Book

Doc Writer m.thirumal@hotmail.com

1. Computer Science

1.1. AWS

1.1.1. Lambda Functions

Prerequisite: Setting up your environment

Create a virtualenv and install Chalice

Make sure you have Python 3 installed. See the env-setup page for instructions on how to install Python.

Create a new virtualenv called chalice-env by running the following command:

```
python3 -m venv chalice-env
```

Activate your newly created virtualenv:

```
source chalice-env/bin/activate
```

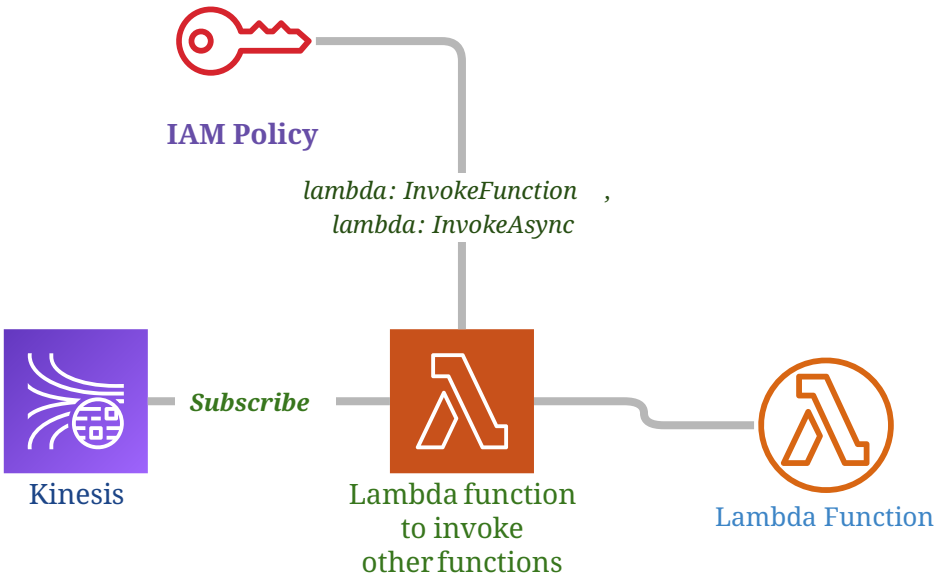
If you are using a Windows environment, you will have to run:

```
.\chalice-env\Scripts\activate
```

Install chalice using pip:

```
pip install chalice
```

Invoke AWS Lambda function from another lambda function



Prerequisite

- `IAM role` role to attach below policy to the `invoking` lambda function
  - `lambda:InvokeFunction`,
  - `lambda:InvokeAsync`

Steps

- Create invoker lambda function
  - Attach `IAM` policy
- Create lambda function for invoking
- Test it

Sample code with AWS Chalice fraework

Saple code to invoke another lambda function

```
from chalice import Chalice

import logging
import boto3

app = Chalice(app_name='lambda-invoker')

logger = logging.getLogger()
logger.setLevel(logging.DEBUG)

# Define the client to interact with AWS Lambda
client = boto3.client('lambda')

@app.on_kinesis_record(stream=os.environ.get('kinesisStreamName'))
def handler(event):
    logger.debug("Received event {}".format(event))
     #Invoke Lambda
    client.invoke(
        FunctionName = "arn:aws:lambda:ca-central-1:047076321700:function:eballot-
pollBetaStack-Handler-N3Z7PX8BK9PE",
        InvocationType = "Event",
        Payload = event
        )
    return
```

Lambda Layer

1.1.2. DynamoDb

DynamoDb to Kinesis Stream

- Create kinesis stream
- select the kinesis stream in Dynamodb table overview

1.2. Linux

1.3. Install/update Visual studio code in Ubuntu

```
sudo dpkg -r code
sudo dpkg -i code_1.56.2-1620838498_amd64.deb
```

1.4. Pagination

1.4.1. Pagination

Pagination is a sequence of pages which are connected and have similar content

Page is a fixed count of results or fixed size of results

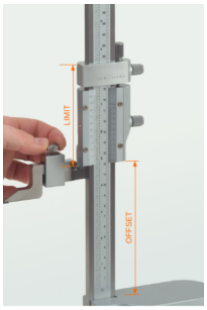
Client application may request to skip multiple page and fetch the following page

Keywords

- LIMIT,
  - `"LIMIT"` as it's name suggests, limits the number of rows returned in a SELECT query
- OFFSET,
- ORDER BY
- Pagination

Thinks that are need to be handled

- Skiping page
- Fetch based on the page & limit



Types of Pagination

- Offset-based Pagination
- Token-based Pagination

Offset Based Pagination

Traverse Forward

Traverse backward

Token Based Pagination

Traverse Forward

`nextToken` is used to specify the record after which additional items should be fetched, along with the page size.

The implementation of the token is system-specific.

`DynamoDB` is an example of at system that uses `token-pagination`

Traverse backward

Pagination in Graph

Vertices can be refered directly but when we execute a graph traversal we are effectively conducting a complex join among the vertices that take part in the traversal.

```
In the general case, to support pagination, we need to store some state for each of
the participating node and edge, as each of them may contribute to the final output
during the computation of the next page.
```

1.5. AsciiDoc

```
include::asciidoc.adoc[]
```

2. Biology

2.1. Botonay

2.1.1. Inosculation

When branches or roots of different trees are in prolonged intimate contact, they often abrade each other, exposing their inner tissues, which may eventually fuse.

It’s not so much one tree feeding another as the formation of a new hybrid organism.