

AWS Lambda

Objectives

Key objectives of this chapter

- Lambda Functions
 - ◇ Use cases
 - ◇ The programming model
 - ◇ Lambda blueprints

1.1 What is AWS Lambda?

- AWS Lambda lets you run your code written in a number of supported languages in a PaaS-like environment with zero administration on your part. There is no EC2 instance needed to run your code.
 - ◇ This arrangement is referred to as Serverless Computing or NoOps
- AWS provisions, scales, and manages servers needed to run your code, referred to as the *Lambda function*
 - ◇ You pay only for the actual time your Lambda function runs
 - ✓ You are charged in 0.1 second increments multiplied by the number of Lambda function invocations
 - ◇ You are only required to provide the amount of computing memory (the default is 128MB)
 - ✓ CPU is allocated in proportion to the requested memory
- You write code as Lambda functions. They can:
 - ◇ Respond to changes (events) in other AWS services (sources of events), such as an S3 update or HTTP requests.

Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com

- ◇ Interact with external resources like Amazon DynamoDB or other web services.

Notes:

Requesting 256MB of memory for your Lambda function allocates approximately twice as much computing power as in case of the default 128MB of memory.

1.2 Supported Languages

- As of 2017, Lambda gives you a choice of the following languages:
 - ◇ Node.js (JavaScript)
 - ◇ Python 2.7
 - ◇ Java 8
 - ✓ Lambda provides the Amazon Linux build of openjdk 1.8
 - ◇ C#
 - ✓ Distributed as a NuGet package with the “dotnetcore1.0” runtime parameter
- Lambda functions can use dependent libraries, including native ones
 - ◇ There is no white-list APIs
- Your code can spawn additional threads, if needed
- **Note:** Disabled activities are mentioned in the Slide Notes

Notes:

According to Lambda documentation:

"There are a few activities that are disabled: Inbound network connections are blocked by AWS Lambda, and for outbound connections only TCP/IP sockets are supported, and ptrace (debugging) system calls are restricted. TCP port 25 traffic is also restricted as an anti-spam measure."

Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com

1.3 Getting Your Code Up And Running in Lambda

- You have three options:
 1. Create your Lambda function code inside the AWS Management Console (the in-line option)
 - ✓ Only available for Node.js and Python runtimes
 2. Develop code locally, build a ZIP or JAR bundle with all the dependencies and upload it to AWS
 3. Upload your ZIP or JAR to S3

Notes:

According to Lambda documentation:

"

Uploads must be no larger than 50MB (compressed). You can use the AWS Eclipse plugin to author and deploy Lambda functions in Java. You can use the Visual Studio plugin to author and deploy Lambda functions in C#, and Node.js.

...

You can define Environment Variables as key-value pairs that are accessible from your function code. These are useful to store configuration settings without the need to change function code. Learn more. For storing sensitive information, we recommend encrypting values using KMS and the console's encryption helpers.

"

Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com

1.4 Examples of the Base Lambda Function

Target Runtime	Lambda Function Code
Node.js (ECMAScript 2015)	<pre>exports.handler = (event, context, callback) => { callback(null, 'Hello from Lambda'); };</pre>
Python	<pre>def lambda_handler(event, context): return 'Hello from Lambda'</pre>

1.5 Use Cases

- High throughput real-time workflows handling billions of events per day
- Back-ends for you apps
- Media objects transcoding
- Real-time tracking of calls made to any Amazon Web Service from your app
- Front-end HTTP service
 - ◇ You can invoke your Lambda function via an HTTP endpoint using Amazon API Gateway

Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com

1.6 How It Works

- AWS Lambda functions run inside a default AWS-managed VPC on the computing infrastructure provided by AWS
 - ◇ Optionally, you can configure Lambda to run within your custom VPC
- It is highly recommended you write your code in a “stateless” style
 - ◇ Any state you may want to retain beyond the lifetime of the request should be externalized to an AWS persistent store, e.g. Amazon S3, Amazon DynamoDB, etc.
- You can configure triggers for the Lambda, such as uploading a file on S3, that will cause the Lambda to be executed.
- You can also invoke lambda functions directly using the AWS SDKs or AWS Mobile SDKs, such as the AWS Mobile SDK for Android
- Lambda provides a great foundation for building microservices

Notes:

According to Lambda documentation:

"

To improve performance, AWS Lambda may choose to retain an instance of your function and reuse it to serve a subsequent request, rather than creating a new copy. Your code should not assume that this will always happen.

"

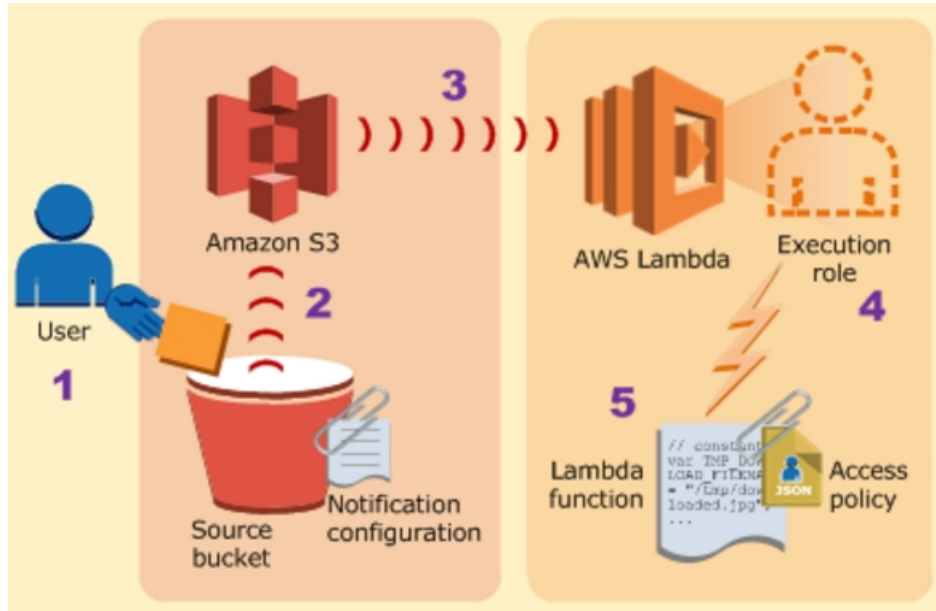
Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com

1.7 Example: Processing S3 Source Events with Lambda



- Source: AWS Documentation

1.8 The Programming Model

- The following core concepts apply to Lambda functions created in any of the supported languages:
 - ◇ **Handler** - The user call-back function invoked by AWS Lambda in response to a registered event; this function is passed the event object and the context object
 - ◇ **The context object** - The AWS Lambda object that provides information of the call context
 - ◇ **Logging** - Any user Lambda function can contain language-specific logging statements output of which AWS redirects to CloudWatch Logs

Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com

Notes:

According to Lambda documentation:

"

AWS Lambda provides this information via the context object:

- *How much time is remaining before AWS Lambda terminates your Lambda function (timeout is one of the Lambda function configuration properties).*
- *The CloudWatch log group and log stream associated with the Lambda function that is executing.*
- *The AWS request ID returned to the client that invoked the Lambda function. You can use the request ID for any follow up inquiry with AWS support.*
- *If the Lambda function is invoked through AWS Mobile SDK, you can learn more about the mobile application calling the Lambda function.*

"

1.9 Configuring Lambda Functions

- The Lambda Dashboard of the AWS Management Console offers you a wizard-like Lambda function creation flow, where you need:
 - ◇ Specify the source of events to which your lambda function will be triggered to respond
 - ◇ Provide the function body written in the language of your choice
 - ✓ Write code in-line or upload the ZIP file containing your code
 - ◇ Specify the role under which your code will be running
 - ◇ Optionally, provide memory size (default is 128 MB), the call timeout (a value between the default 3 seconds and the maximum of 5 minutes), and configuration key-value pairs

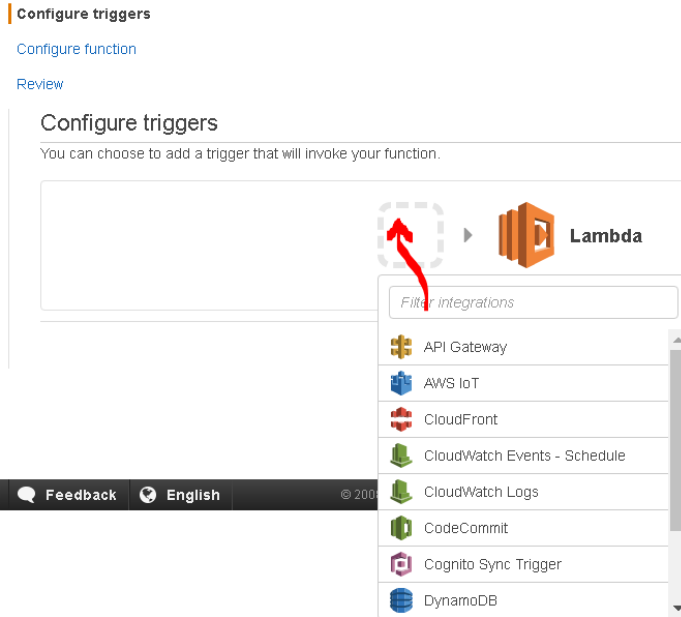
Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com

1.10 Configure Triggers Page



Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com

1.11 Lambda Function Blueprints

- You have an option to either create a Lambda function from scratch, or use Lambda blueprints
- A Lambda blueprint is a sample configuration of an event source and a skeletal Lambda function for handling such events
- As of 2017, Lambda offers 78 blueprints, such as
 - ◇ ***dynamodb-process-stream***
 - ✓ *"An Amazon DynamoDB trigger that logs the updates made to a table"*
 - ◇ ***lex-book-trip-python***
 - ✓ *"Book details of a visit, using Amazon Lex to perform natural language understanding"*
 - ◇ ***microservice-http-endpoint***
 - ✓ *"A simple backend (read/write to DynamoDB) with a RESTful API endpoint using Amazon API Gateway"*

Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com

1.12 How Do I Troubleshoot and Monitor My Lambda Functions?

- AWS Lambda automatically monitors your Lambda functions reporting the following real-time metrics through Amazon CloudWatch:
 - ◇ Invocation count
 - ◇ Invocation duration
 - ◇ Invocation errors
- You can use these metrics to set CloudWatch custom alarms
- The CloudWatch Logs group associated with your Lambda function can be accessed by this logical path: */aws/lambda/<your function name>*

Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com

1.13 Developing Lambda in Java

- You can create a Lambda in a plain Maven project or use the AWS Toolkit plugin for Eclipse. If you are using Maven, add dependency for `com.amazonaws:aws-lambda-java-core`.
- Develop POJO classes that will carry request and response data.
- Develop the Lambda class which must implement

```
public class MyLambda
    implements RequestHandler<MyRequest, MyResponse> {

    public MyResponse handleRequest(MyRequest request,
Context context) {
    //...
    }

}
```

- Build a JAR file that includes all the dependency classes and upload it in for your Lambda.

1.14 Summary



- In this chapter, we reviewed the AWS Lambda service that provides an excellent platform for building microservices in the Amazon Cloud.

Canada

821A Bloor Street West, Toronto, Ontario, M6G 1M1
1 866 206 4644 getinfo@webagesolutions.com

United States

744 Yorkway Place, Jenkintown, PA. 19046
1 877 517 6540 getinfousa@webagesolutions.com