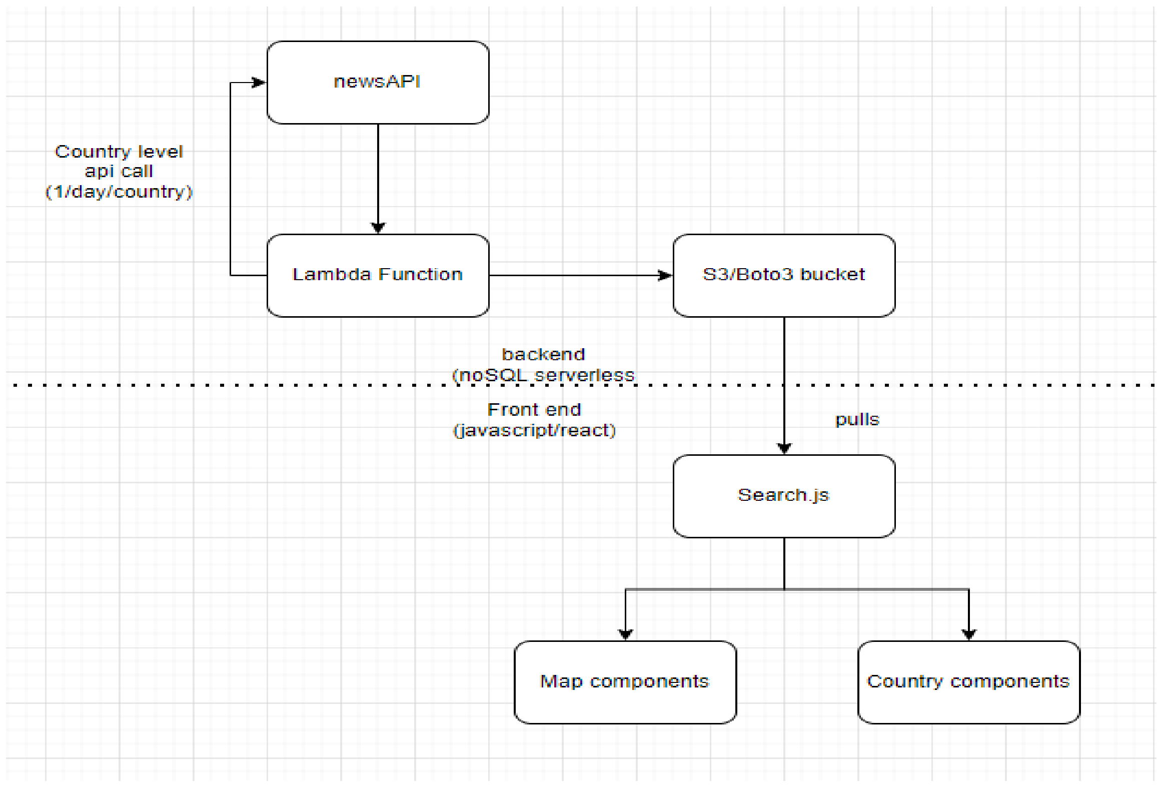
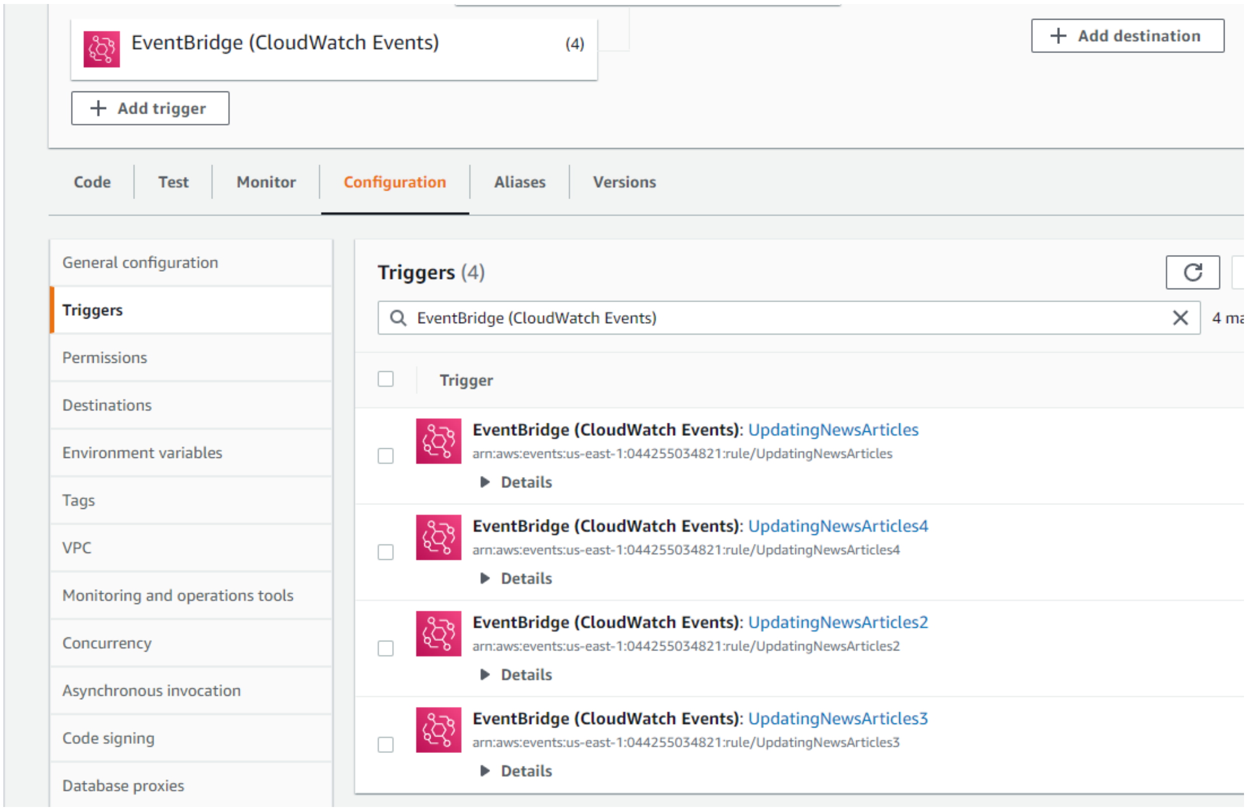

newsAPI
Country level api call (1/day/country)
Lambda Function
S3/Boto3 bucket
backend (noSQL serverless
Front end (javascript/react)
pulls
Search.js
Map components
Country components

The above picture will give a rough understanding of how headline data gets from newsAPI to the lambda function and further into the parts of the project.

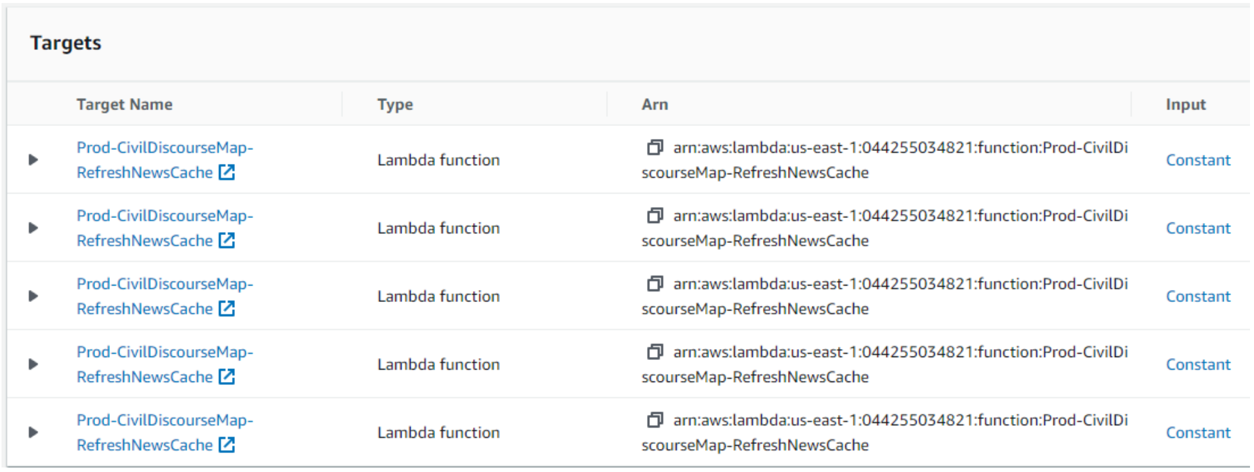
After this brief understanding of the flow of the project, Eventbridge needs to be understood.



Trigger
EventBridge (CloudWatch Events): UpdatingNewsArticles arn:aws:events:us-east-1:044255034821:rule/UpdatingNewsArticles
EventBridge (CloudWatch Events): UpdatingNewsArticles4 arn:aws:events:us-east-1:044255034821:rule/UpdatingNewsArticles4
EventBridge (CloudWatch Events): UpdatingNewsArticles2 arn:aws:events:us-east-1:044255034821:rule/UpdatingNewsArticles2
EventBridge (CloudWatch Events): UpdatingNewsArticles3 arn:aws:events:us-east-1:044255034821:rule/UpdatingNewsArticles3

Here are the event bridges registered within a given lambda function, each have their own rules and targets. In the case of “UpdatingNewsArticles<n>” they have similar rules but different targets.

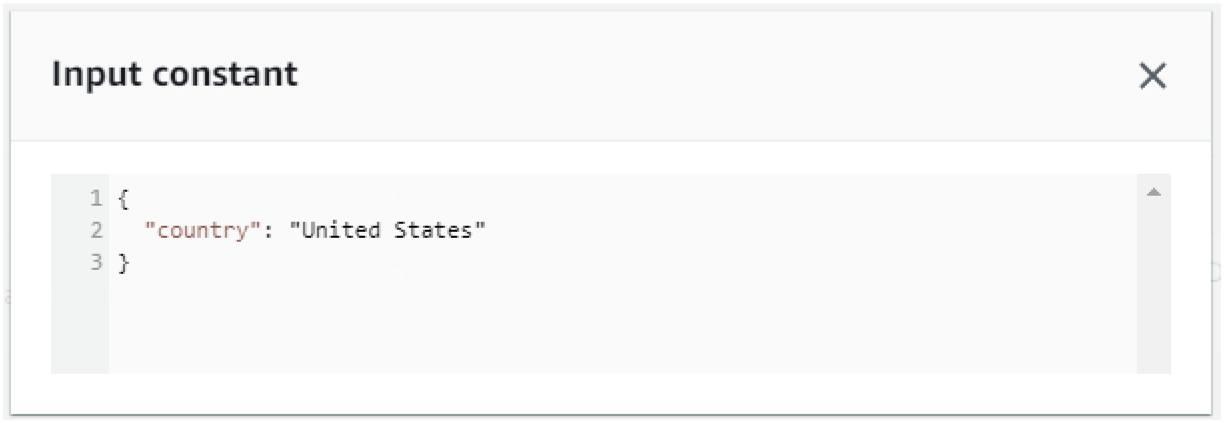
Using the first news rule as an example:



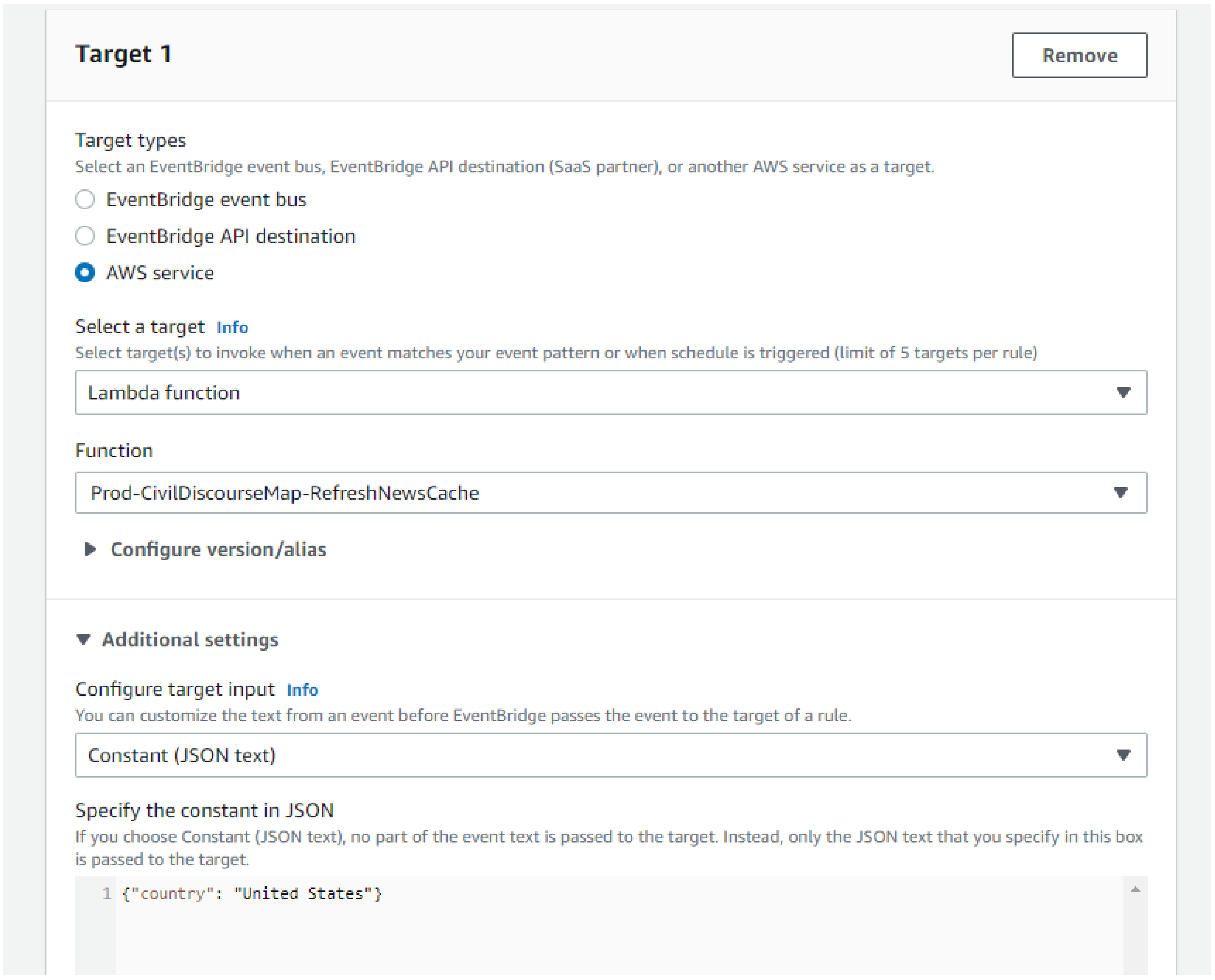
Targets

Target Name	Type	Arn	Input
Prod-CivilDiscourseMap-RefreshNewsCache	Lambda function	arn:aws:lambda:us-east-1:044255034821:function:Prod-CivilDiscourseMap-RefreshNewsCache	Constant
Prod-CivilDiscourseMap-RefreshNewsCache	Lambda function	arn:aws:lambda:us-east-1:044255034821:function:Prod-CivilDiscourseMap-RefreshNewsCache	Constant
Prod-CivilDiscourseMap-RefreshNewsCache	Lambda function	arn:aws:lambda:us-east-1:044255034821:function:Prod-CivilDiscourseMap-RefreshNewsCache	Constant
Prod-CivilDiscourseMap-RefreshNewsCache	Lambda function	arn:aws:lambda:us-east-1:044255034821:function:Prod-CivilDiscourseMap-RefreshNewsCache	Constant
Prod-CivilDiscourseMap-RefreshNewsCache	Lambda function	arn:aws:lambda:us-east-1:044255034821:function:Prod-CivilDiscourseMap-RefreshNewsCache	Constant

Here are the listed targets, but if we look more specifically at what the constant input is we see:


Input constant

```
{
  "country": "United States"
}
```


Target 1

Target types
Select an EventBridge event bus, EventBridge API destination (SaaS partner), or another AWS service as a target.

EventBridge event bus [✓ AWS service] EventBridge API destination

Select a target Info
Select target(s) to invoke when an event matches your event pattern or when schedule is triggered (limit of 5 targets per rule)

Lambda function

Function

Prod-CivilDiscourseMap-RefreshNewsCache

Configure target input Info
You can customize the text from an event before EventBridge passes the event to the target of a rule.

Constant (JSON text)

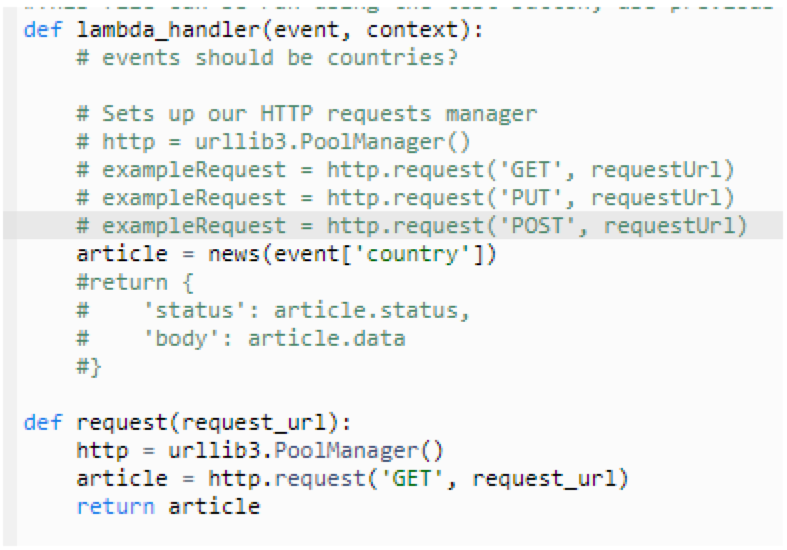
Specify the constant in JSON
If you choose Constant (JSON text), no part of the event text is passed to the target. Instead, only the JSON text that you specify in this box is passed to the target.

```
{"country": "United States"}
```

(the second picture above is for examples of editing a rule, what it should look like to accomplish the single country input)

that a single input constant is simply the string for the United states. This means that when this rule runs, it will pass the string “United States” to be parsed by the lambda function into an API call.

Below shows how the lambda function gets the information from the event bridges, and then begins to setup a URL request using a python library, urllib3

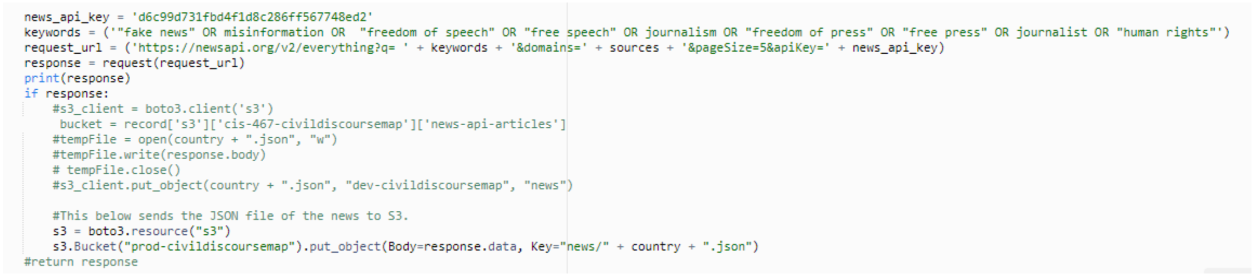


```python
def lambda_handler(event, context):
    # events should be countries?

    # Sets up our HTTP requests manager
    # http = urllib3.PoolManager()
    # exampleRequest = http.request('GET', requestUrl)
    # exampleRequest = http.request('PUT', requestUrl)
    # exampleRequest = http.request('POST', requestUrl)
    article = news(event['country'])
    #return {
    #    'status': article.status,
    #    'body': article.data
    #}

def request(request_url):
    http = urllib3.PoolManager()
    article = http.request('GET', request_url)
    return article
```

After this step is complete, we compare the string to country names for the request, and then build the request as shown below



```python
news_api_key = 'd6c99d731fbd4f1d8c286ff567748ed2'
keywords = ('"fake news" OR misinformation OR  "freedom of speech" OR "free speech" OR journalism OR "freedom of press" OR "free press" OR journalist OR "human rights"')
request_url = ('https://newsapi.org/v2/everything?q= ' + keywords + '&domains=' + sources + '&pageSize=5&apiKey=' + news_api_key)
response = request(request_url)
print(response)
if response:
    #s3_client = boto3.client('s3')
     bucket = record['s3']['cis-467-civildiscoursemap']['news-api-articles']
    #tempFile = open(country + ".json", "w")
    #tempFile.write(response.body)
    # tempFile.close()
    #s3_client.put_object(country + ".json", "dev-civildiscoursemap", "news")

    #This below sends the JSON file of the news to S3.
    s3 = boto3.resource("s3")
    s3.Bucket("prod-civildiscoursemap").put_object(Body=response.data, Key="news/" + country + ".json")
#return response
```

At the bottom of the code above you see S3, S3 is how we store the information within AWS, by making use of the bucket functionality. This all works in tandem to set up a daily, automated headline search request through the newsAPI call for each country, then storing it to be later accessed through an API call in Search.js

(refer to Search.js pdf for further information on data flow)