Set-Up Instructions

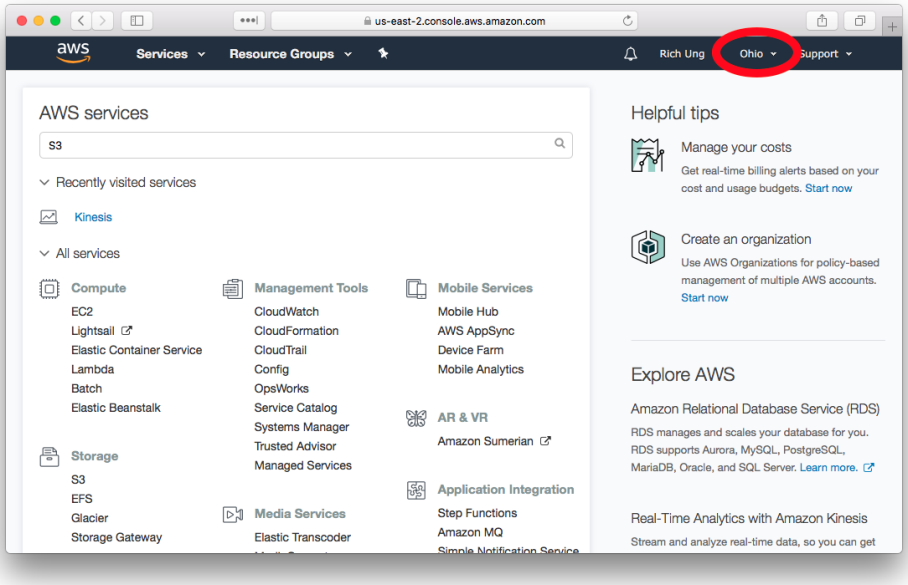
Signing up for AWS

If you don't already have an AWS account, you must sign up for one. If you already have an account, you can skip this prerequisite and use your existing account.

1. Open https://aws.amazon.com/, and then choose Create an AWS Account.
    a. Note: This might be unavailable in your browser if you previously signed into the AWS Management Console. In that case, choose Sign In to the Console, and then choose Create a new AWS account.
2. Follow the online instructions.
    a. Part of the sign-up procedure involves receiving a phone call and entering a PIN using the phone keypad.

Things to Note:

- All your services and work should be located within the same region. Make sure the selected region is consistent throughout this document, otherwise connection and auto fill-in issues might surface up.

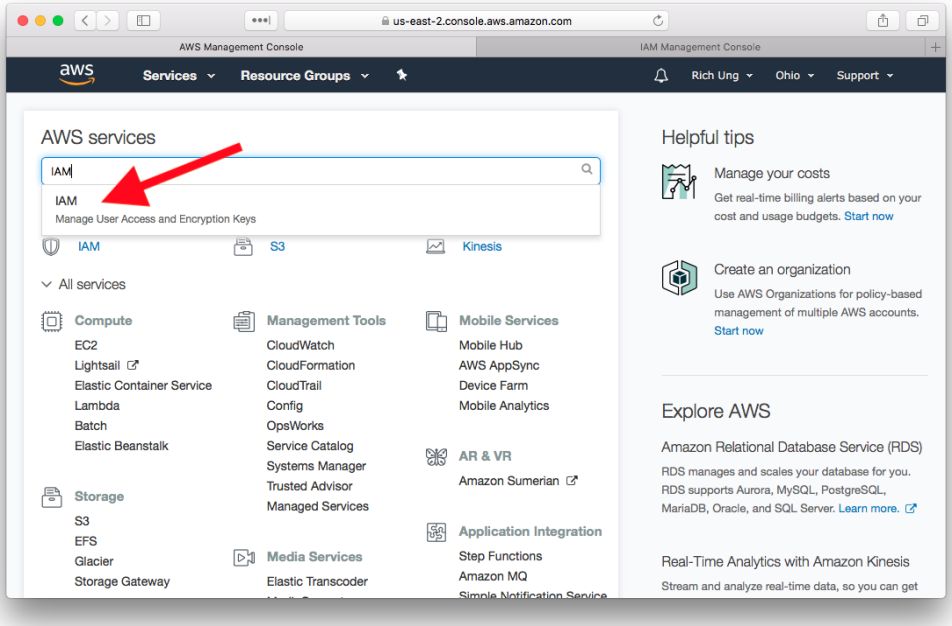


    ○

Table of Contents

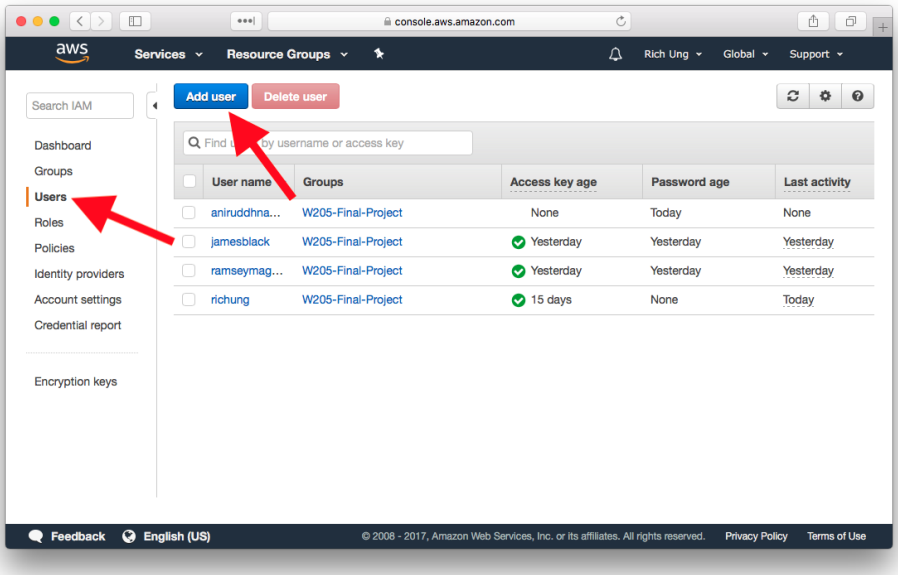
Create an IAM User to grant applications to your AWS Account

1. Log into your AWS Console and select the IAM service



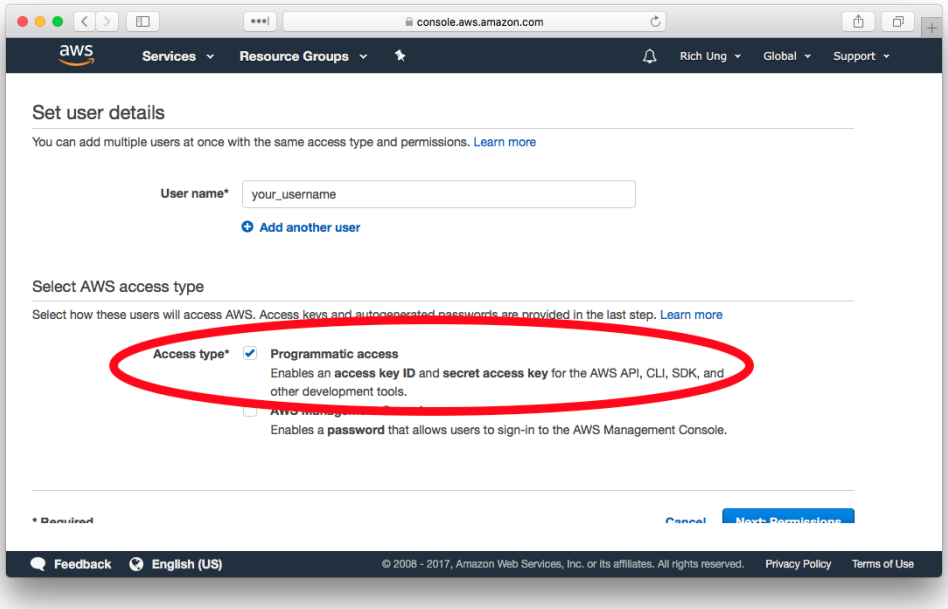
a.

2. Under the “Users” tab select “Add user”



a.

3. Enter a username, and under “Access type” make sure “Programmatic access” is checked.

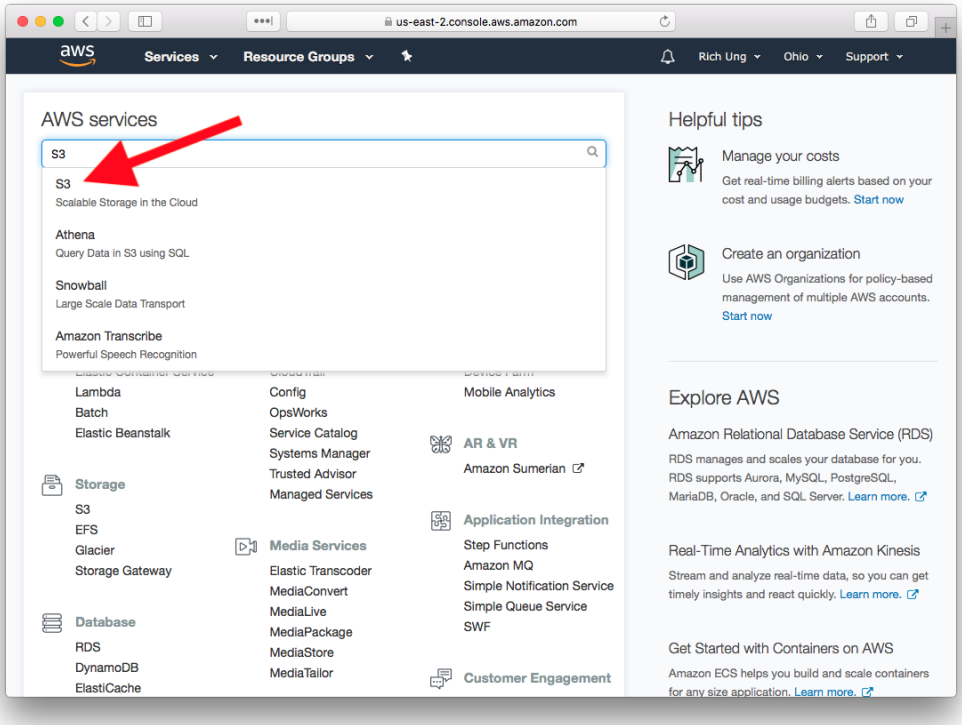


a.

4. Select “Next: Permissions”
5. Attach the following policies to the user (either through creating a group with the attached policies, or attaching existing policies directly to the user)
    a. AWSLambdaFullAccess
    b. AmazonS3FullAccess
    c. AmazonRedshiftFullAccess
    d. AmazonKinesisFullAccess
    e. AmazonKinesisVideoStreamsFullAccess
    f. AmazonKinesisFirehoseFullAccess
    g. AmazonKinesisAnalyticsFullAccess
6. Select “Next: Review”
7. Verify information is correct and select “Create user”
8. Record your Access Key ID and Secret Access Key. If you need to create a new Access Key ID, select your username under the “Users” tab, select the “Security credentials” tab, and select “Create access key” under the “Access keys” section.

Create a S3 bucket to hold your data

1. Log into your AWS Console and select the S3 service



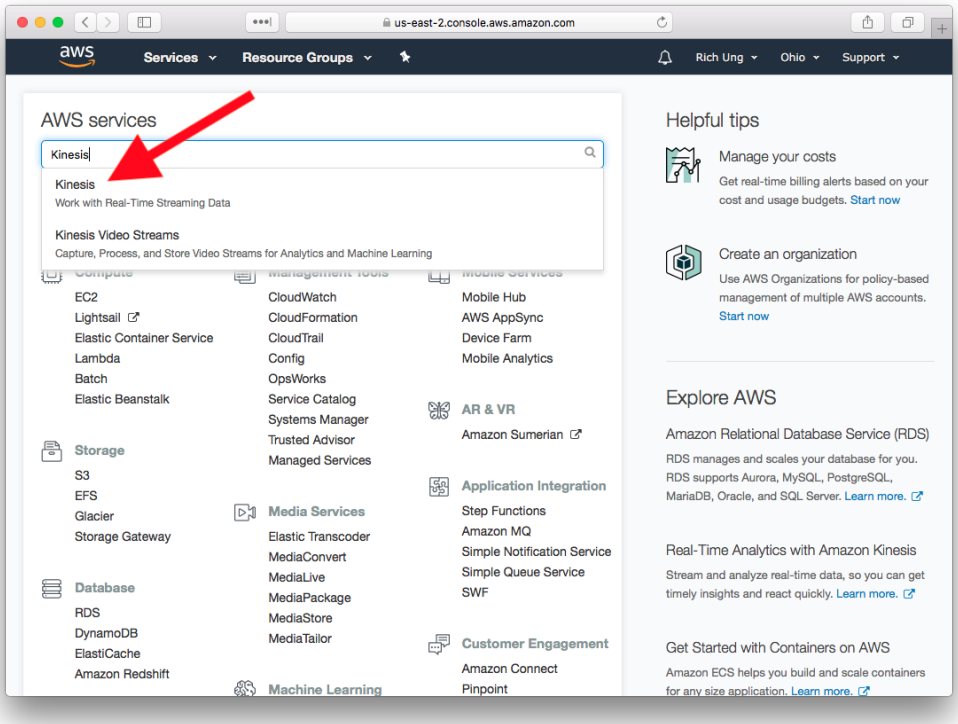
    a.
2. Select “Create bucket”
3. Enter a name and region
    a. Select the same region where all your other provisioned services are located
    b. Add the bucket name and **<u>take a note of it</u>**, to be used later for configuration.
4. Select “Next”
5. Select “Next” (Using default properties under the “Set properties” step)
6. Select “Next” (Using default properties under the “Set permissions” step)
7. Select “Next”
8. Review settings and select “Create bucket”

Create a Amazon Redshift cluster

- Please follow the directions located within the “Get Started Guide”:
    - http://docs.aws.amazon.com/redshift/latest/gsg/getting-started.html
- Note: There is also a free trial for new users if you would like to minimize charges
    - https://aws.amazon.com/redshift/free-trial/

Create a Kinesis Data Stream

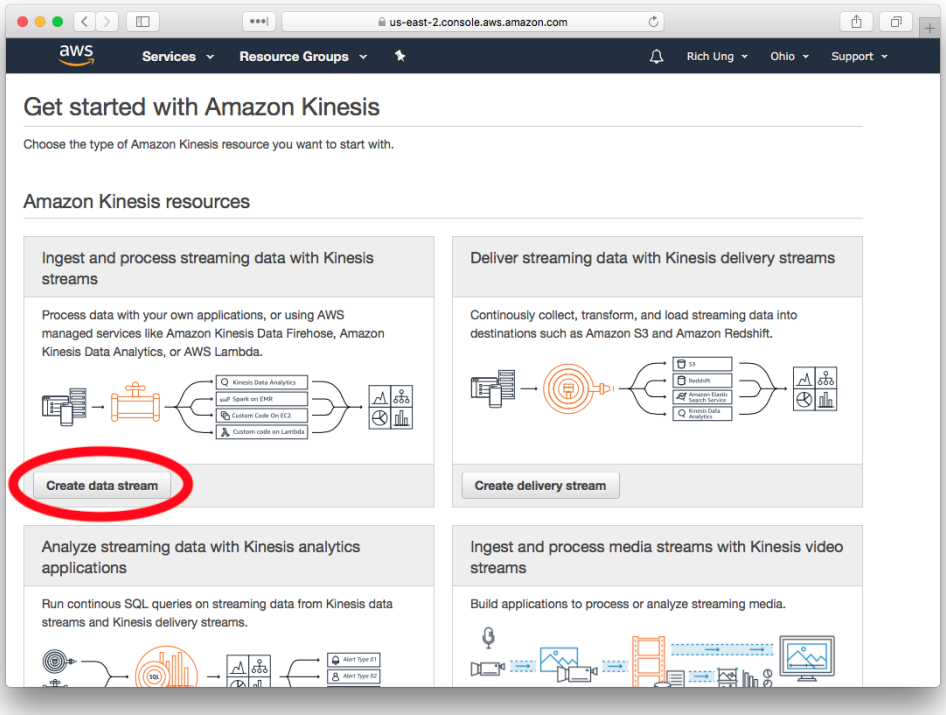
1. Log into your AWS Console and select the Kinesis service



   a.

   Note: Click “Get Started” Button for first time use of Kineses console

2. Under “Ingest and process streaming data with Kinesis streams” (or under “Kinesis data streams” if this is not your first Kinesis data stream), select “Create data stream”



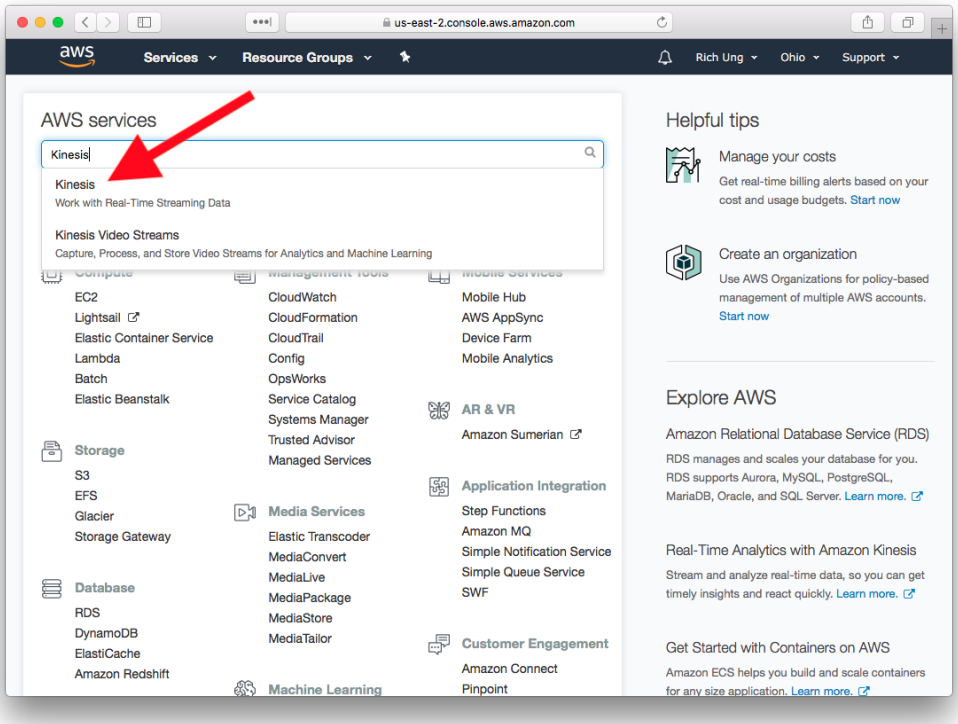
a.

b. Enter the following information
    i. Kinesis stream name: W205
    ii. Number of shards: 1

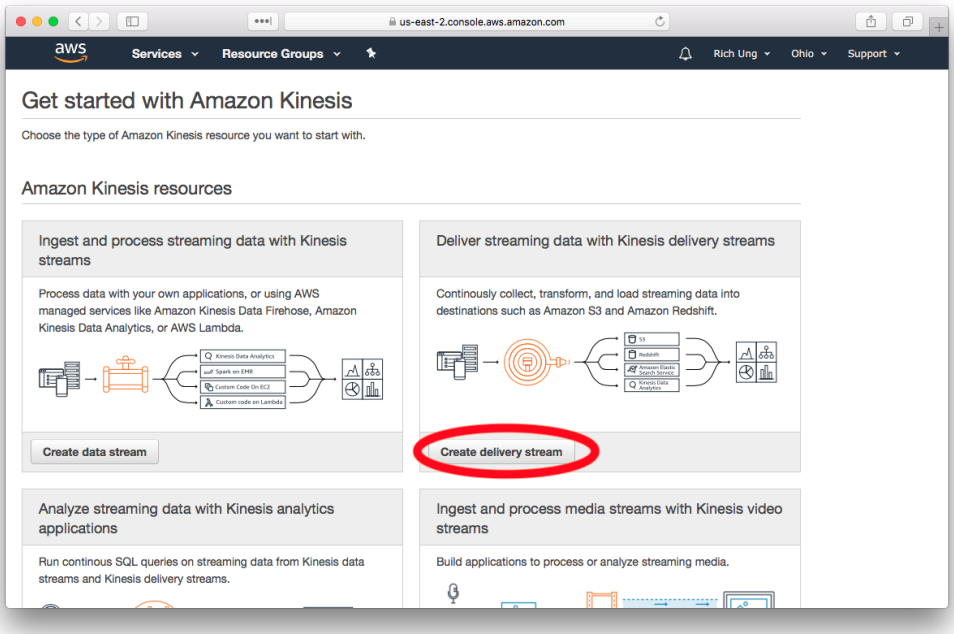
c. Click on “Create Kinesis stream”

Create a Kinesis Delivery Stream to S3

1. Log into your AWS Console and select the Kinesis service

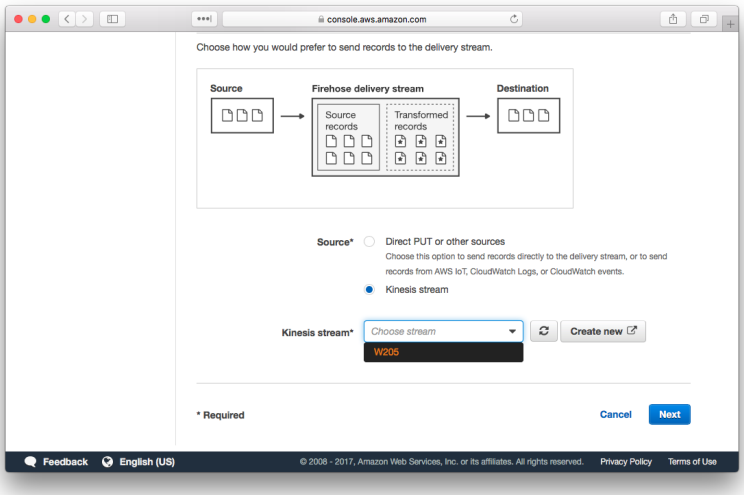


   a.
2. Under “Deliver streaming data with Kinesis delivery streams” (or under “Kinesis delivery streams” if this is not your first Kinesis delivery stream), select “Create delivery stream”



a.

3. Enter the following information:
    a. Under “Delivery stream name”, enter a name for your delivery stream to S3, for example “W205-to-S3”.
    b. Under “Source”, select “Kinesis stream”
        i. A new “Kinesis stream” section should appear. Select the Kinesis stream that you created in the previous section

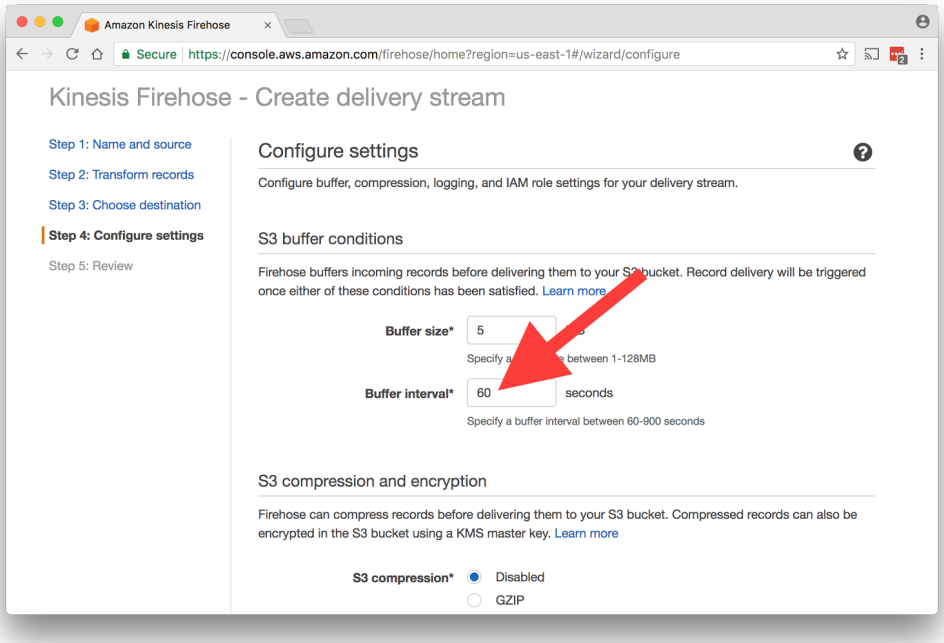


1.

4. Select “Next”
5. Leave “Record transformation” Disabled. Select “Next.”
6. Enter the following information:
    a. Under “Destination”, select “Amazon S3”
    b. Under “S3 bucket”, select the S3 bucket you created in the previous section.

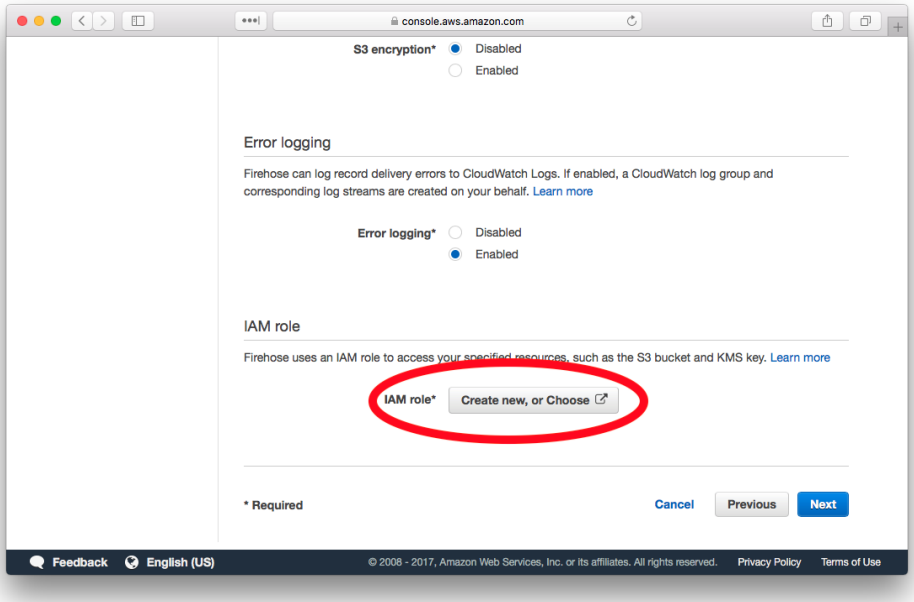
c. Prefix is optional, feel free to add a prefix if you would like.

7. Select “Next”
8. Change “Buffer interval” to 60 seconds.

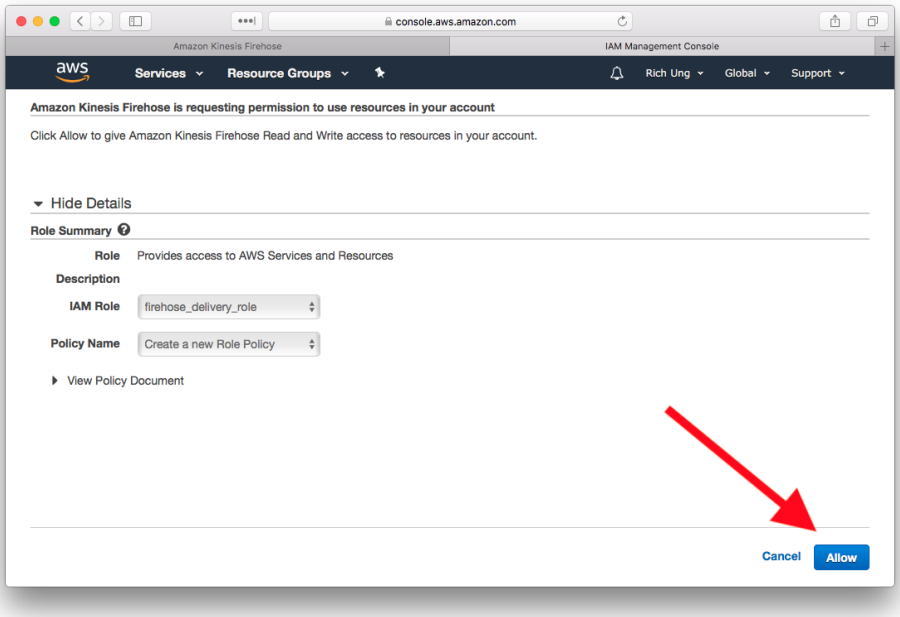


a.

9. Under “IAM role”, select “Create new, or Choose”:



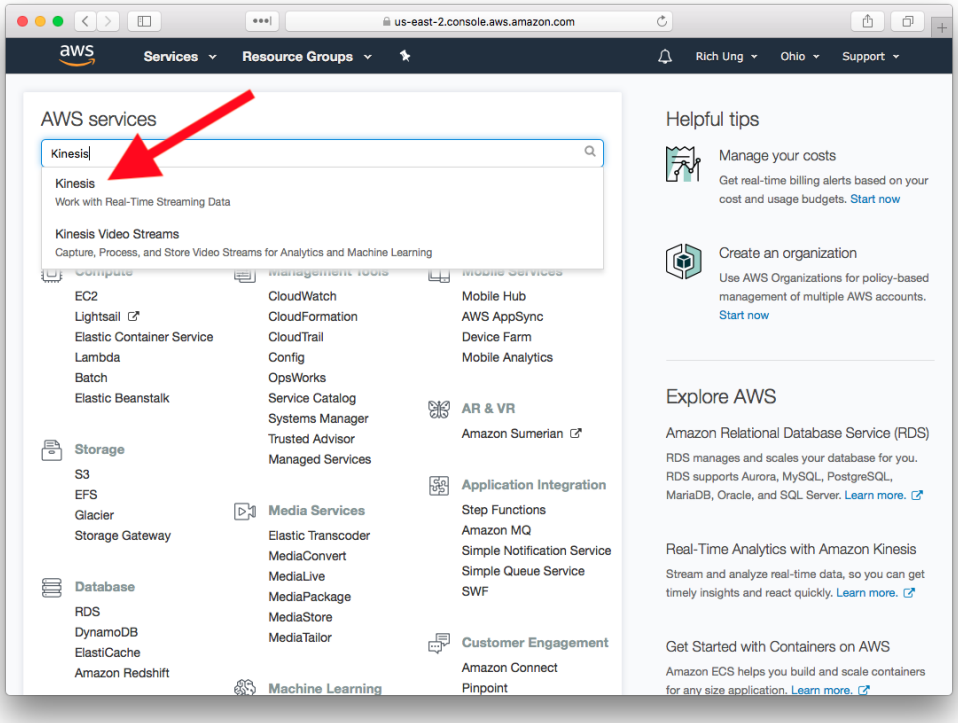
10. Leave default settings and select “Allow”, creating an IAM Role or new Policy Name with default configurations if you are not allowed to select “Allow”:



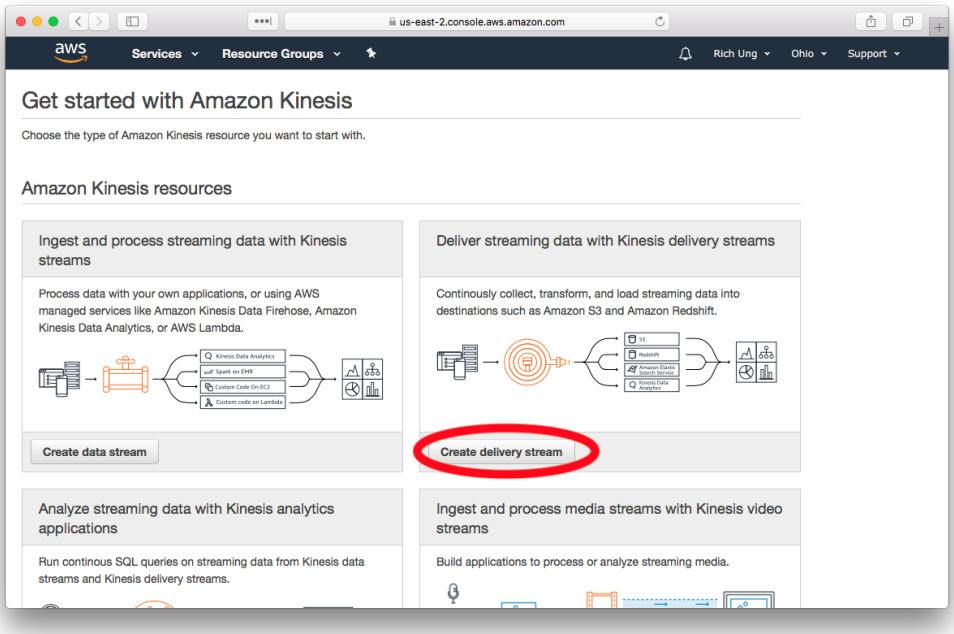
11. Select “Next”
12. Review information is correct and select “Create delivery stream”.

Create a Kinesis Delivery Stream to Redshift

1. Log into your AWS Console and select the Kinesis service

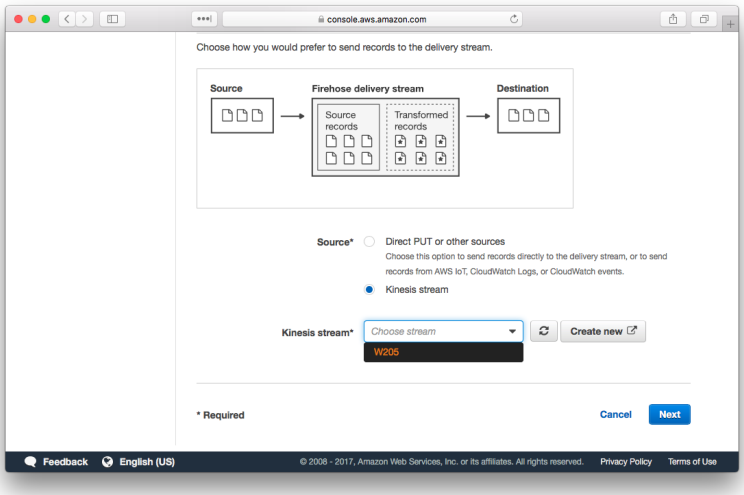


   a.
2. Under “Deliver streaming data with Kinesis delivery streams” (or under “Kinesis delivery streams” this is not your first Kinesis delivery stream), select “Create delivery stream”



a.

3. Enter the following information:
    a. Under “Delivery stream name”, enter a name for your delivery stream to Redshift, for example “W205-to-Redshift”.
    b. Under “Source”, select “Kinesis stream”
        i. A new “Kinesis stream” section should appear. Select the Kinesis stream that you created in the previous section

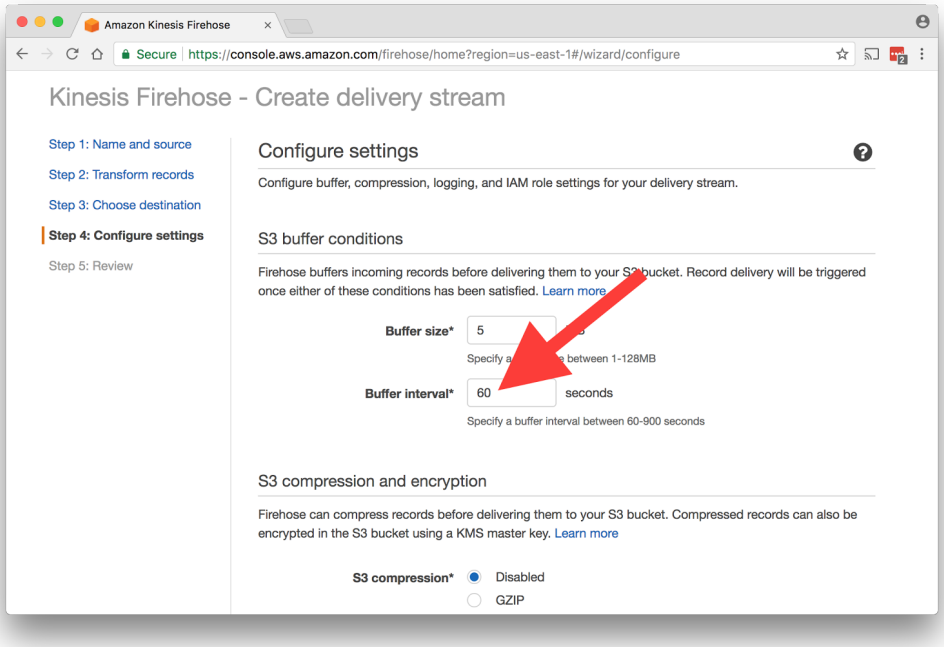


1.

4. Select “Next”
5. Leave “Record transformation” Disabled. Select “Next.”
6. Enter the following information:
    a. Under “Destination”, select “Amazon Redshift”
    b. Under “Cluster,” select the Redshift cluster you created in the previous section

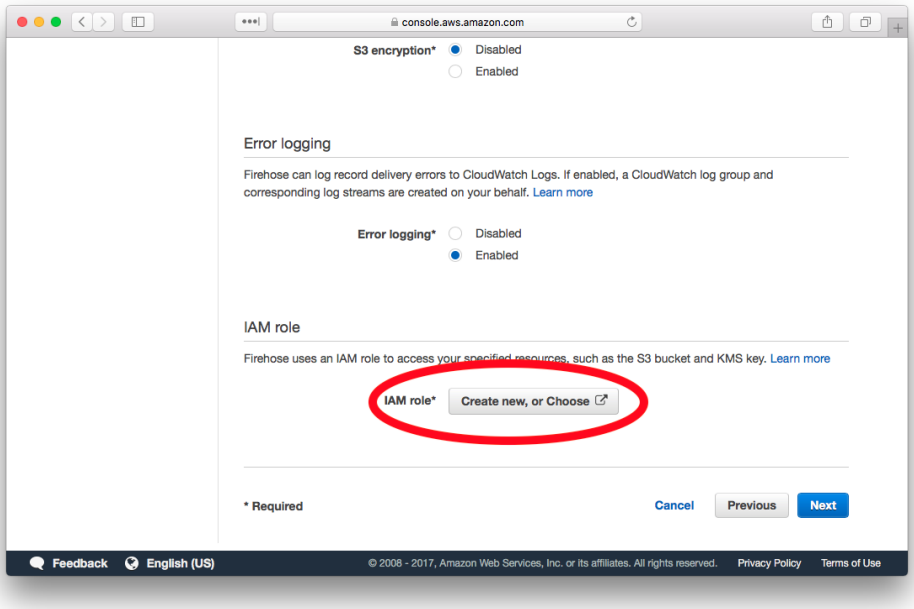
c. Enter the master username, password and the database name you created in the previous section when creating the Redshift cluster
d. Add the table name as: data_feed
e. Under “Intermediate S3 bucket”, select “Create new”.
    i. Enter a S3 bucket name for your intermediate S3 bucket and select the same region as where your other services are located.
    ii. Select “Create S3 bucket”

7. Select “Next”
8. Change “Buffer interval” to 60 seconds.



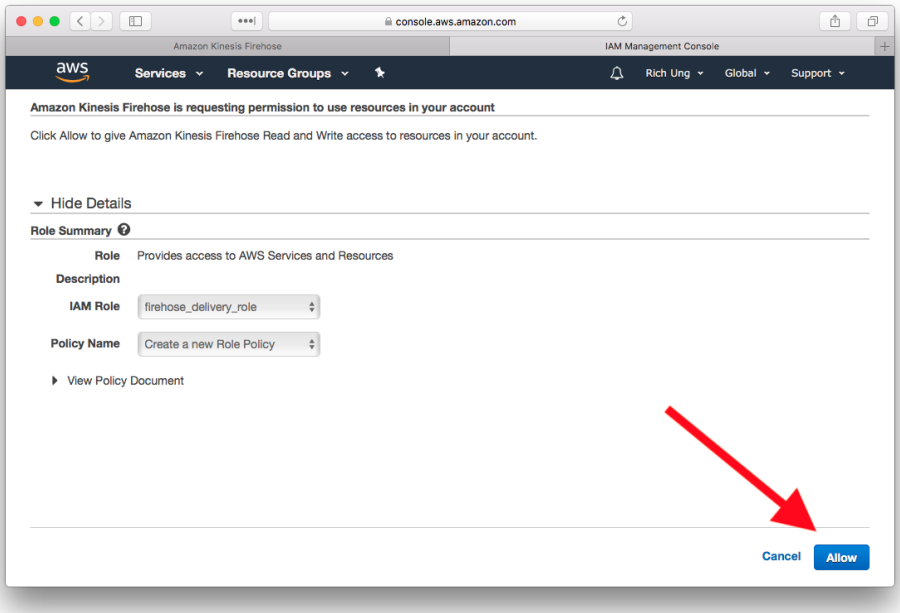
a.

9. Under “IAM role” select “Create new, or Choose”:



a.

10. Leave default settings and select “Allow”, creating an IAM Role or new Policy Name with default configurations if you are not allowed to select “Allow”:

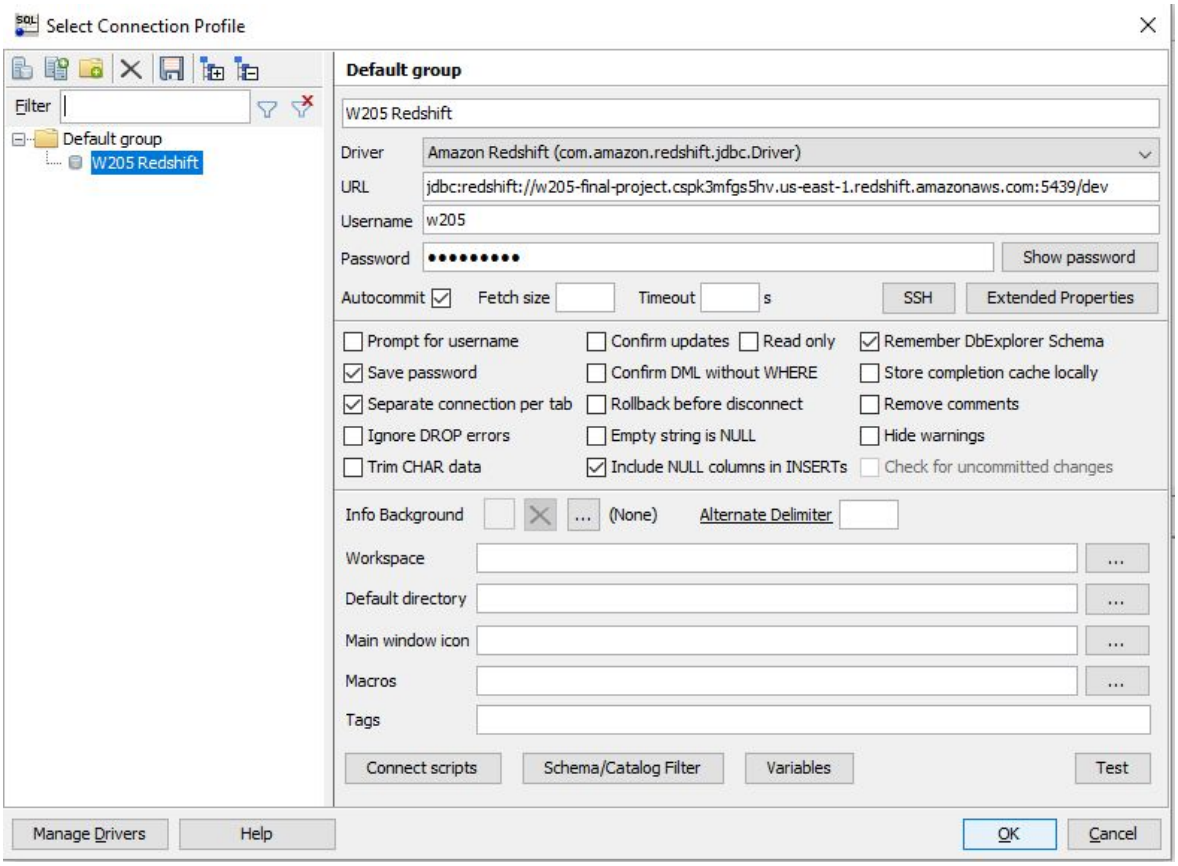


a.

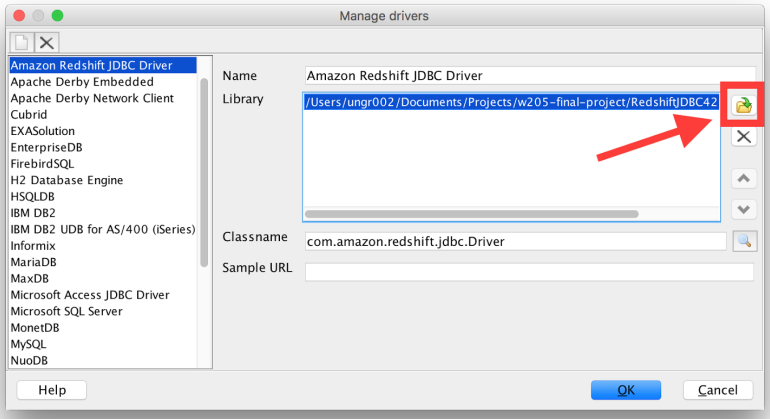
11. Select “Next”
12. Review information is correct and select “Create delivery stream”.

Create Redshift Table Structure

1. Install SQLWorkbench/J, a cross platform SQL query tool used to configure the Redshift database structure
    a. Download available at http://www.sql-workbench.net/
2. Install an Amazon Redshift JDBC driver to enable SQL Workbench/J to connect to your cluster
    a. Download available at http://docs.aws.amazon.com/redshift/latest/mgmt/configure-jdbc-connection.html#download-jdbc-driver
    b. For the purpose of this project, the suitable driver has been stored within the GitHub repository
3. Launch SQLWorkbench/J and create a connection profile to access your Redshift cluster using JDBC
    a. Example connection profile



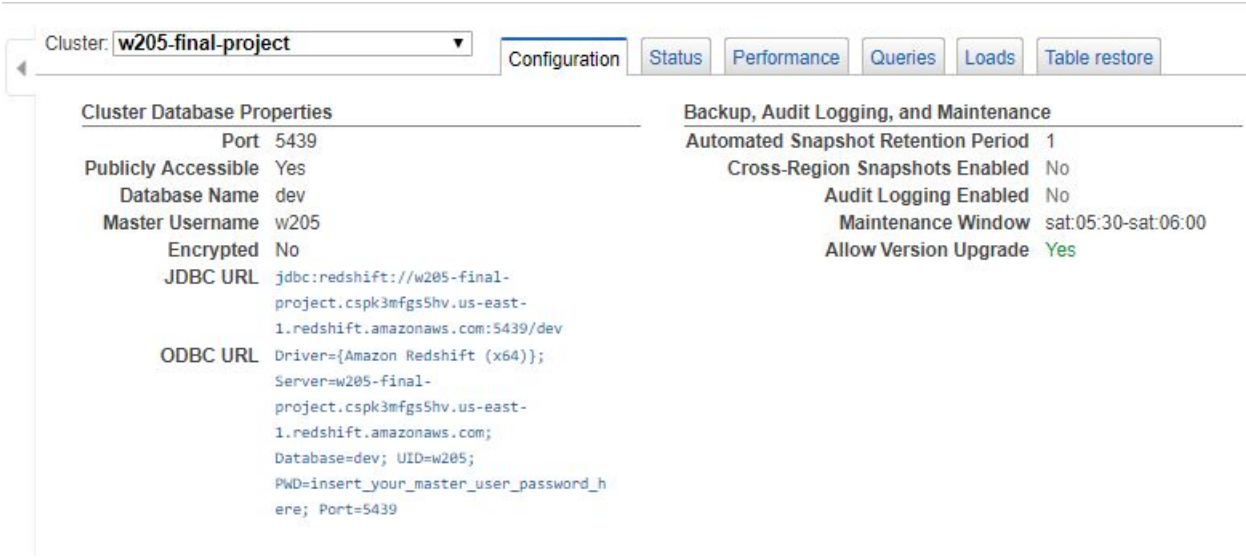
    b. Import the Amazon Redshift JDBC driver (either previously downloaded, or available in the project GitHub repository)
        i. Click on “Manage Drivers” on the lower left corner
        ii. Select the driver by clicking on the open folder icon (see screenshot below) and navigating to the downloaded JDBC driver



1.

iii. Click “OK”

c. Enter the Redshift JDBC URL, found on your Redshift cluster’s “Cluster Database Properties” details



d. Enter your previously provisioned credentials (or in the case of this group project, use credentials available in the setup materials and README)

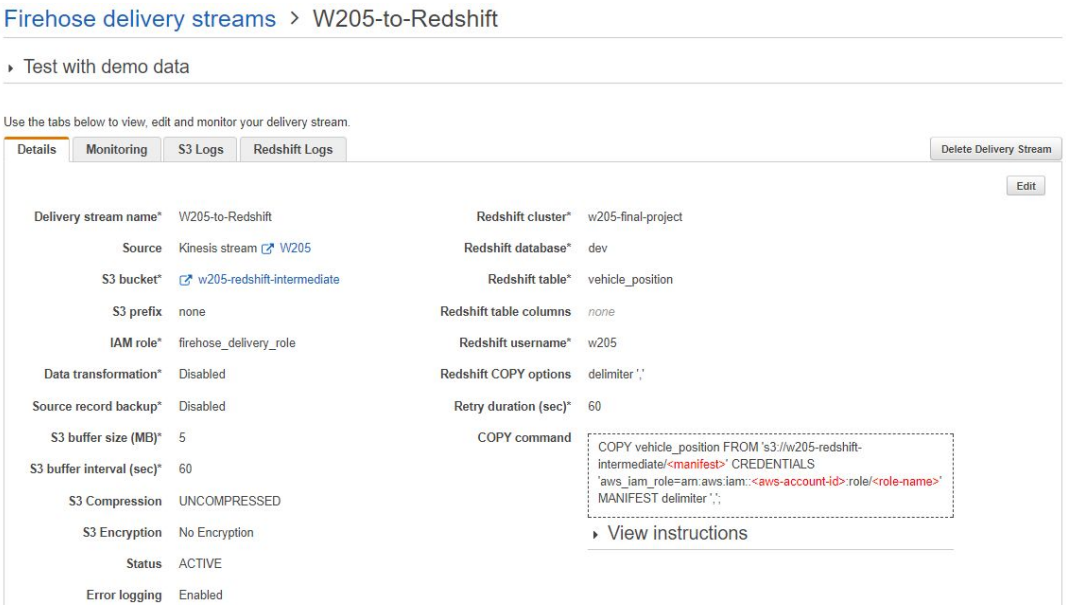
e. Test the connection. If the test fails, please refer to the SQLWorkbench/J site for additional troubleshooting. If the connection is successful, we recommend saving the profile before clicking “OK” within the “Select Connection Profile” wizard

4. Once your SQLWorkbench/J successfully establishes a connection to your Redshift cluster, a workspace will be created that allows you to execute SQL commands on your Redshift cluster

a. For the purposes of this workflow, a number of table creation statements have been added to the GitHub repository, under the “redshift_sql_scripts” directory

b. Please execute each set of table creation statements to create the appropriate tables. These can be copied and pasted into the workspace and run by pressing Ctrl+E, or by clicking the “execute” arrow.

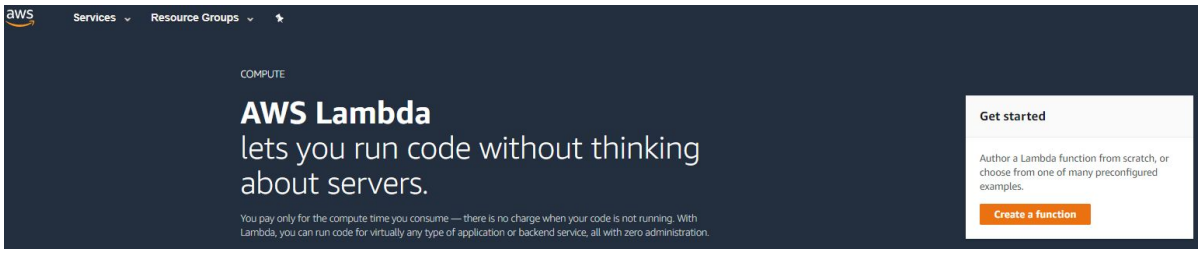
5. Note that your Kinesis Firehose delivery stream may need to be modified to match the Redshift table name and source data delimiter for successful ingestion from Kinesis to Redshift
    a. For example, the W205-to-Redshift Kinesis Firehose delivery stream was modified to refer to the created “vehicle_position” Redshift table, and was updated to identify the use of comma delimited data:



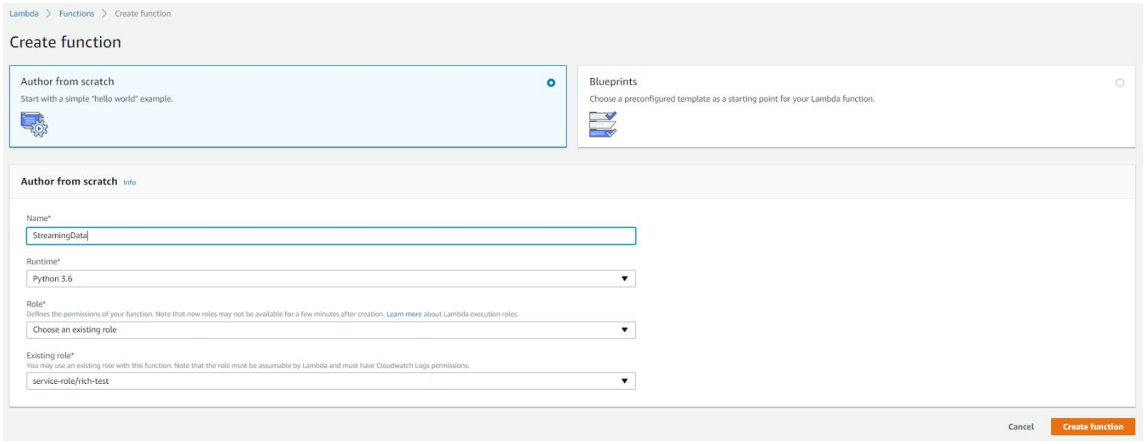
    b. Additionally, the COPY command available on the relevant Firehose delivery stream details page can be used to manually verify the ability to send data from S3 into Redshift. You may need to update the code to include relevant IAM credentials. Note, once properly configured, Kinesis will automatically pipe ingested data into the Redshift table.

Create Lambda Function

1. Amazon Lambda is a serverless compute function designed to execute code. For the purpose of this project, a package was created to pull streaming data via API call and push it into the Kinesis pipeline.
2. Navigate to Lambda (https://console.aws.amazon.com/lambda) and click “Create a Function”

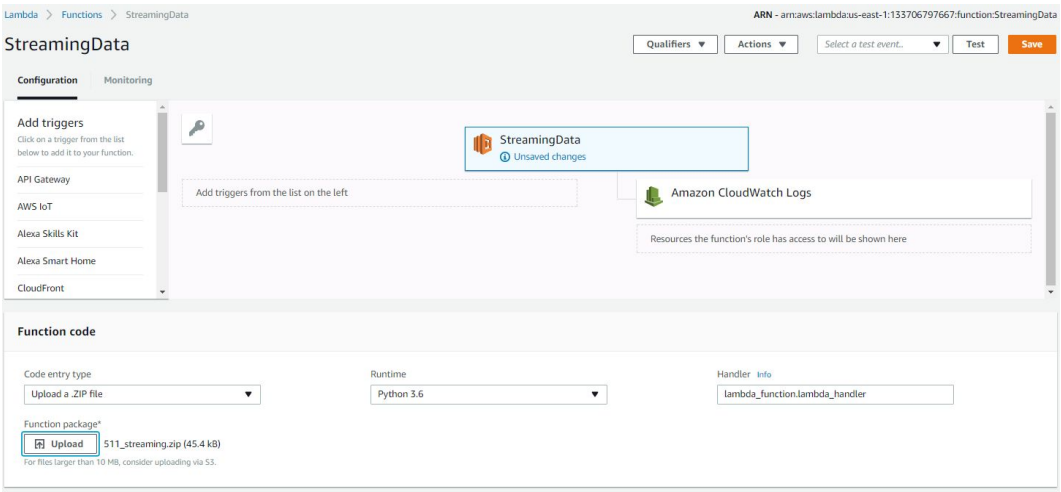


3. A “Create function” window will open. Populate the following fields:
    a. Name (e.g. create a function name)
    b. Runtime (e.g. identify the package as Python 3.x)
    c. Role (e.g. assign permissions to the function)
    d. Existing role (e.g. elect to use an existing IAM role)
    e.

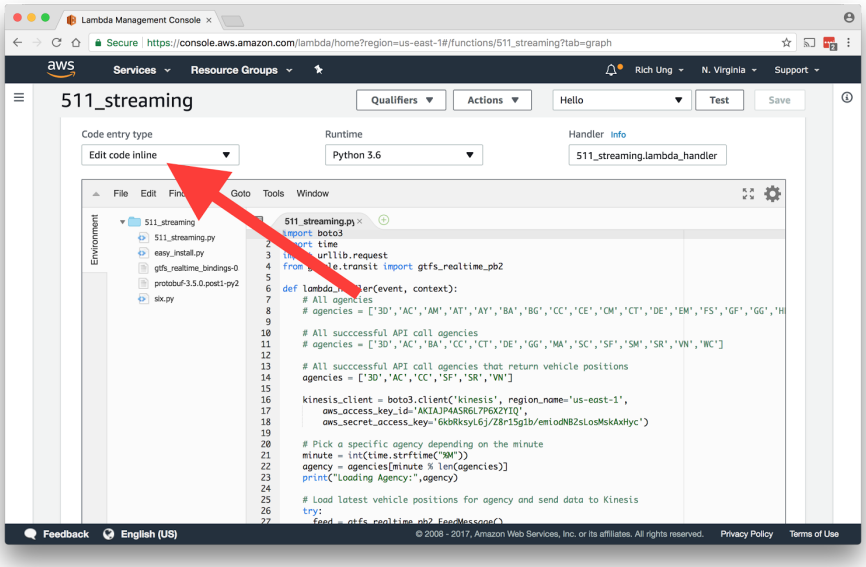


4. Once the function has been created, change “Code entry type” to “Upload a .ZIP file” and upload the project package zip file (titled “511_streaming.zip” within the

“Lambda_function” folder):

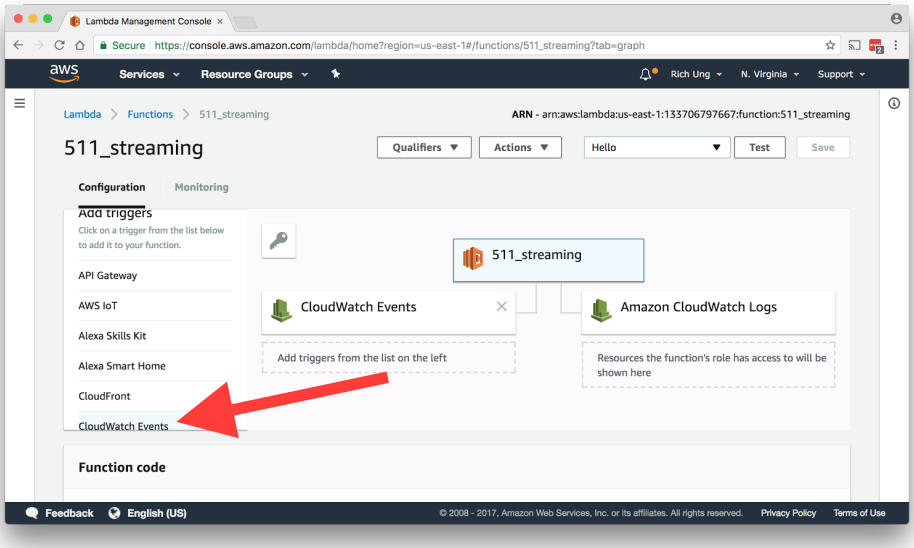


5. Click “Save” to save the ZIP file into the Lambda function.
6. Click “Test” to verify correct execution.
    a. Note: After uploading the zip file into the Lambda function, you can edit the zip file’s contents by changing the “Code entry type” from “Upload a ZIP file” back to “Edit code inline”.



    i.

7. Add a Cloudwatch Event Trigger by clicking on “CloudWatch Events” under the “Add triggers” section

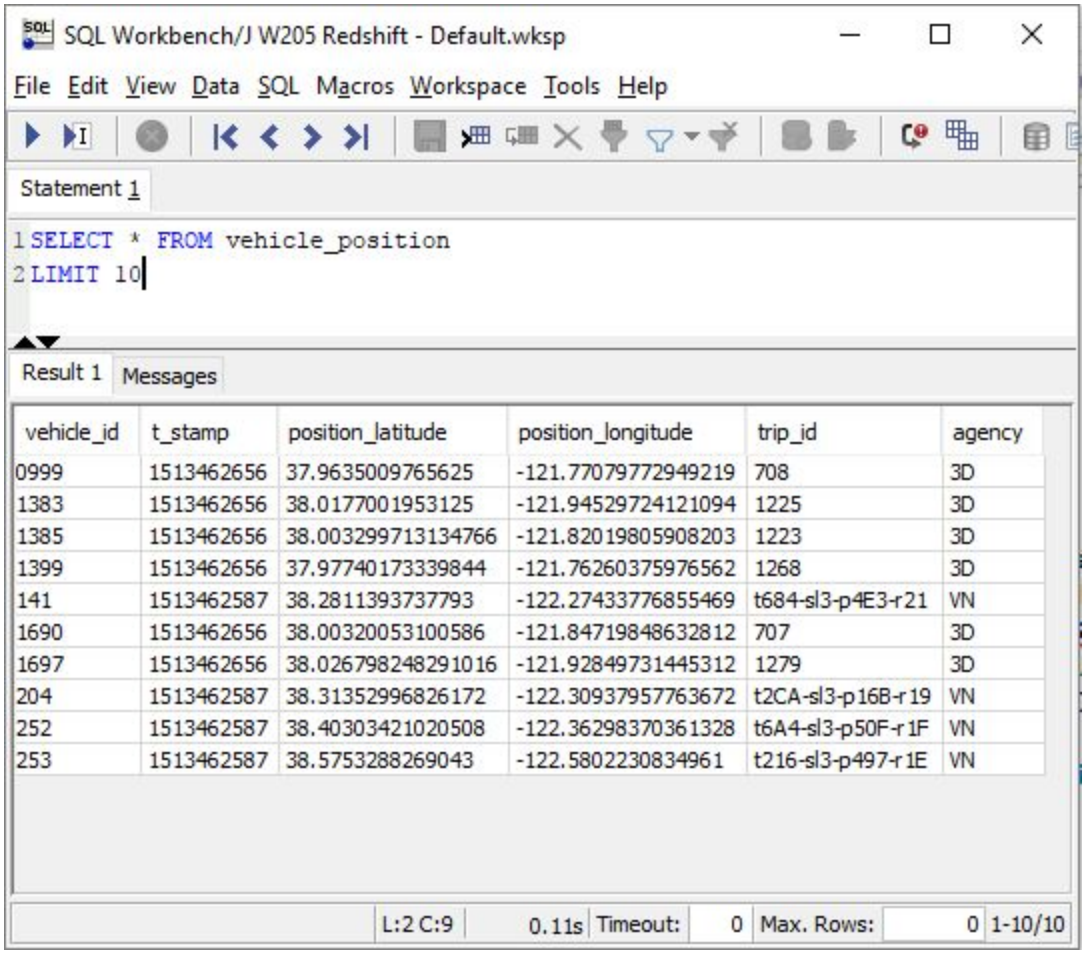


a.

b. Give a name for the CloudWatch event, and set the schedule to "**rate(1 minute)**". This will execute the Lambda script every minute, which will push data from the 511 SF API into Kinesis.
c. Enable the CloudWatch Event to automate the execution of the script every minute.

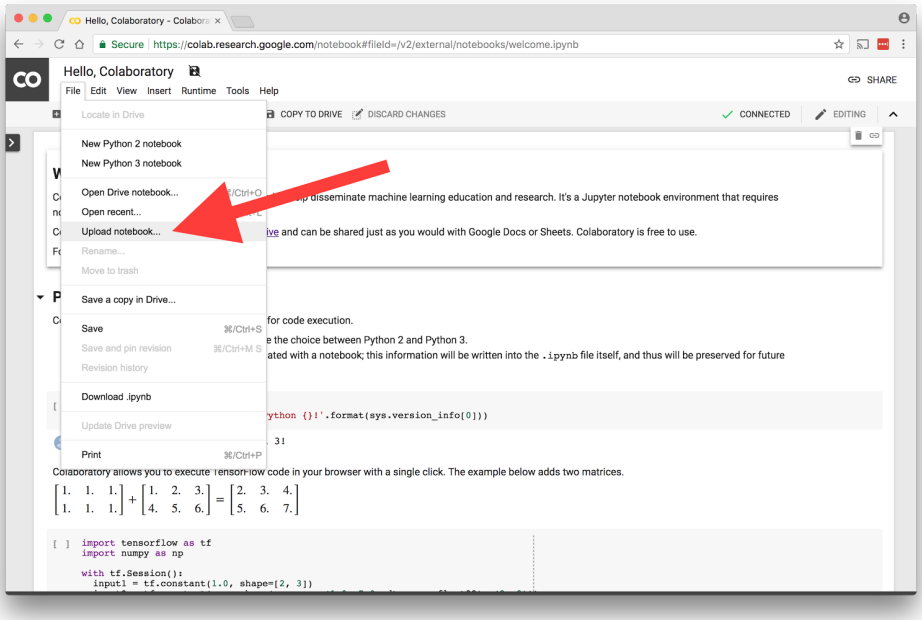
8. Once data has been piped into Kinesis, you can verify successful ingestion into S3 by examining S3 logs, and Redshift by executing SQL commands to return results. For example, the following screenshot shows a subsequent command to return 10 results

from the vehicle_position table within Redshift:



Add static data into S3 and Redshift

1. Create a 511 API Access Key Token
    a. Go to https://511.org/developers/list/tokens/create and create a token by filling out the form.
2. Upload the Jupyter notebook, “W205 Final Project.ipynb”, located within the “load_static_datasets” folder, onto Colaboratory
    a. https://colab.research.google.com/
        i. This is a free Google product that lets you run Python code on Google’s servers
            1. Note: This notebook uses Python 3



    b.
3. Make the following changes to the script:
    a. Change **W205-redshift-intermediate** to the name of your S3 bucket created in “Create a S3 bucket to hold your data”
    b. Change **ACCESS_KEY = 'AKIAJP4ASR6L7P6X2YIQ'** and **SECRET_KEY = '6kbRksyL6j/Z8r15g1b/emiodNB2sLosMskAxHyc'** to the access key and secret key generated when you created your IAM user in “Create an IAM User to grant applications to your AWS Account”
    c. Change **baa045f5-dff4-44f4-ad59-8d50f70b12ad** and **c446f9f0-5979-4667-a37b-d31b41480fa9** to your API Access Key obtained in step 1.
    d. Change the following lines of code to your own connection information for redshift
        i. **thisdb = "dev"**
        ii. **thisuser = "w205"**
        iii. **thispassword = "W205final"**

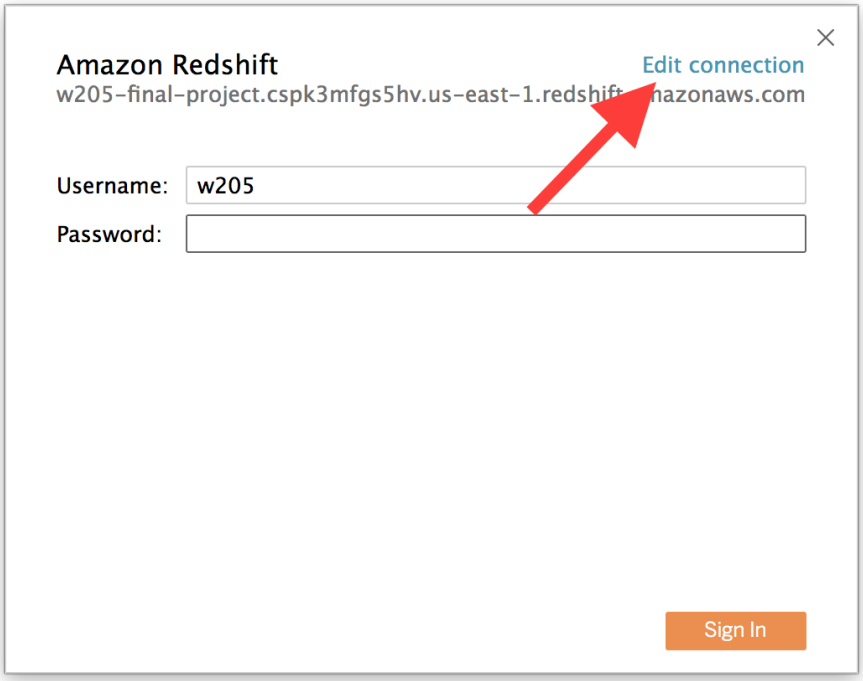
iv. **thishost = "w205-final-project.cspk3mfgs5hv.us-east-1.redshift.amazonaws.com"**
v. **thisport = "5439"**

e. Change the s3_load_buckets URLs from referencing the **w205-redshift-intermediate** bucket to the bucket you created in “Create a S3 bucket to hold your data”.

4. Run the entire notebook.
    a. This will load static data into S3, and copy that data into Redshift

Connect Tableau to Redshift

1. Download Tableau
    a. You may start a free 14-day trial here:
        i. https://www.tableau.com/products/desktop/download
    b. You may also sign up for a free license if you are a student here:
        i. https://www.tableau.com/academic/students
    c. Please download Tableau Desktop 10.4, older versions will not be compatible
2. Open “W205 Final Visualizations.twb” within the GitHub Repository
    a. Edit the connection so that it properly connects to your Amazon Redshift instance



        i.
    b. After the connection is established, the workbook should automatically connect to the “latest_position” view within Redshift, which executes a SQL query that joins the three tables we loaded into Redshift together.

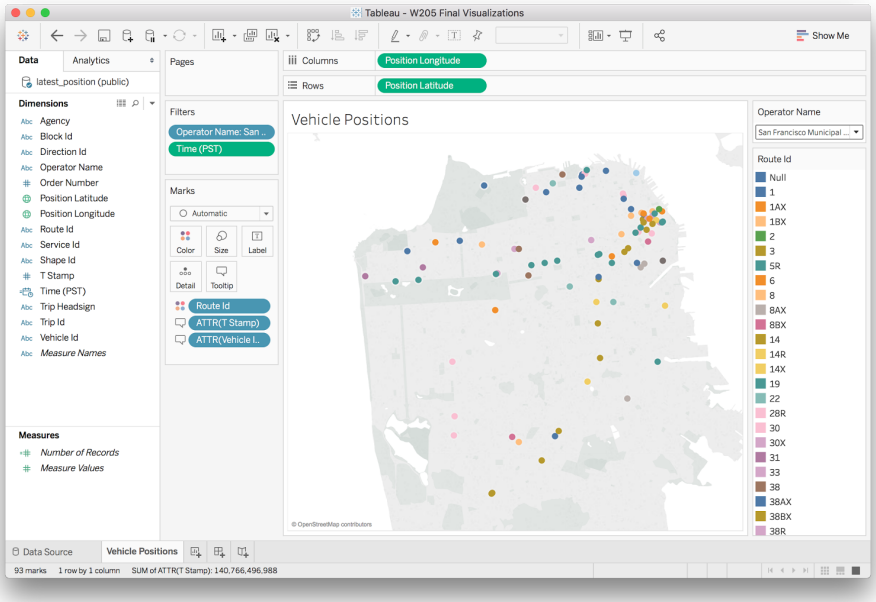


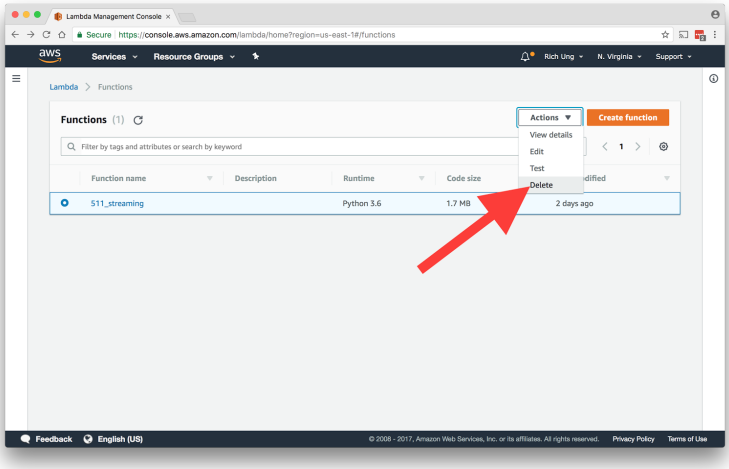
Tableau - W205 Final Visualizations
Show Me
Data
Analytics
latest_position (public)
Dimensions
Agency
Block Id
Direction Id
Operator Name
Order Number
Position Latitude
Position Longitude
Route Id
Service Id
Shape Id
T Stamp
Time (PST)
Trip Headsign
Trip Id
Vehicle Id
Measure Names
Measures
Number of Records
Measure Values
Pages
Filters
Operator Name: San ..
Time (PST)
Marks
Automatic
Color
Size
Label
Detail
Tooltip
Route Id
ATTR(T Stamp)
ATTR(Vehicle I..
Columns
Position Longitude
Rows
Position Latitude
Vehicle Positions
© OpenStreetMap contributors
Operator Name
San Francisco Municipal ...
Route Id
Null
1
1AX
1BX
2
3
5R
6
8
8AX
8BX
14
14R
14X
19
22
28R
30
30X
31
33
38
38AX
38BX
38R
Data Source
Vehicle Positions
93 marks
1 row by 1 column
SUM of ATTR(T Stamp): 140,766,496,988

i.

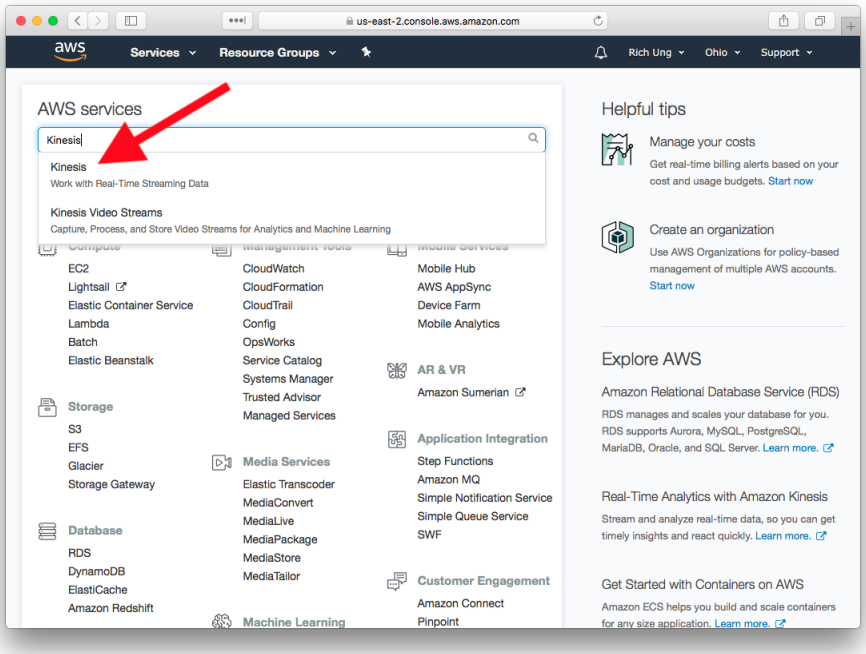
Terminating your AWS Environment

In order to avoid accruing extra charges from AWS, remove the following services from AWS once you are done with this project:

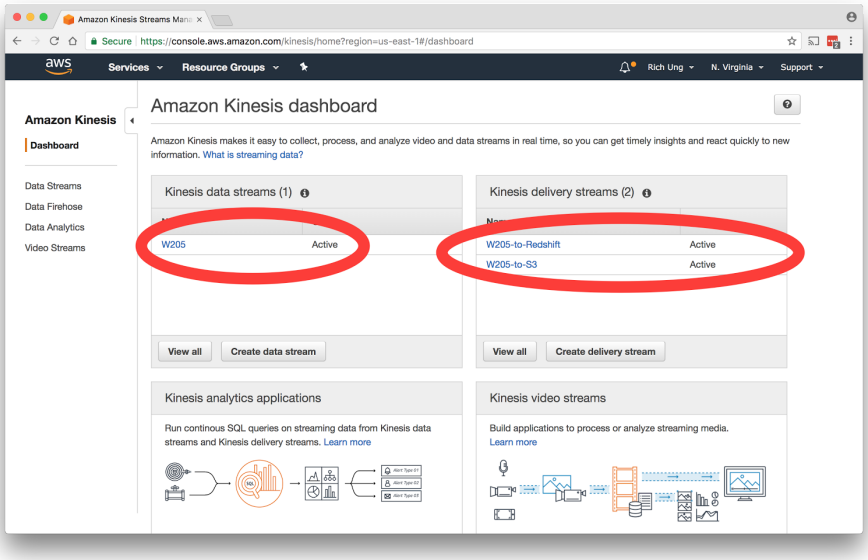
1. Lambda function
    a. Under the Lambda service, select your lambda function and select Actions > Delete



    i.
2. Kinesis
    a. Log into your AWS console and select the Kinesis service:

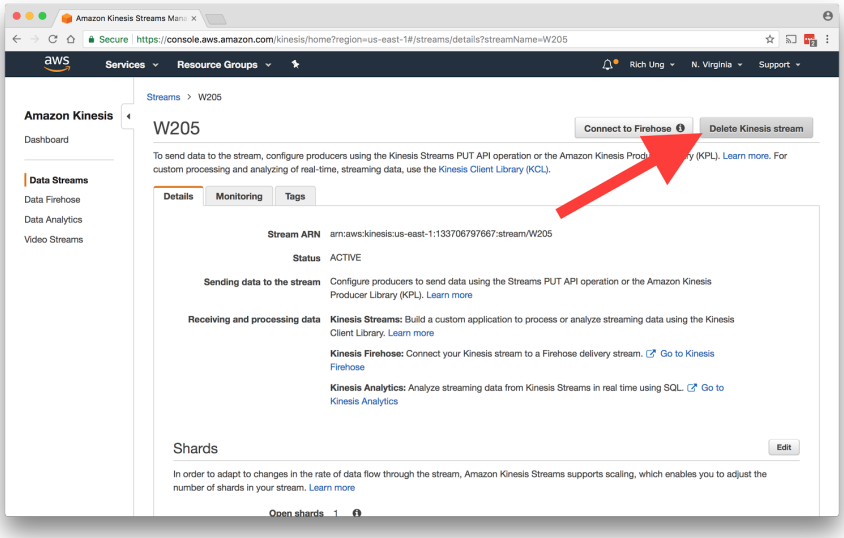


    i.
    b. Select your Kinesis data stream or Kinesis delivery stream

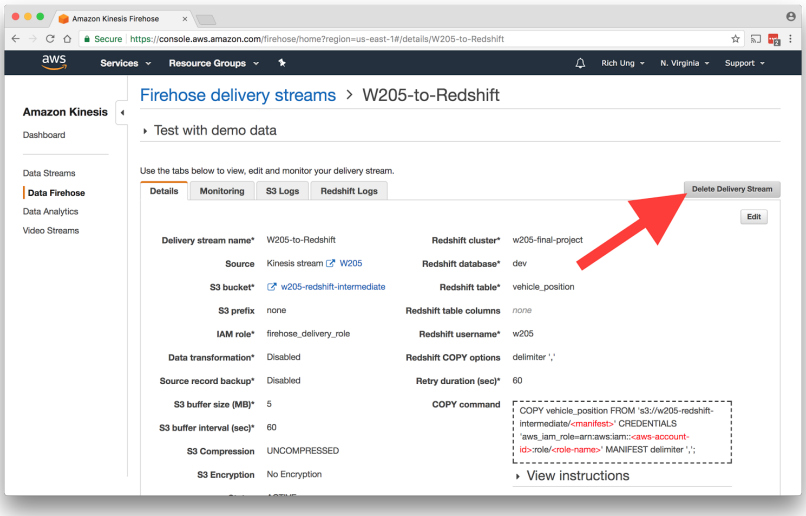


i.

c. Select either “Delete Kinesis stream” or “Delete Delivery Stream”



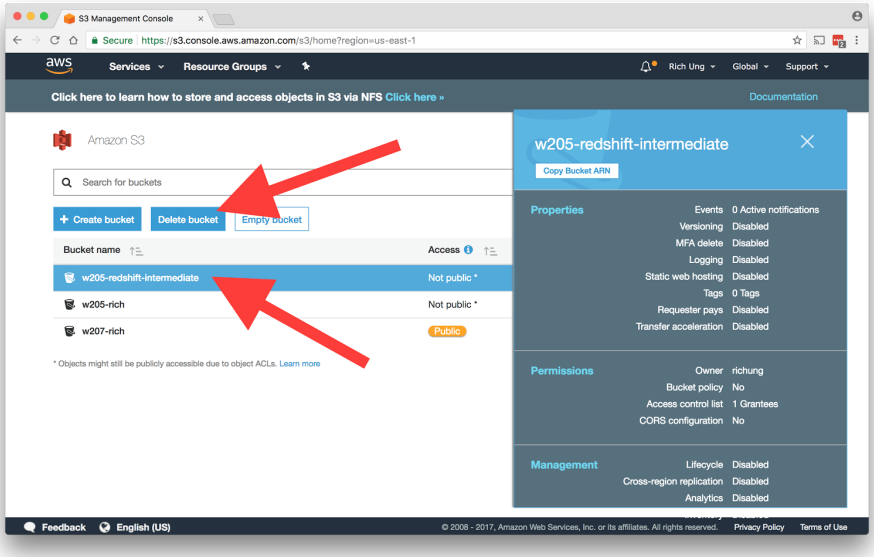
i.



ii.

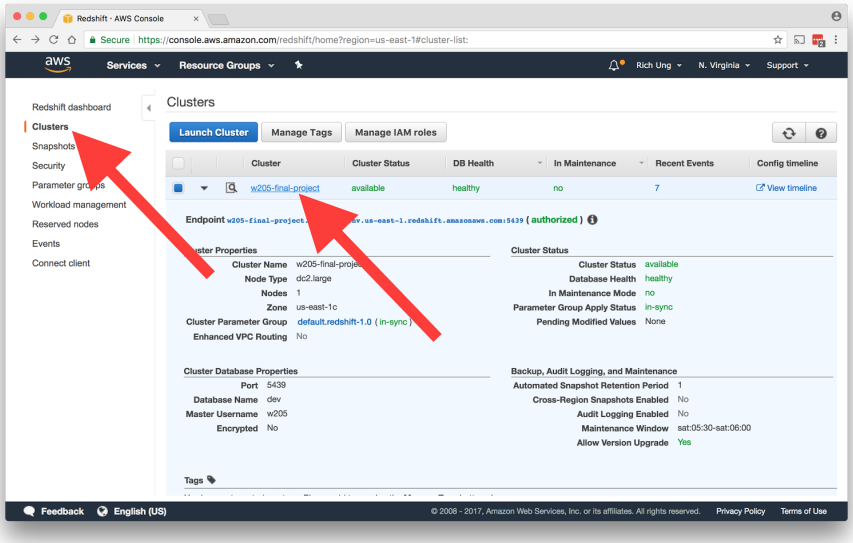
d. Repeat for all Kinesis data streams and Kinesis delivery streams

3. S3 bucket
    a. Under the S3 service, select your S3 bucket, and select “Delete Bucket”



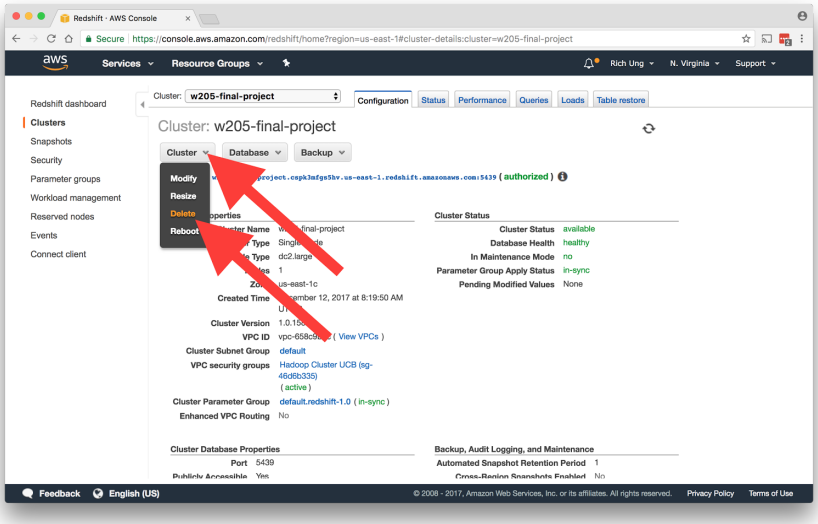
i.

4. Redshift
    a. Under the Redshift service, select the Clusters tab. Then click on your Redshift cluster.



i.

b. Click on the Cluster icon, then select “Delete”



i.