12	8	FL
13	0	STATE
14	23	UT
15	121	CA
16	121	OR
17	10	WY
18	12	LA
19	2	OK
20	111	NM
21	8	NC
22	1	NE
23	23	SD

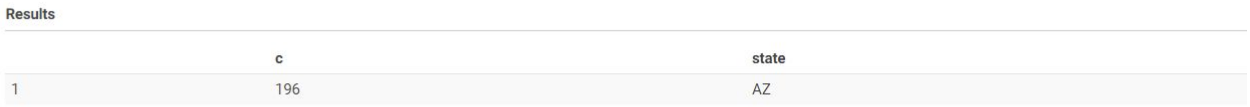
SELECT count(fod_id) as count, county FROM sampledb.test_266 GROUP BY county;



Results

	count	county
1	33	37
2	17	33
3	3	87
4	19	21
5	10	29
6	22	25
7	8	75
8	33	3
9	5	63
10	3	59
11	2	67
12	69	17
13	22	47
14	3	55
15	14	51
16	2	91
17	1	95
18	1	213
19	1	165
20	3	510
21	1	81
22	2	93
23	1	89

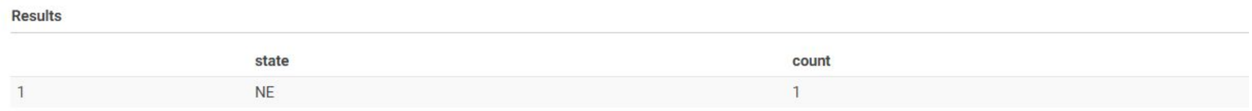
select max(count) as c, state from
(SELECT state,count(fod_id) as count FROM sampledb.test_266 GROUP BY state order by count desc) group by state order by c desc limit 1 ;



Results

	c	state
1	196	AZ

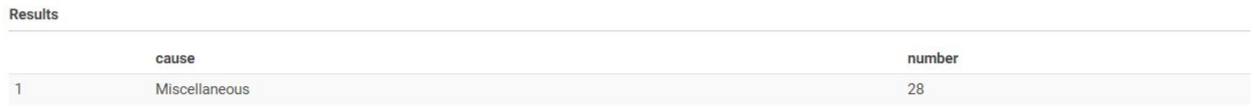
```
select * from
(SELECT state,count(fod_id) as count FROM sampledb.test_266 GROUP  BY state order by count
limit 2) where state <> 'STATE';
```



Results

	state	count
1	NE	1

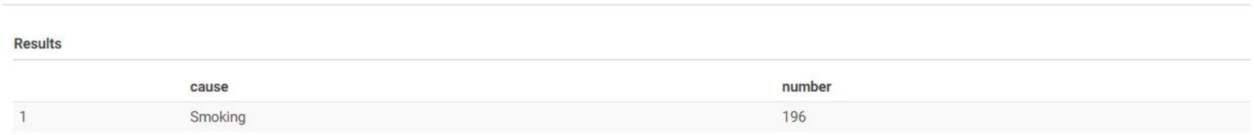
```
select max(stat_cause_descr) as cause,count(stat_cause_descr) as number from
sampledb.test_266 where state='CO' ;
```



Results

	cause	number
1	Miscellaneous	28

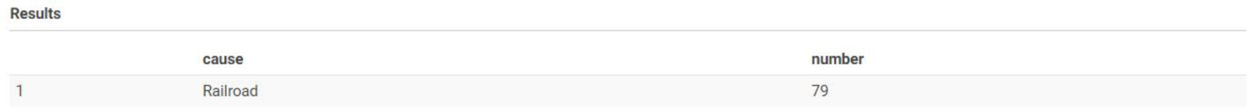
```
select max(stat_cause_descr) as cause,count(stat_cause_descr) as number from
sampledb.test_266 where state='AZ' ;
```



Results

	cause	number
1	Smoking	196

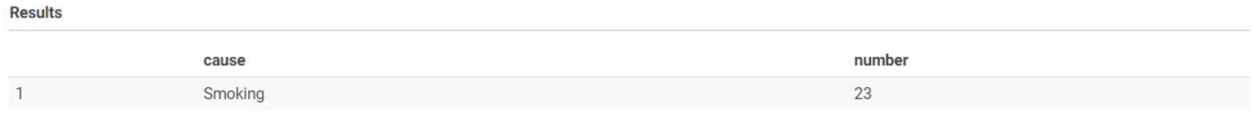
```
select max(stat_cause_descr) as cause,count(stat_cause_descr) as number from
sampledb.test_266 where state='MT' ;
```



Results

	cause	number
1	Railroad	79

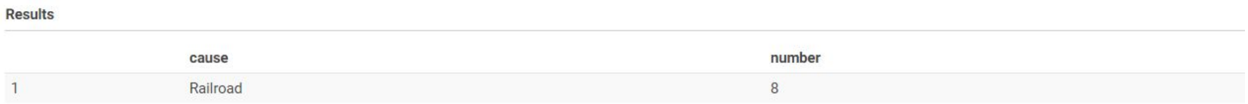
```
select max(stat_cause_descr) as cause,count(stat_cause_descr) as number from
sampledb.test_266 where state='UT' ;
```



Results

	cause	number
1	Smoking	23

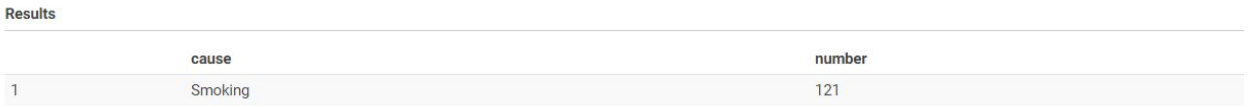
```
select max(stat_cause_descr) as cause,count(stat_cause_descr) as number from
sampledb.test_266 where state='WA' ;
```



Results

	cause	number
1	Railroad	8

select max(stat_cause_descr) as cause,count(stat_cause_descr) as number from sampledb.test_266 where state='OR' ;

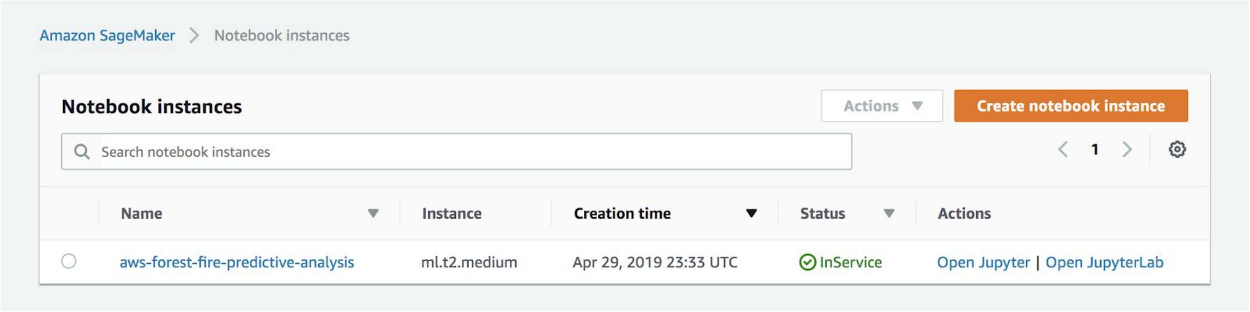


Results

	cause	number
1	Smoking	121

AWS SageMaker

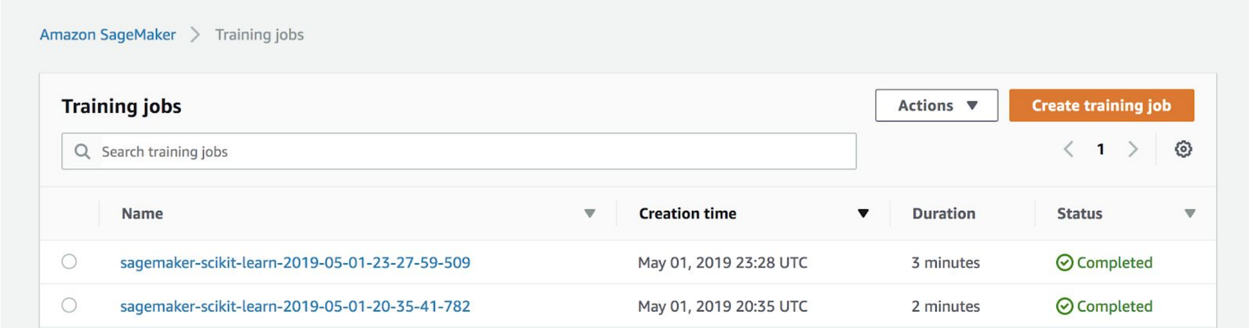
Notebook Instances



Amazon SageMaker > Notebook instances
Notebook instances
Actions ▼
Create notebook instance
Search notebook instances
< 1 >

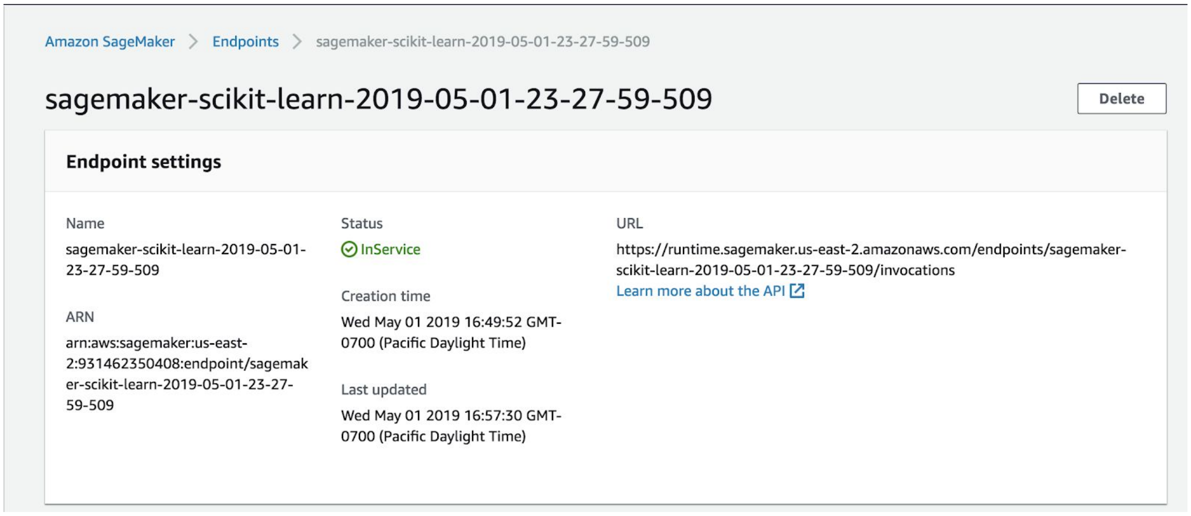
	Name	Instance	Creation time	Status	Actions
○	aws-forest-fire-predictive-analysis	ml.t2.medium	Apr 29, 2019 23:33 UTC	✓ InService	Open Jupyter | Open JupyterLab

Training Jobs



	Name	Creation time	Duration	Status
○	sagemaker-scikit-learn-2019-05-01-23-27-59-509	May 01, 2019 23:28 UTC	3 minutes	✓ Completed
○	sagemaker-scikit-learn-2019-05-01-20-35-41-782	May 01, 2019 20:35 UTC	2 minutes	✓ Completed

SageMaker Endpoint (Deployed Model)



Amazon SageMaker > Endpoints > sagemaker-scikit-learn-2019-05-01-23-27-59-509

sagemaker-scikit-learn-2019-05-01-23-27-59-509

Endpoint settings

Name
sagemaker-scikit-learn-2019-05-01-23-27-59-509

ARN
arn:aws:sagemaker:us-east-2:931462350408:endpoint/sagemaker-scikit-learn-2019-05-01-23-27-59-509

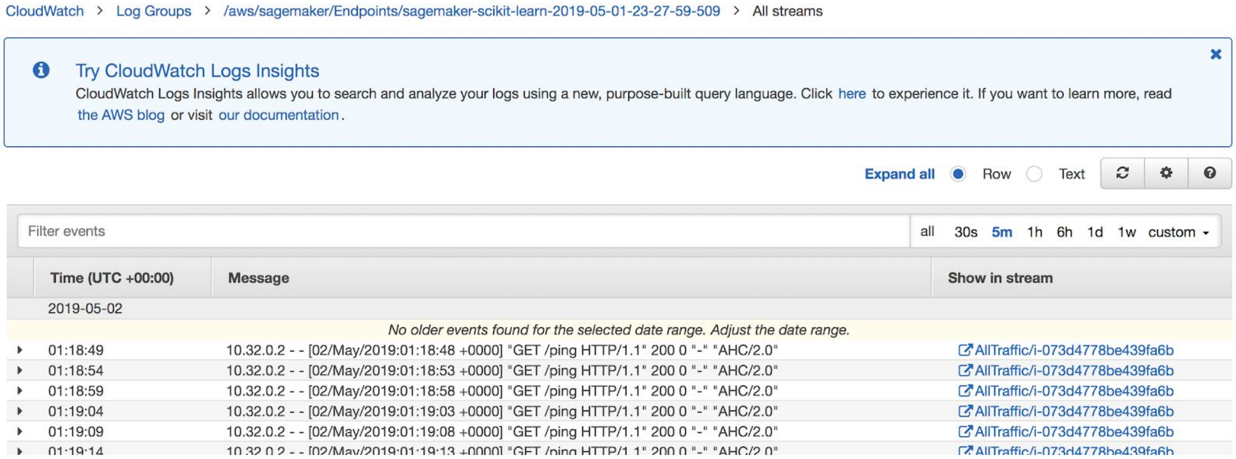
Status
InService

Creation time
Wed May 01 2019 16:49:52 GMT-0700 (Pacific Daylight Time)

Last updated
Wed May 01 2019 16:57:30 GMT-0700 (Pacific Daylight Time)

URL
https://runtime.sagemaker.us-east-2.amazonaws.com/endpoints/sagemaker-scikit-learn-2019-05-01-23-27-59-509/invocations
Learn more about the API

CloudWatch Logs



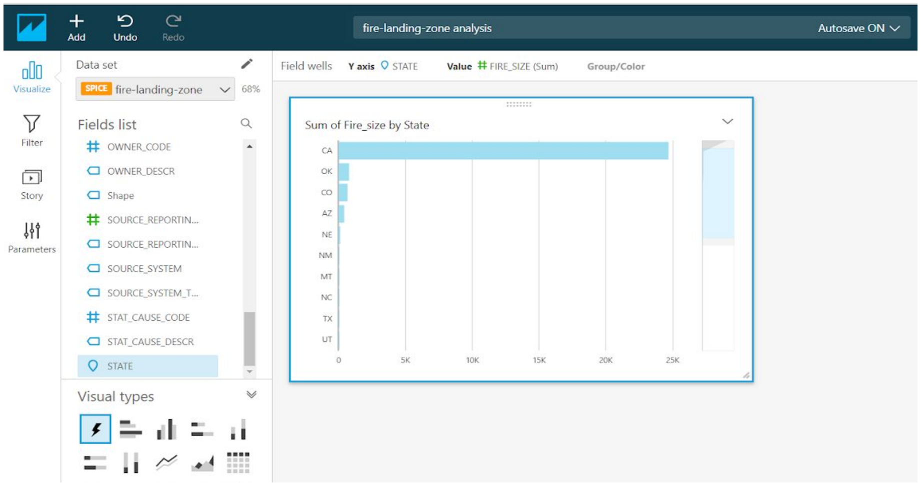
CloudWatch > Log Groups > /aws/sagemaker/Endpoints/sagemaker-scikit-learn-2019-05-01-23-27-59-509 > All streams

Try CloudWatch Logs Insights
CloudWatch Logs Insights allows you to search and analyze your logs using a new, purpose-built query language. Click here to experience it. If you want to learn more, read the AWS blog or visit our documentation.

```
Time (UTC +00:00)  Message                                                                      Show in stream
2019-05-02
                   No older events found for the selected date range. Adjust the date range.
01:18:49           10.32.0.2 - - [02/May/2019:01:18:48 +0000] "GET /ping HTTP/1.1" 200 0 "-" "AHC/2.0"   AllTraffic/i-073d4778be439fa6b
01:18:54           10.32.0.2 - - [02/May/2019:01:18:53 +0000] "GET /ping HTTP/1.1" 200 0 "-" "AHC/2.0"   AllTraffic/i-073d4778be439fa6b
01:18:59           10.32.0.2 - - [02/May/2019:01:18:58 +0000] "GET /ping HTTP/1.1" 200 0 "-" "AHC/2.0"   AllTraffic/i-073d4778be439fa6b
01:19:04           10.32.0.2 - - [02/May/2019:01:19:03 +0000] "GET /ping HTTP/1.1" 200 0 "-" "AHC/2.0"   AllTraffic/i-073d4778be439fa6b
01:19:09           10.32.0.2 - - [02/May/2019:01:19:08 +0000] "GET /ping HTTP/1.1" 200 0 "-" "AHC/2.0"   AllTraffic/i-073d4778be439fa6b
01:19:14           10.32.0.2 - - [02/May/2019:01:19:13 +0000] "GET /ping HTTP/1.1" 200 0 "-" "AHC/2.0"   AllTraffic/i-073d4778be439fa6b
```

QuickSight Analysis

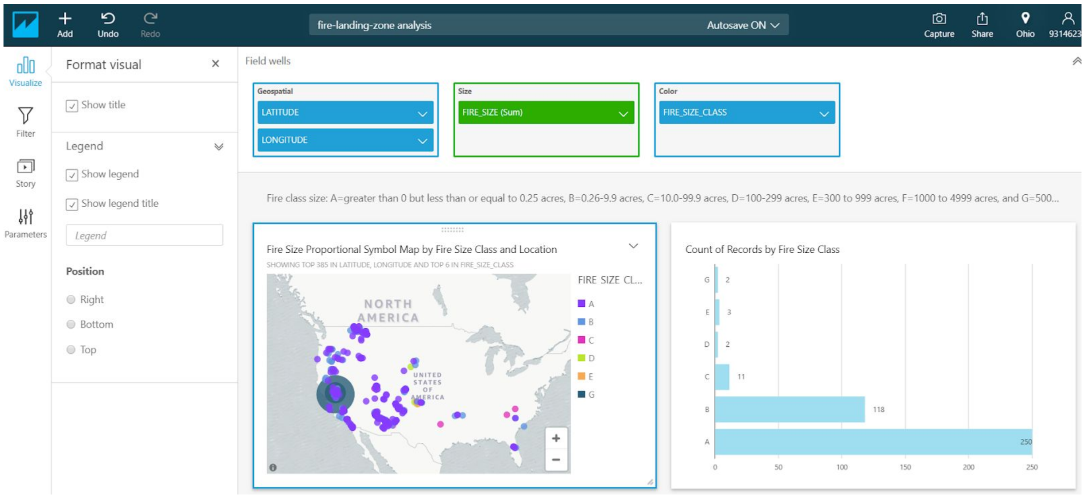
Dashboard Setup



fire-landing-zone analysis

Data set: SPICE fire-landing-zone 68%

Field wells — Y axis: STATE — Value: FIRE_SIZE (Sum) — Group/Color



fire-landing-zone analysis

Format visual

☑ Show title

Legend

☑ Show legend

☑ Show legend title

Legend

Position

○ Right

○ Bottom

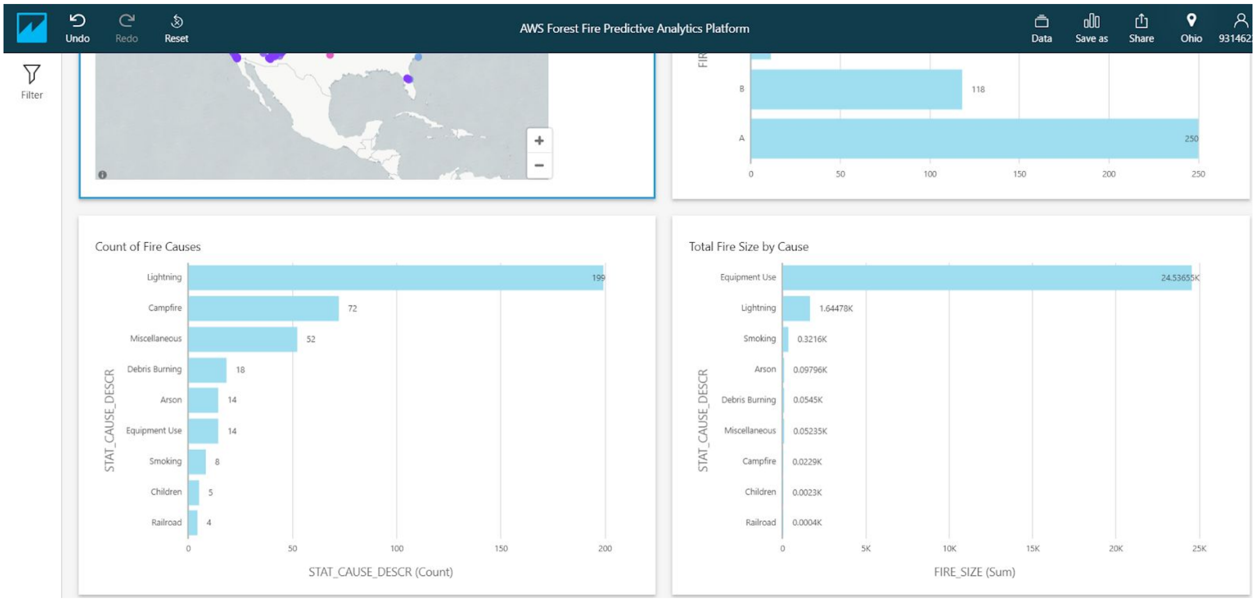
○ Top

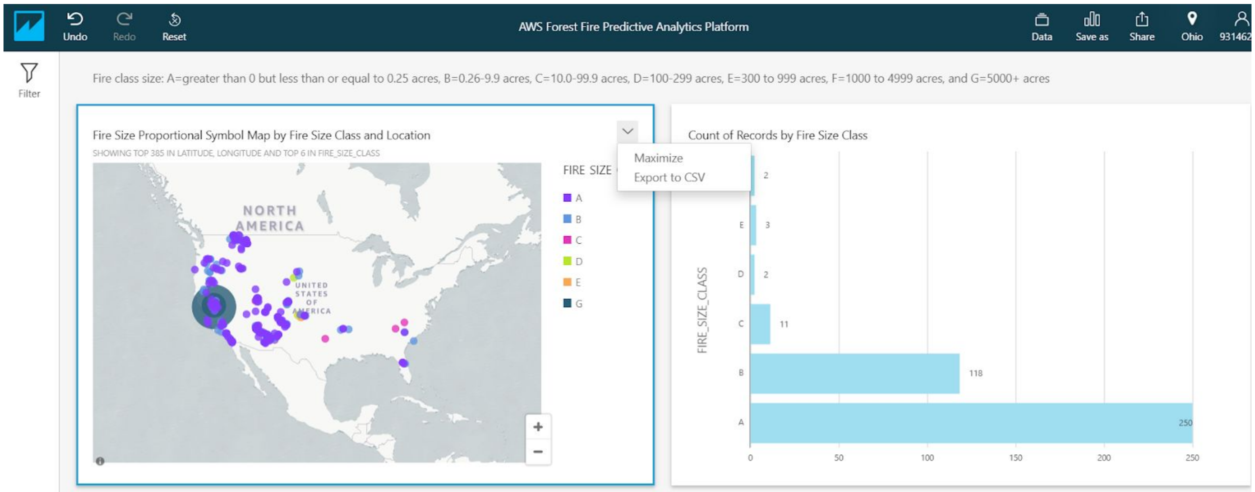
Field wells — Geospatial: LATITUDE, LONGITUDE — Size: FIRE_SIZE (Sum) — Color: FIRE_SIZE_CLASS

Fire class size: A=greater than 0 but less than or equal to 0.25 acres, B=0.26-9.9 acres, C=10.0-99.9 acres, D=100-299 acres, E=300 to 999 acres, F=1000 to 4999 acres, and G=500...

Add
Undo
Redo
Autosave ON
Capture
Share
Ohio
9314623
Visualize
Filter
Story
Parameters
Fire Size Proportional Symbol Map by Fire Size Class and Location
SHOWING TOP 385 IN LATITUDE, LONGITUDE AND TOP 6 IN FIRE_SIZE_CLASS
NORTH AMERICA
UNITED STATES OF AMERICA
FIRE SIZE CL...
A
B
C
D
E
G
Count of Records by Fire Size Class
G
2
E
3
D
2
C
11
B
118
A
250
0
50
100
150
200
250

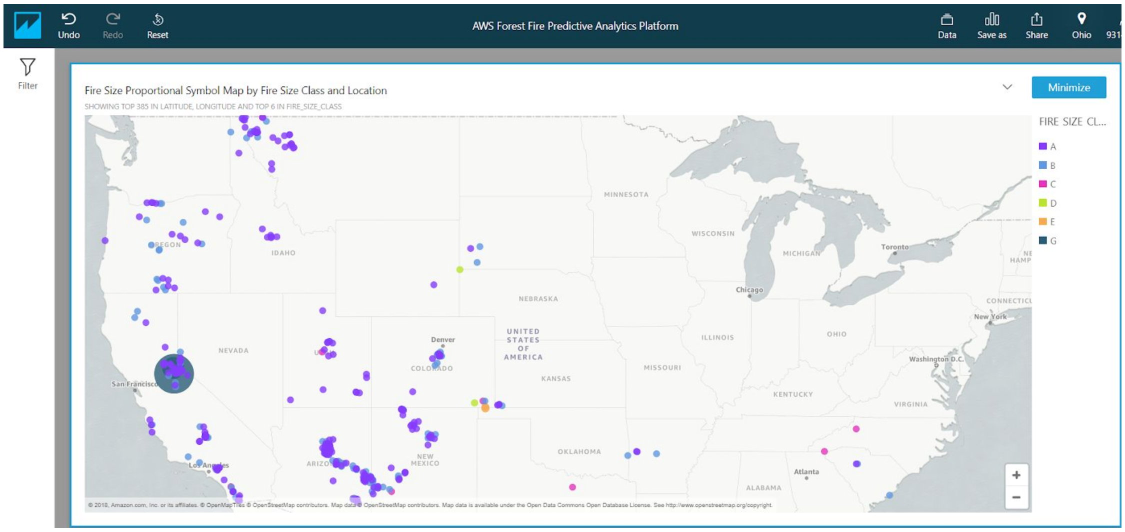
QuickSight Dashboard Overview




Undo
Redo
Reset
AWS Forest Fire Predictive Analytics Platform
Data
Save as
Share
Ohio
931462
Filter

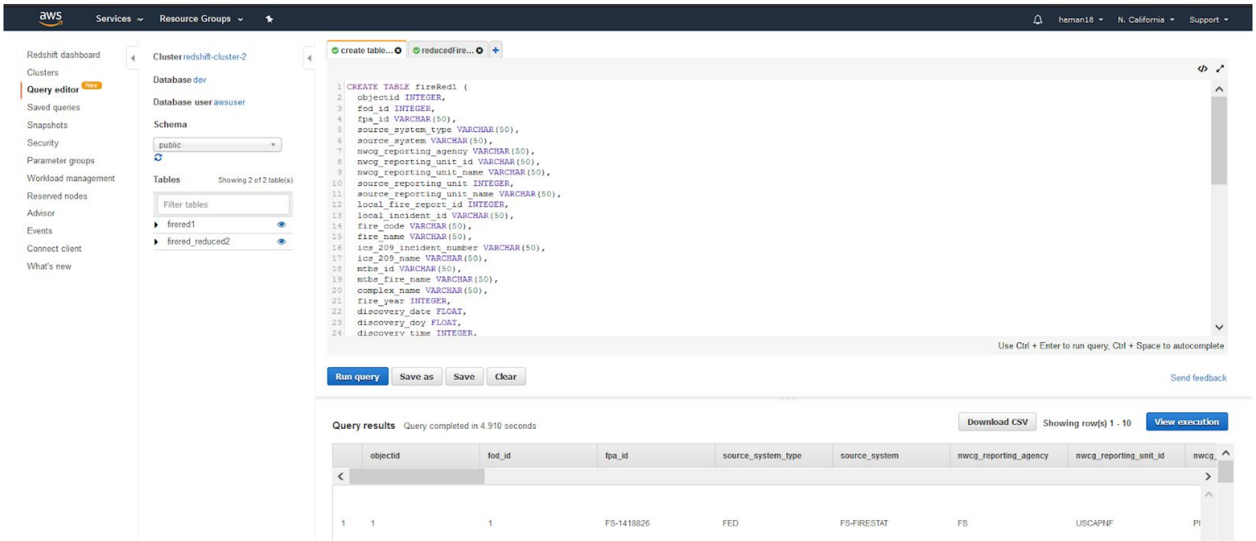
Fire class size: A=greater than 0 but less than or equal to 0.25 acres, B=0.26-9.9 acres, C=10.0-99.9 acres, D=100-299 acres, E=300 to 999 acres, F=1000 to 4999 acres, and G=5000+ acres

Fire Size Proportional Symbol Map by Fire Size Class and Location
SHOWING TOP 385 IN LATITUDE, LONGITUDE AND TOP 6 IN FIRE_SIZE_CLASS
NORTH AMERICA
UNITED STATES OF AMERICA
FIRE SIZE
A
B
C
D
E
G
Maximize
Export to CSV
Count of Records by Fire Size Class
FIRE_SIZE_CLASS
2
E 3
D 2
C 11
B 118
A 250
0
50
100
150
200
250


Undo
Redo
Reset
AWS Forest Fire Predictive Analytics Platform
Data
Save as
Share
Ohio
931
Filter
Fire Size Proportional Symbol Map by Fire Size Class and Location
SHOWING TOP 385 IN LATITUDE, LONGITUDE AND TOP 6 IN FIRE_SIZE_CLASS
Minimize
FIRE SIZE CL...
A
B
C
D
E
G
MINNESOTA
WISCONSIN
MICHIGAN
Toronto
OREGON
IDAHO
NEBRASKA
Chicago
NEVADA
Denver
COLORADO
UNITED STATES OF AMERICA
ILLINOIS
OHIO
New York
CONNECTICUT
Washington D.C.
San Francisco
KANSAS
MISSOURI
KENTUCKY
VIRGINIA
OKLAHOMA
ARIZONA
NEW MEXICO
Los Angeles
Atlanta
ALABAMA
© 2018, Amazon.com, Inc. or its affiliates. © OpenMapTiles © OpenStreetMap contributors. Map data © OpenStreetMap contributors. Map data is available under the Open Data Commons Open Database License. See http://www.openstreetmap.org/copyright.

RedShift Demo

Querying data loaded from S3 (raw data)

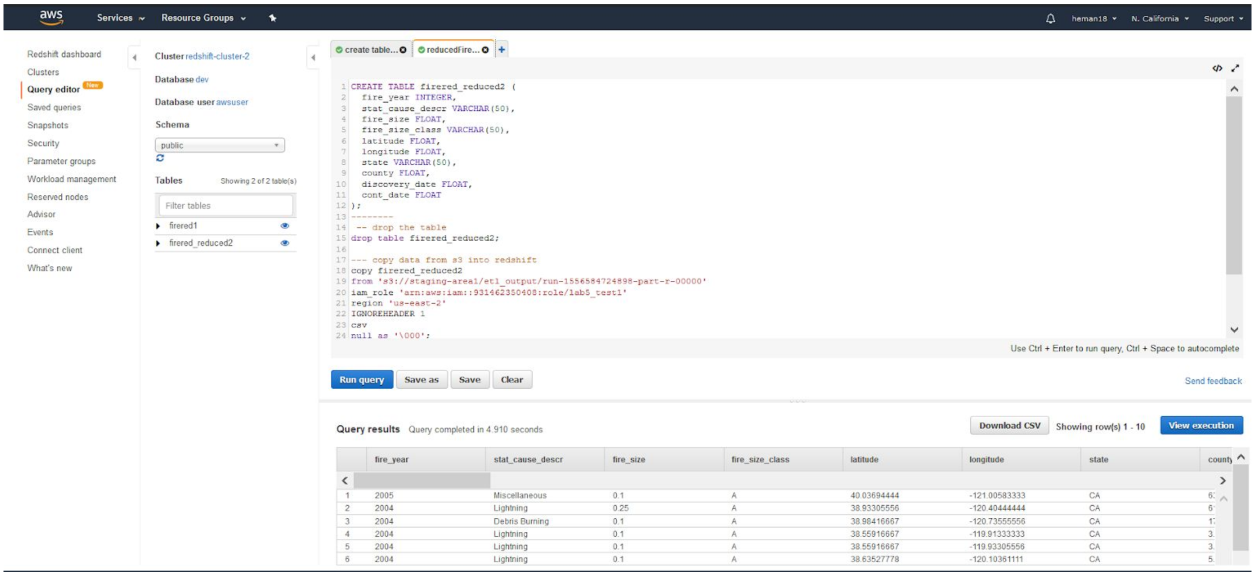


```
CREATE TABLE fireRed1 (
  objectid INTEGER,
  fod_id INTEGER,
  fpa_id VARCHAR(50),
  source_system_type VARCHAR(50),
  source_system VARCHAR(50),
  nwcg_reporting_agency VARCHAR(50),
  nwcg_reporting_unit_id VARCHAR(50),
  nwcg_reporting_unit_name VARCHAR(50),
  source_reporting_unit INTEGER,
  source_reporting_unit_name VARCHAR(50),
  local_fire_report_id INTEGER,
  local_incident_id VARCHAR(50),
  fire_code VARCHAR(50),
  fire_name VARCHAR(50),
  ics_209_incident_number VARCHAR(50),
  ics_209_name VARCHAR(50),
  mtbs_id VARCHAR(50),
  mtbs_fire_name VARCHAR(50),
  complex_name VARCHAR(50),
  fire_year INTEGER,
  discovery_date FLOAT,
  discovery_doy FLOAT,
  discovery_time INTEGER,
```

Query results Query completed in 4.910 seconds

	objectid	fod_id	fpa_id	source_system_type	source_system	nwcg_reporting_agency	nwcg_reporting_unit_id	nwcg_
1	1	1	FS-1418826	FED	FS-FIRESTAT	FS	USCAPNF	PI

Querying data loaded from S3 (reduced data set)



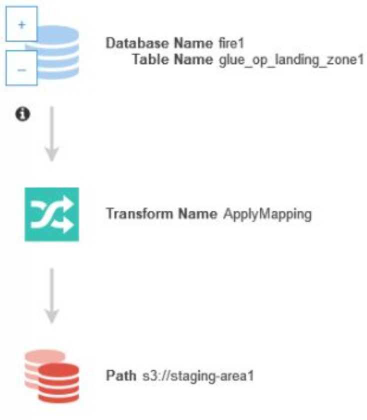
```
CREATE TABLE firered_reduced2 (
  fire_year INTEGER,
  stat_cause_descr VARCHAR(50),
  fire_size FLOAT,
  fire_size_class VARCHAR(50),
  latitude FLOAT,
  longitude FLOAT,
  state VARCHAR(50),
  county FLOAT,
  discovery_date FLOAT,
  cont_date FLOAT
);
--------
 -- drop the table
drop table firered_reduced2;

--- copy data from s3 into redshift
copy firered_reduced2
from 's3://staging-area1/etl_output/run-1556584724898-part-r-00000'
iam_role 'arn:aws:iam::931462350408:role/lab5_test1'
region 'us-east-2'
IGNOREHEADER 1
csv
null as '\000';
```

Query results Query completed in 4.910 seconds

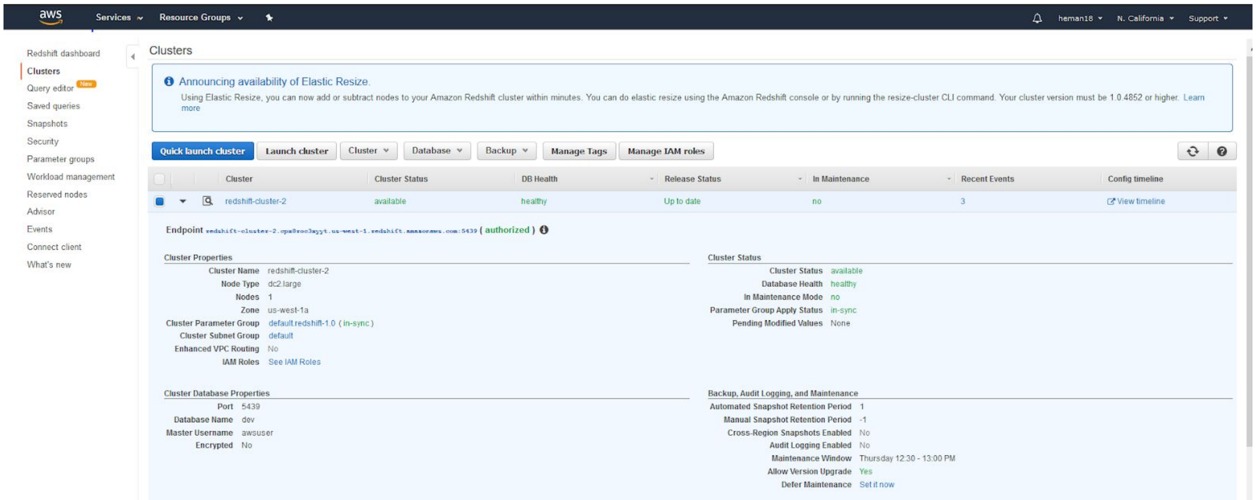
	fire_year	stat_cause_descr	fire_size	fire_size_class	latitude	longitude	state	county
1	2005	Miscellaneous	0.1	A	40.03694444	-121.00583333	CA	6:
2	2004	Lightning	0.25	A	38.93305556	-120.40444444	CA	6
3	2004	Debris Burning	0.1	A	38.98416667	-120.73555556	CA	1:
4	2004	Lightning	0.1	A	38.55916667	-119.91333333	CA	3.
5	2004	Lightning	0.1	A	38.55916667	-119.93305556	CA	3.
6	2004	Lightning	0.1	A	38.63527778	-120.10361111	CA	5.

AWS Glue



5. AWS Configurations Screenshots

5.1 Redshift Cluster



Clusters

Announcing availability of Elastic Resize.

Using Elastic Resize, you can now add or subtract nodes to your Amazon Redshift cluster within minutes. You can do elastic resize using the Amazon Redshift console or by running the resize-cluster CLI command. Your cluster version must be 1.0.4852 or higher. Learn more

Cluster	Cluster Status	DB Health	Release Status	In Maintenance	Recent Events	Config timeline
redshift-cluster-2	available	healthy	Up to date	no	3	View timeline

Endpoint redshift-cluster-2.cpx8roc3xyyt.us-west-1.redshift.amazonaws.com:5439 (authorized)

Cluster Properties

Cluster Name	redshift-cluster-2
Node Type	dc2.large
Nodes	1
Zone	us-west-1a
Cluster Parameter Group	default.redshift-1.0 (in-sync)
Cluster Subnet Group	default
Enhanced VPC Routing	No
IAM Roles	See IAM Roles

Cluster Status

Cluster Status	available
Database Health	healthy
In Maintenance Mode	no
Parameter Group Apply Status	in-sync
Pending Modified Values	None

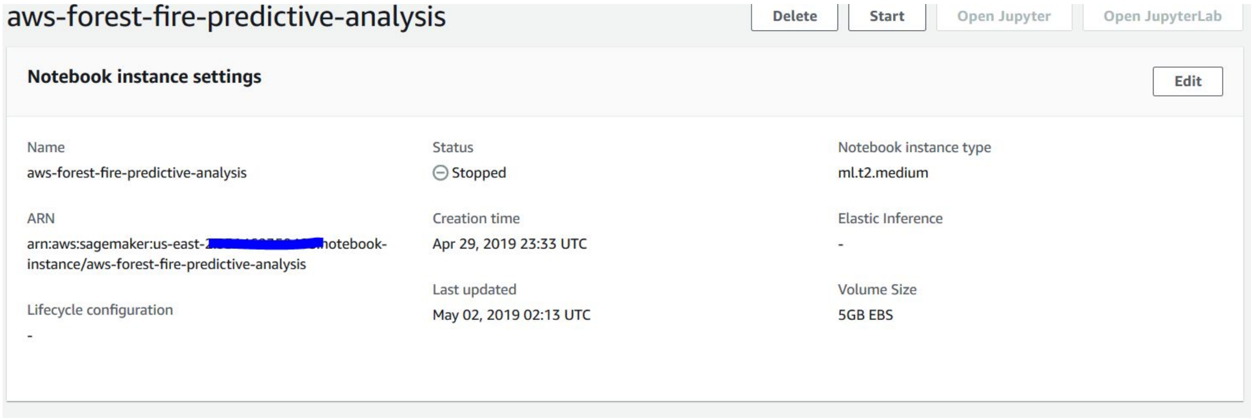
Cluster Database Properties

Port	5439
Database Name	dev
Master Username	awsuser
Encrypted	No

Backup, Audit Logging, and Maintenance

Automated Snapshot Retention Period	1
Manual Snapshot Retention Period	-1
Cross-Region Snapshots Enabled	No
Audit Logging Enabled	No
Maintenance Window	Thursday 12:30 - 13:00 PM
Allow Version Upgrade	Yes
Defer Maintenance	Set it now

5.2 Sagemaker

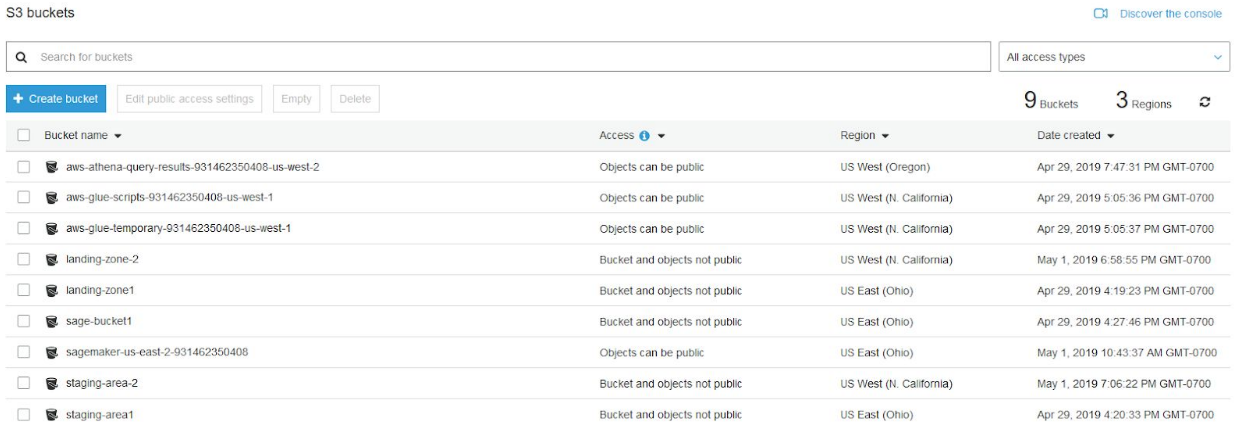


aws-forest-fire-predictive-analysis

Notebook instance settings

Name aws-forest-fire-predictive-analysis	Status ⊖ Stopped	Notebook instance type ml.t2.medium
ARN arn:aws:sagemaker:us-east-[redacted]otebook-instance/aws-forest-fire-predictive-analysis	Creation time Apr 29, 2019 23:33 UTC	Elastic Inference -
Lifecycle configuration -	Last updated May 02, 2019 02:13 UTC	Volume Size 5GB EBS

5.3 S3

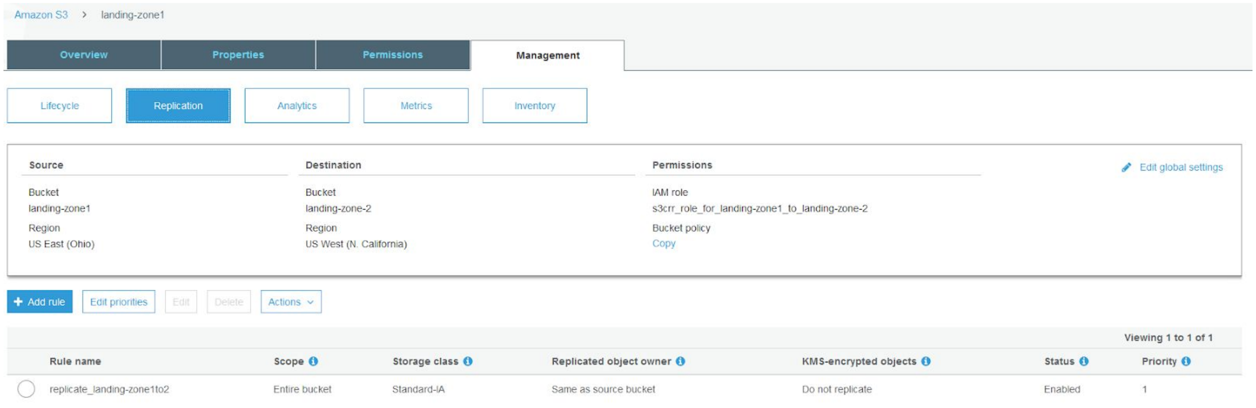


S3 buckets

9 Buckets 3 Regions

□	Bucket name	Access	Region	Date created
□	aws-athena-query-results-931462350408-us-west-2	Objects can be public	US West (Oregon)	Apr 29, 2019 7:47:31 PM GMT-0700
□	aws-glue-scripts-931462350408-us-west-1	Objects can be public	US West (N. California)	Apr 29, 2019 5:05:36 PM GMT-0700
□	aws-glue-temporary-931462350408-us-west-1	Objects can be public	US West (N. California)	Apr 29, 2019 5:05:37 PM GMT-0700
□	landing-zone-2	Bucket and objects not public	US West (N. California)	May 1, 2019 6:58:55 PM GMT-0700
□	landing-zone1	Bucket and objects not public	US East (Ohio)	Apr 29, 2019 4:19:23 PM GMT-0700
□	sage-bucket1	Bucket and objects not public	US East (Ohio)	Apr 29, 2019 4:27:46 PM GMT-0700
□	sagemaker-us-east-2-931462350408	Objects can be public	US East (Ohio)	May 1, 2019 10:43:37 AM GMT-0700
□	staging-area-2	Bucket and objects not public	US West (N. California)	May 1, 2019 7:06:22 PM GMT-0700
□	staging-area1	Bucket and objects not public	US East (Ohio)	Apr 29, 2019 4:20:33 PM GMT-0700

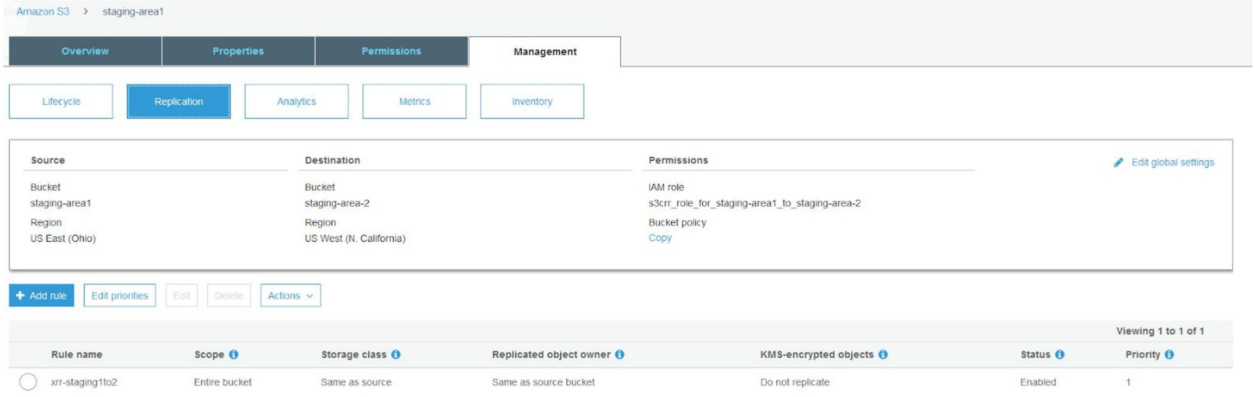
5.3.1 Cross Region Replication



Source	Destination	Permissions
Bucket landing-zone1	Bucket landing-zone-2	IAM role s3crr_role_for_landing-zone1_to_landing-zone-2
Region US East (Ohio)	Region US West (N. California)	Bucket policy Copy

Viewing 1 to 1 of 1

	Rule name	Scope	Storage class	Replicated object owner	KMS-encrypted objects	Status	Priority
○	replicate_landing-zone1to2	Entire bucket	Standard-IA	Same as source bucket	Do not replicate	Enabled	1



Source

Bucket
staging-area1
Region
US East (Ohio)

Destination

Bucket
staging-area-2
Region
US West (N. California)

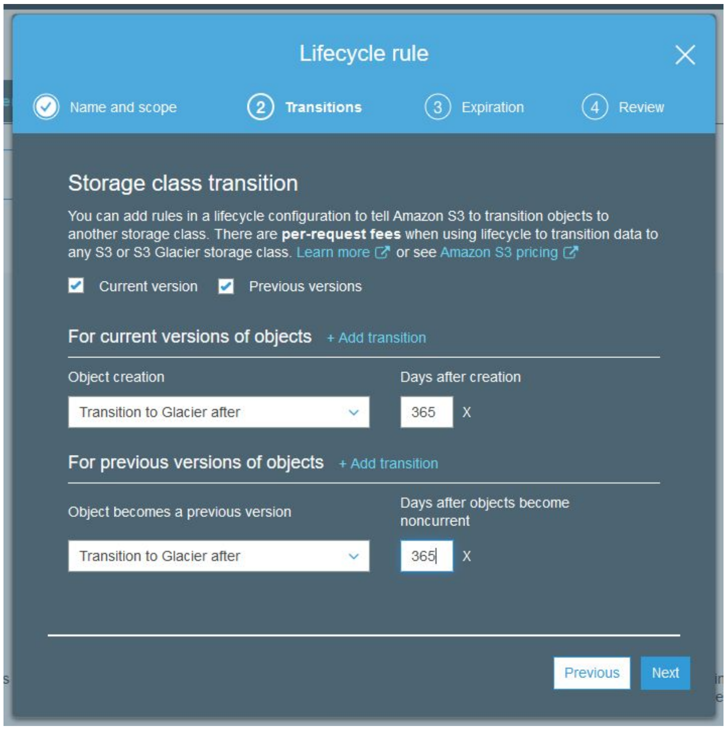
Permissions

IAM role
s3crr_role_for_staging-area1_to_staging-area-2
Bucket policy
Copy

Viewing 1 to 1 of 1

	Rule name	Scope	Storage class	Replicated object owner	KMS-encrypted objects	Status	Priority
○	xrr-staging1to2	Entire bucket	Same as source	Same as source bucket	Do not replicate	Enabled	1

5.3.2 Transition to Glacier



Storage class transition

You can add rules in a lifecycle configuration to tell Amazon S3 to transition objects to another storage class. There are **per-request fees** when using lifecycle to transition data to any S3 or S3 Glacier storage class. Learn more or see Amazon S3 pricing

☑ Current version ☑ Previous versions

For current versions of objects + Add transition

Object creation: Transition to Glacier after

Days after creation: 365 X

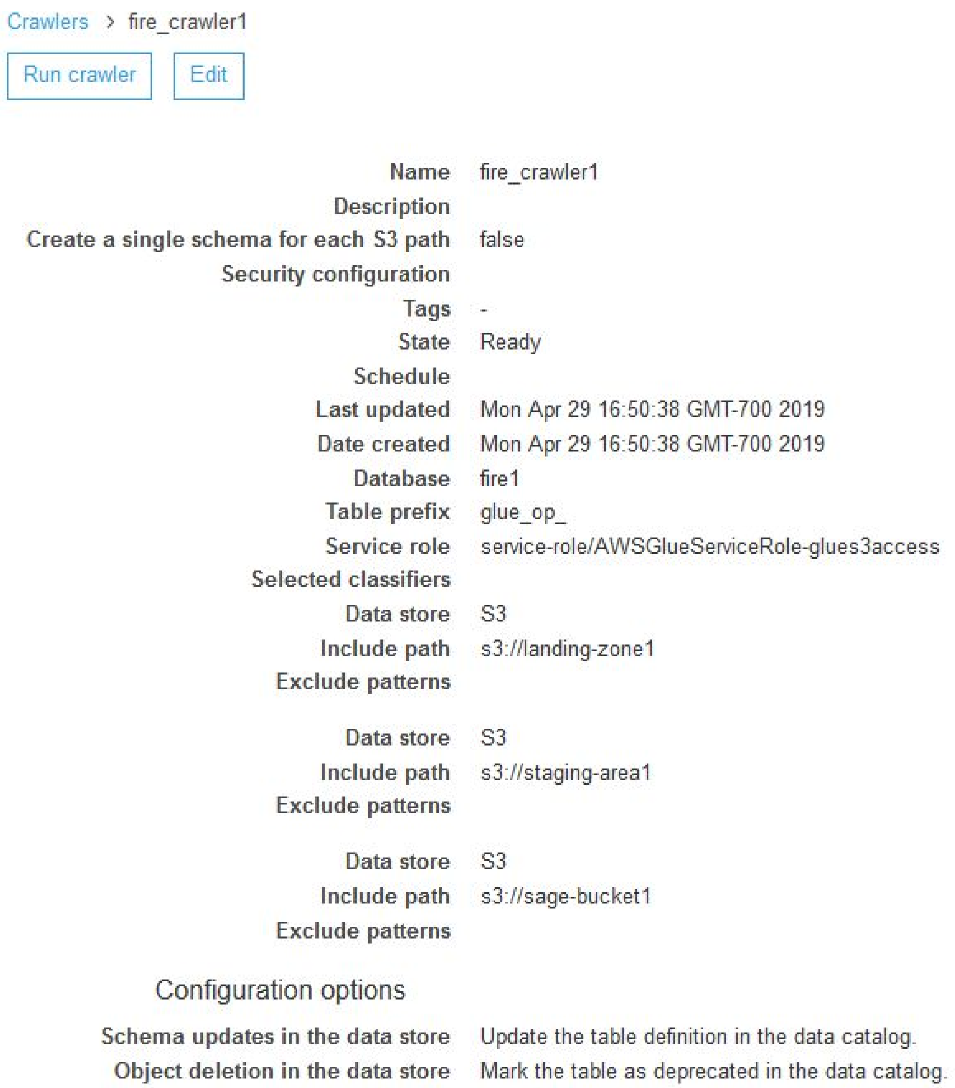
For previous versions of objects + Add transition

Object becomes a previous version: Transition to Glacier after

Days after objects become noncurrent: 365 X

5.4 Glue

5.4.1 Crawler

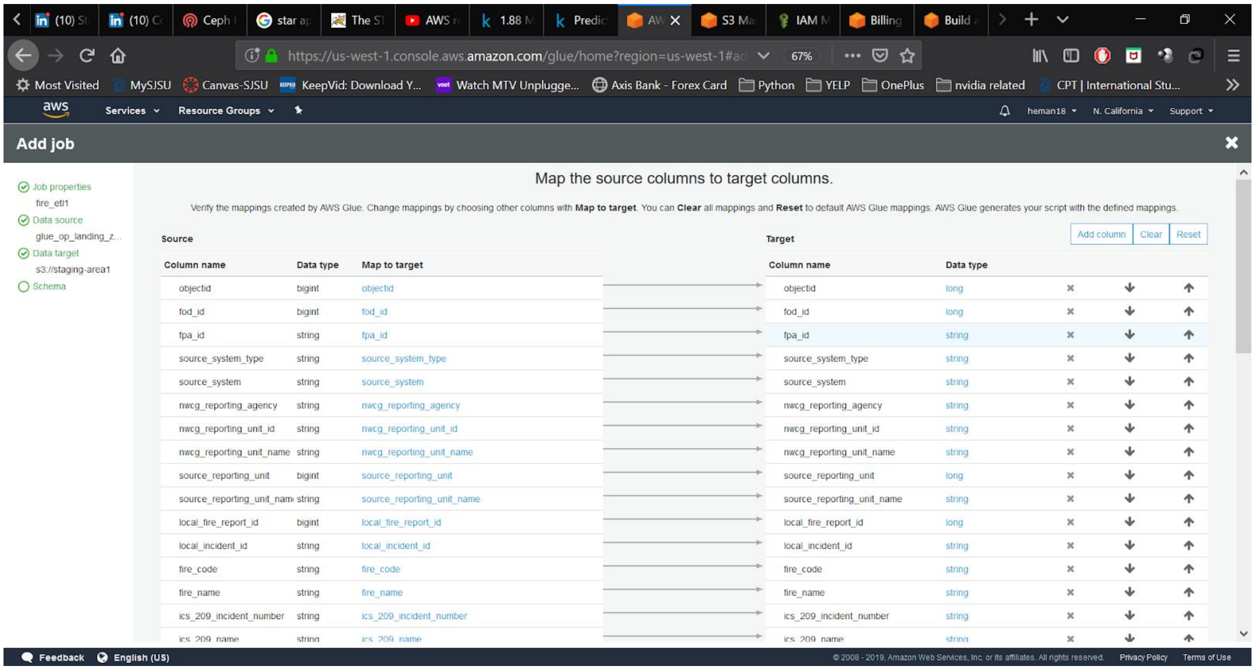


Name	fire_crawler1
Description	
Create a single schema for each S3 path	false
Security configuration	
Tags	-
State	Ready
Schedule	
Last updated	Mon Apr 29 16:50:38 GMT-700 2019
Date created	Mon Apr 29 16:50:38 GMT-700 2019
Database	fire1
Table prefix	glue_op_
Service role	service-role/AWSGlueServiceRole-glues3access
Selected classifiers	
Data store	S3
Include path	s3://landing-zone1
Exclude patterns	
Data store	S3
Include path	s3://staging-area1
Exclude patterns	
Data store	S3
Include path	s3://sage-bucket1
Exclude patterns	

Configuration options

Schema updates in the data store	Update the table definition in the data catalog.
Object deletion in the data store	Mark the table as deprecated in the data catalog.

5.4.2 ETL Mapping



Job properties: fire_etl1

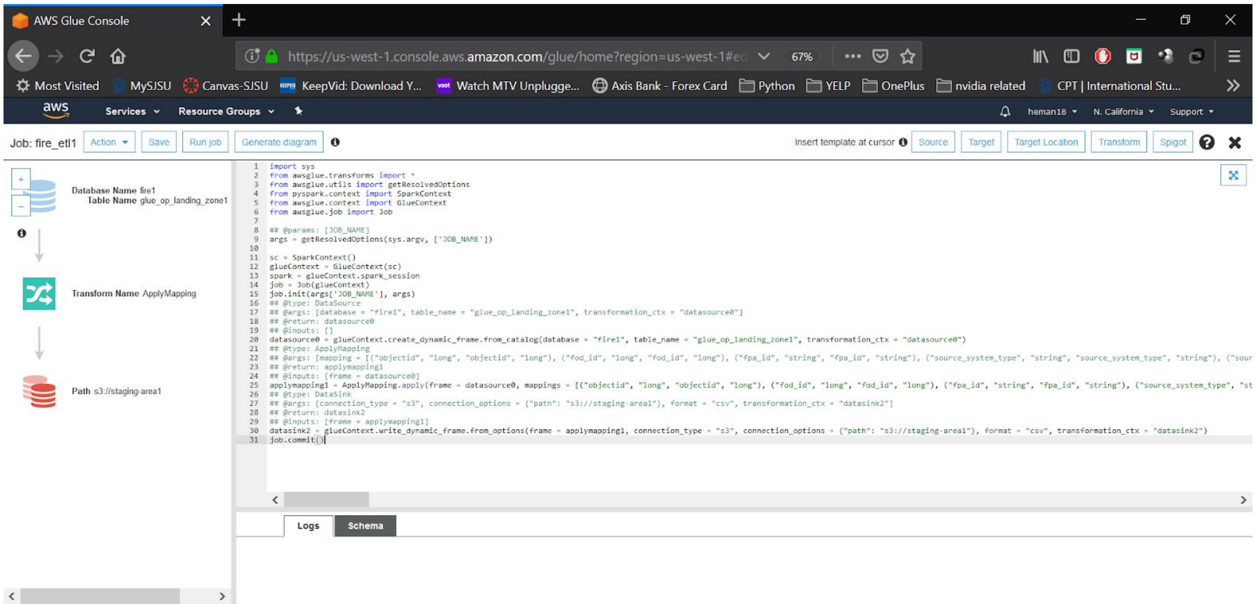
Data source: glue_op_landing_z...

Data target: s3://staging-area1

Map the source columns to target columns.

Verify the mappings created by AWS Glue. Change mappings by choosing other columns with **Map to target**. You can **Clear** all mappings and **Reset** to default AWS Glue mappings. AWS Glue generates your script with the defined mappings.

Source Column name	Data type	Map to target	Target Column name	Data type
objectid	bigint	objectid	objectid	long
fod_id	bigint	fod_id	fod_id	long
fpa_id	string	fpa_id	fpa_id	string
source_system_type	string	source_system_type	source_system_type	string
source_system	string	source_system	source_system	string
nwcg_reporting_agency	string	nwcg_reporting_agency	nwcg_reporting_agency	string
nwcg_reporting_unit_id	string	nwcg_reporting_unit_id	nwcg_reporting_unit_id	string
nwcg_reporting_unit_name	string	nwcg_reporting_unit_name	nwcg_reporting_unit_name	string
source_reporting_unit	bigint	source_reporting_unit	source_reporting_unit	long
source_reporting_unit_nam	string	source_reporting_unit_name	source_reporting_unit_name	string
local_fire_report_id	bigint	local_fire_report_id	local_fire_report_id	long
local_incident_id	string	local_incident_id	local_incident_id	string
fire_code	string	fire_code	fire_code	string
fire_name	string	fire_name	fire_name	string
ics_209_incident_number	string	ics_209_incident_number	ics_209_incident_number	string
ics_209_name	string	ics_209_name	ics_209_name	string



Job: fire_etl1

Database Name fire1
Table Name glue_op_landing_zone1

Transform Name ApplyMapping

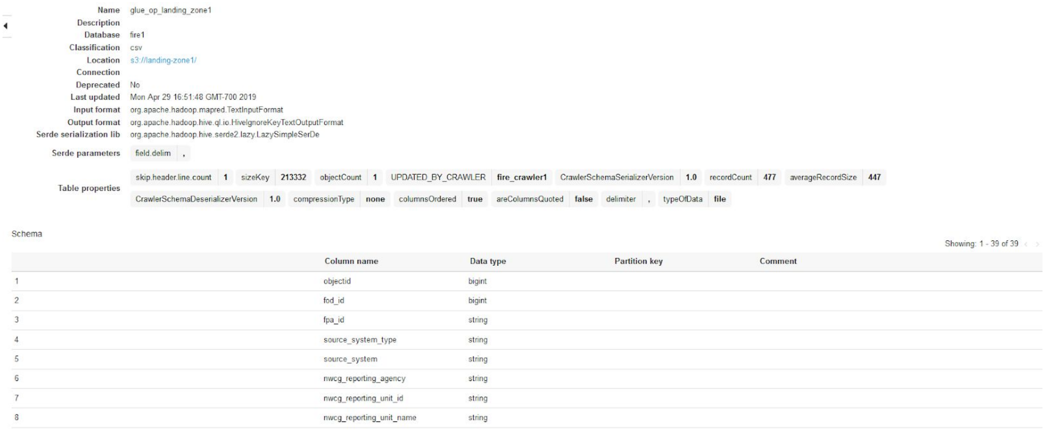
Path s3://staging-area1

```
import sys
from awsglue.transforms import *
from awsglue.utils import getResolvedOptions
from pyspark.context import SparkContext
from awsglue.context import GlueContext
from awsglue.job import Job

## @params: [JOB_NAME]
args = getResolvedOptions(sys.argv, ['JOB_NAME'])

sc = SparkContext()
glueContext = GlueContext(sc)
spark = glueContext.spark_session
job = Job(glueContext)
job.init(args['JOB_NAME'], args)
## @type: DataSource
## @args: [database = "fire1", table_name = "glue_op_landing_zone1", transformation_ctx = "datasource0"]
## @return: datasource0
## @inputs: []
datasource0 = glueContext.create_dynamic_frame.from_catalog(database = "fire1", table_name = "glue_op_landing_zone1", transformation_ctx = "datasource0")
## @type: ApplyMapping
## @args: [mapping = [("objectid", "long", "objectid", "long"), ("fod_id", "long", "fod_id", "long"), ("fpa_id", "string", "fpa_id", "string"), ("source_system_type", "string", "source_system_type", "string"), ("sour
## @return: applymapping1
## @inputs: [frame = datasource0]
applymapping1 = ApplyMapping.apply(frame = datasource0, mappings = [("objectid", "long", "objectid", "long"), ("fod_id", "long", "fod_id", "long"), ("fpa_id", "string", "fpa_id", "string"), ("source_system_type", "st
## @type: DataSink
## @args: [connection_type = "s3", connection_options = {"path": "s3://staging-area1"}, format = "csv", transformation_ctx = "datasink2"]
## @return: datasink2
## @inputs: [frame = applymapping1]
datasink2 = glueContext.write_dynamic_frame.from_options(frame = applymapping1, connection_type = "s3", connection_options = {"path": "s3://staging-area1"}, format = "csv", transformation_ctx = "datasink2")
job.commit()
```

5.4.3 Glue Table



Name	glue_op_landing_zone1
Description	
Database	fire1
Classification	csv
Location	s3://landing-zone1/
Connection	
Deprecated	No
Last updated	Mon Apr 29 16:51:48 GMT-700 2019
Input format	org.apache.hadoop.mapred.TextInputFormat
Output format	org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat
Serde serialization lib	org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe
Serde parameters	field.delim ,
Table properties	skip.header.line.count 1 sizeKey 213332 objectCount 1 UPDATED_BY_CRAWLER fire_crawler1 CrawlerSchemaSerializerVersion 1.0 recordCount 477 averageRecordSize 447 CrawlerSchemaDeserializerVersion 1.0 compressionType none columnsOrdered true areColumnsQuoted false delimiter , typeOfData file

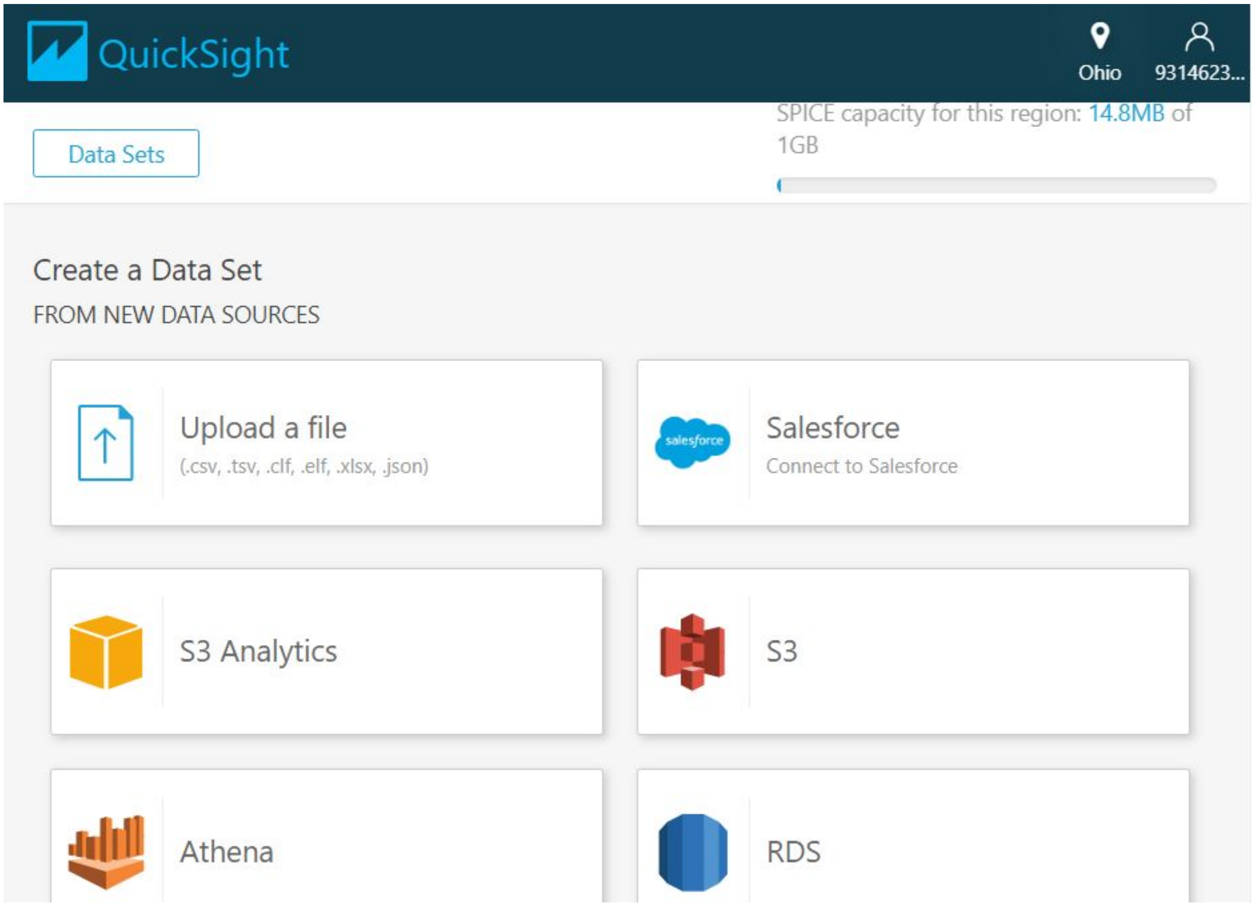
Schema

Showing: 1 - 39 of 39

	Column name	Data type	Partition key	Comment
1	objectid	bigint		
2	fod_id	bigint		
3	fpa_id	string		
4	source_system_type	string		
5	source_system	string		
6	nwcg_reporting_agency	string		
7	nwcg_reporting_unit_id	string		
8	nwcg_reporting_unit_name	string		

5.5 QuickSight

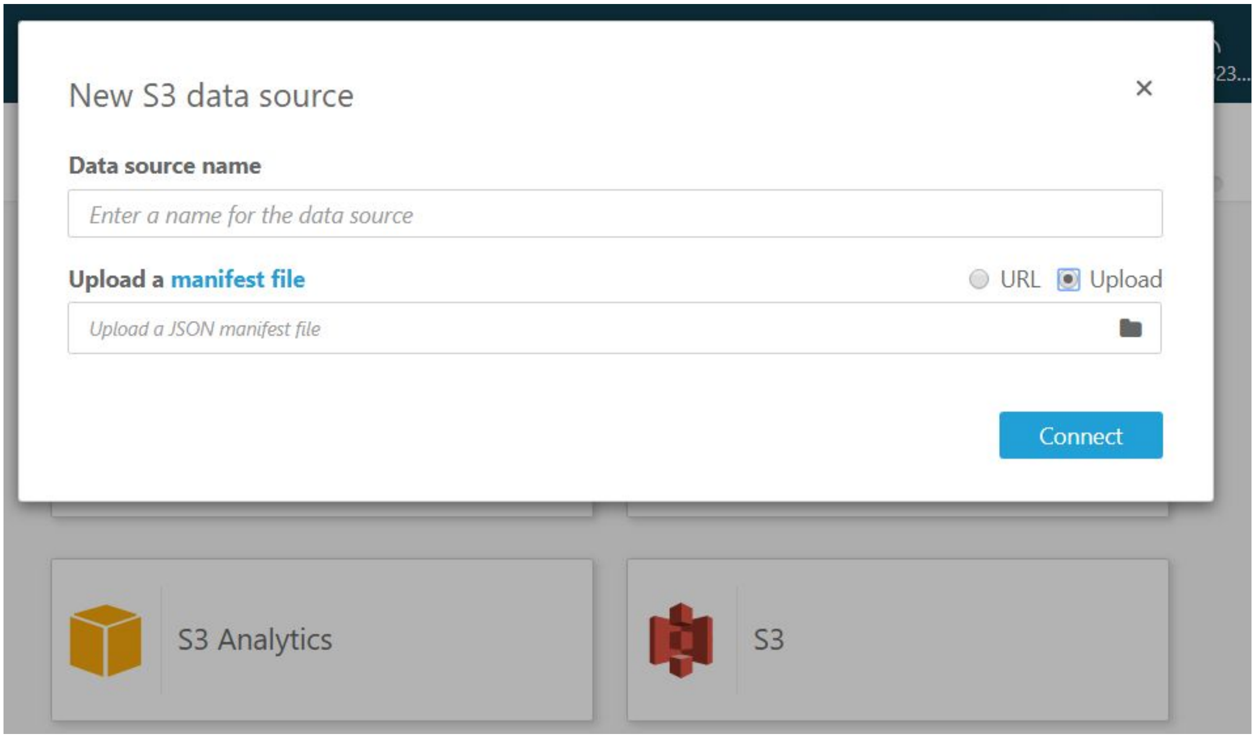
Connect an S3 bucket to QuickSight



SPICE capacity for this region: 14.8MB of 1GB

Create a Data Set

FROM NEW DATA SOURCES



New S3 data source

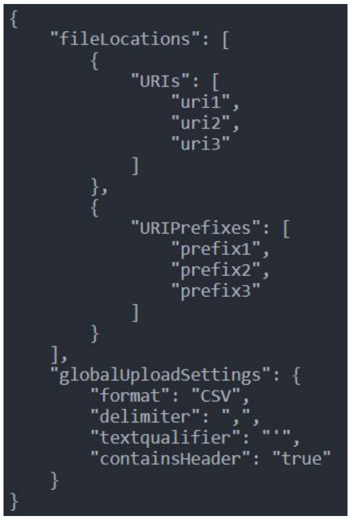
Data source name

Enter a name for the data source

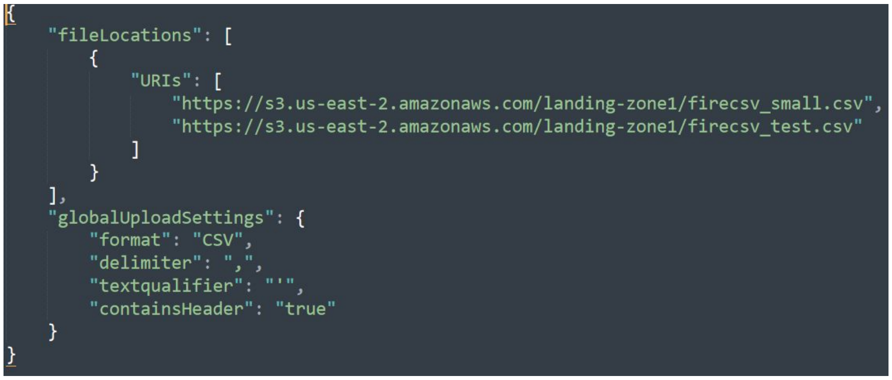
Upload a manifest file ○ URL ● Upload

Upload a JSON manifest file

Manifest JSON file template and S3 file specifications



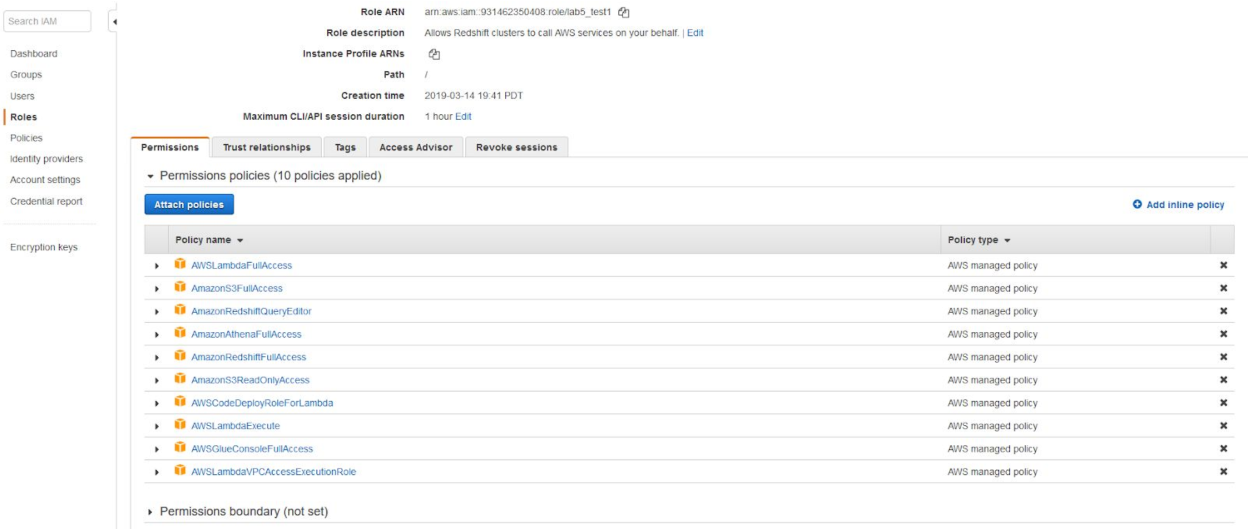
```
{
    "fileLocations": [
        {
            "URIs": [
                "uri1",
                "uri2",
                "uri3"
            ]
        },
        {
            "URIPrefixes": [
                "prefix1",
                "prefix2",
                "prefix3"
            ]
        }
    ],
    "globalUploadSettings": {
        "format": "CSV",
        "delimiter": ",",
        "textqualifier": "'",
        "containsHeader": "true"
    }
}
```



```
{
    "fileLocations": [
        {
            "URIs": [
                "https://s3.us-east-2.amazonaws.com/landing-zone1/firecsv_small.csv",
                "https://s3.us-east-2.amazonaws.com/landing-zone1/firecsv_test.csv"
            ]
        }
    ],
    "globalUploadSettings": {
        "format": "CSV",
        "delimiter": ",",
        "textqualifier": "'",
        "containsHeader": "true"
    }
}
```

5.6 IAM

Policies:



Role ARN arn:aws:iam::931462350408:role/lab5_test1

Role description Allows Redshift clusters to call AWS services on your behalf. | Edit

Instance Profile ARNs

Path /

Creation time 2019-03-14 19:41 PDT

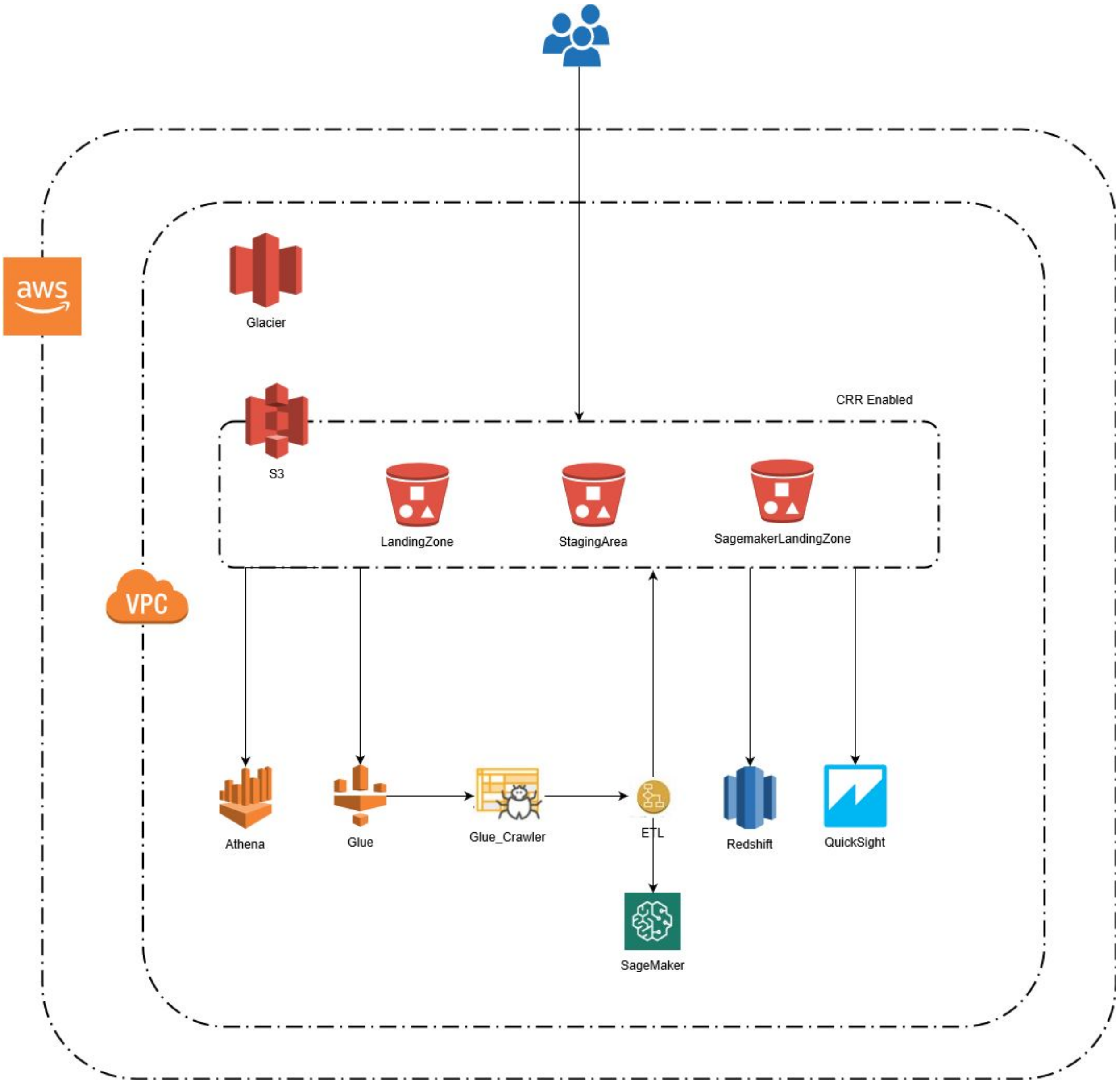
Maximum CLI/API session duration 1 hour Edit

Permissions policies (10 policies applied)

Policy name	Policy type	
AWSLambdaFullAccess	AWS managed policy	✖
AmazonS3FullAccess	AWS managed policy	✖
AmazonRedshiftQueryEditor	AWS managed policy	✖
AmazonAthenaFullAccess	AWS managed policy	✖
AmazonRedshiftFullAccess	AWS managed policy	✖
AmazonS3ReadOnlyAccess	AWS managed policy	✖
AWSCodeDeployRoleForLambda	AWS managed policy	✖
AWSLambdaExecute	AWS managed policy	✖
AWSGlueConsoleFullAccess	AWS managed policy	✖
AWSLambdaVPCAccessExecutionRole	AWS managed policy	✖

Permissions boundary (not set)

6. Solution Architecture



- The main aim of the project is to develop a model which can identify and categorize the various fire incidents based on their cause.
- The data file is first uploaded onto the S3 bucket.
- We use Athena and Glue (using Glue crawlers) to read the data from the S3 buckets.
- Since the initial data has multiple columns, we perform feature engineering to extract only the required columns
- The data is cleaned using ETL functions and then the data is sent to the SageMaker where it is used to train the model

- A model has been developed using Python libraries to categorize the various fire causes.
- We are using QuickSight to analyze the data from the S3 buckets and create dashboards presenting the data in charts showing the various areas affected by fire. We created histograms depicting various information.
- We are using three S3 buckets, each in different regions to increase availability and reliability.

7. URL to GitHub.

https://github.com/k-chuang/aws-forest-fire-predictive-analytics

8. Future Scope

- Using Lambda to automate the flow so that as soon as the document lands in the S3 landing area, the other functions such as ETL jobs and training models are triggered automatically.
- Online Training (i.e. training on the fly for data as it comes in) for SageMaker
- Create an interactive frontend for the user to upload the document for which analysis is wanted.
- The user can also query the system via the frontend.