

 λ

Living on the Edge with Lambda@Edge



[linkedin.com/in/sarah-victoria-bernard](https://www.linkedin.com/in/sarah-victoria-bernard)



 realestate.com.au

 realestate.com.au
Home Loans

 realcommercial.com.au

 [Flatmates
.com.au](http://Flatmates.com.au)

 spacely

 smartline
personal mortgage advisers

 hometrack



 PROPTIGER.COM

 HOUSING.com

 makaan

 [买房
myfun.com](http://买房 myfun.com)

 iProperty.com.my

 iProperty.com.sg

 squarefoot.com.hk

 SMARTExpo
Investment & International Property Expo

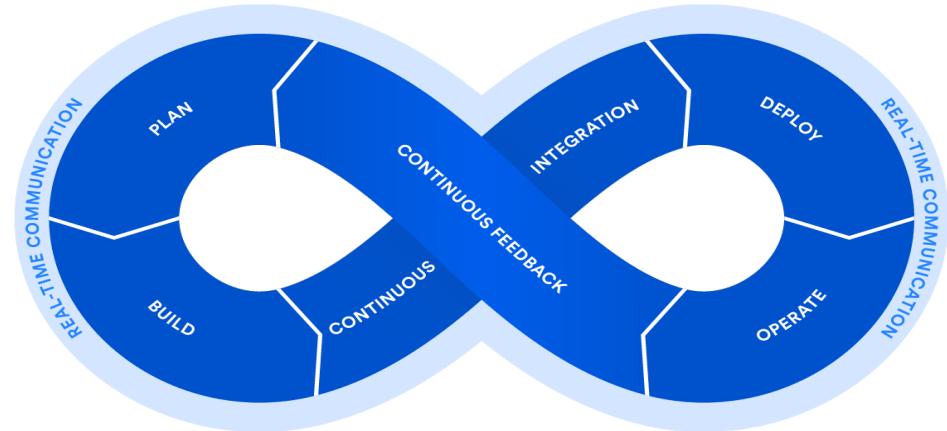
 rumah123.com
Situs Properti No.1 di Indonesia

 thinkofliving.com

 Prakard.com



 realtor.com
move



What is Lambda@Edge?

- What is it
- How is works

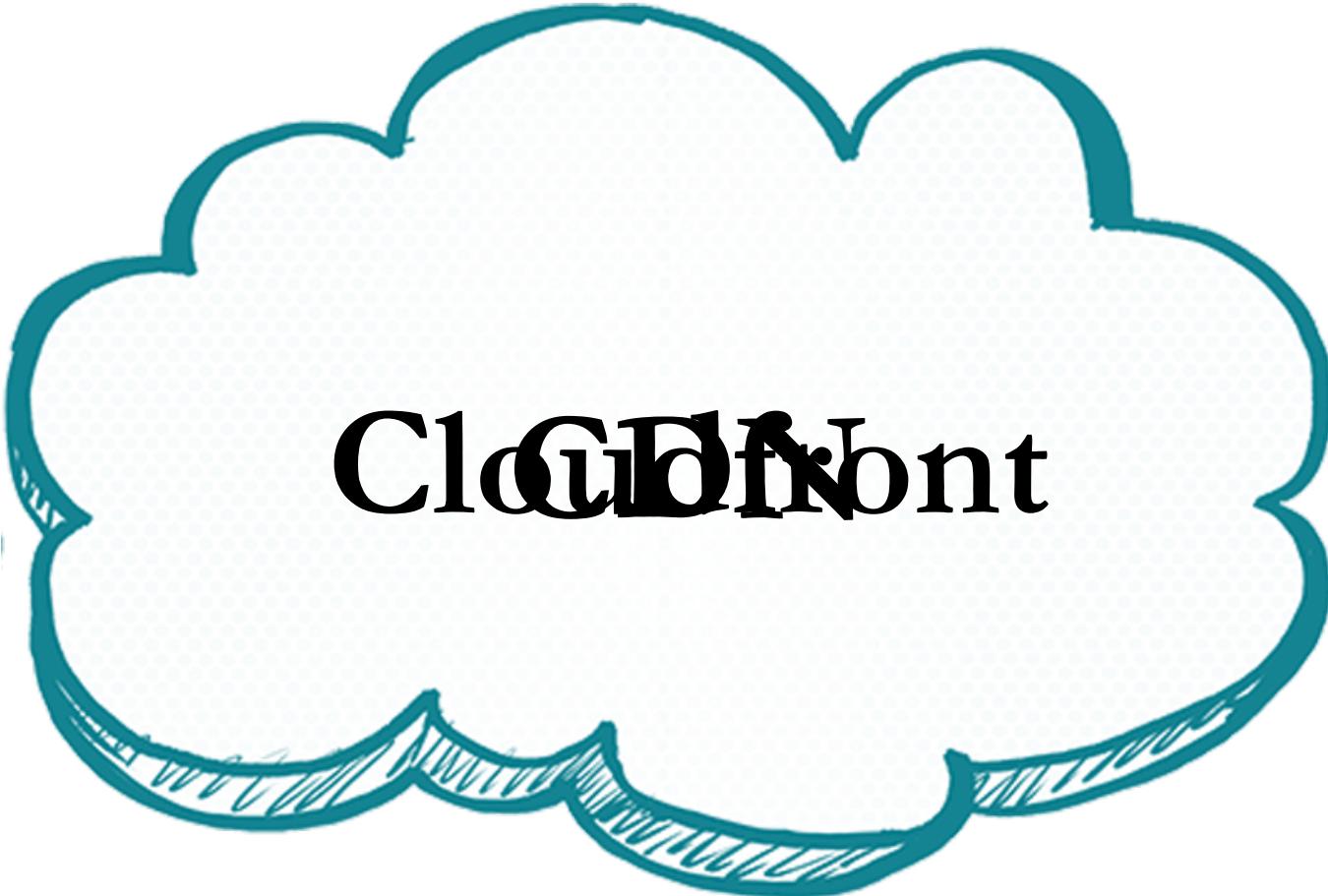
How to use it

- Hello World Example
- Beginner Pain Points

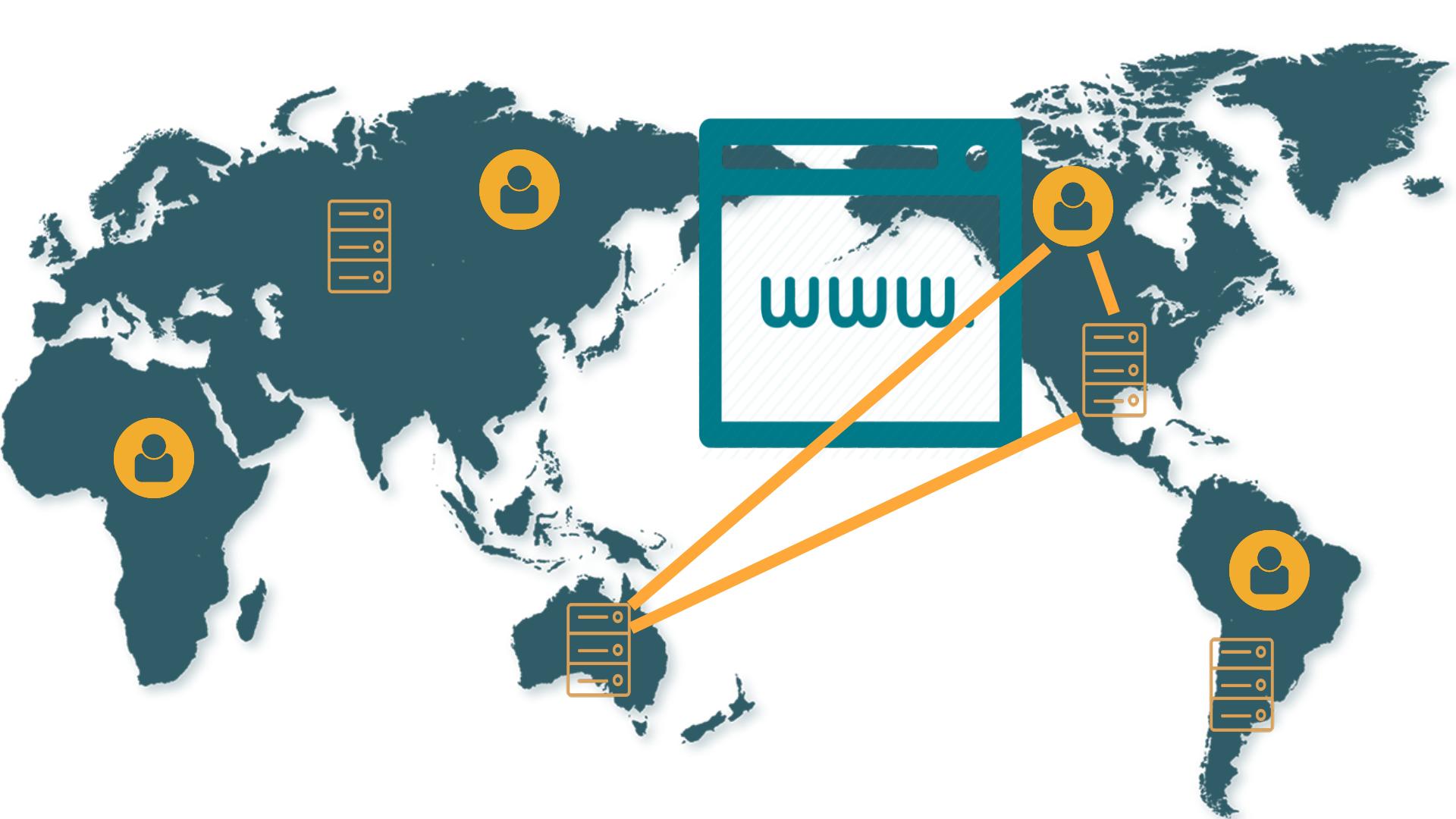
Uses

- How it helped me
- What problems could it solve for you?

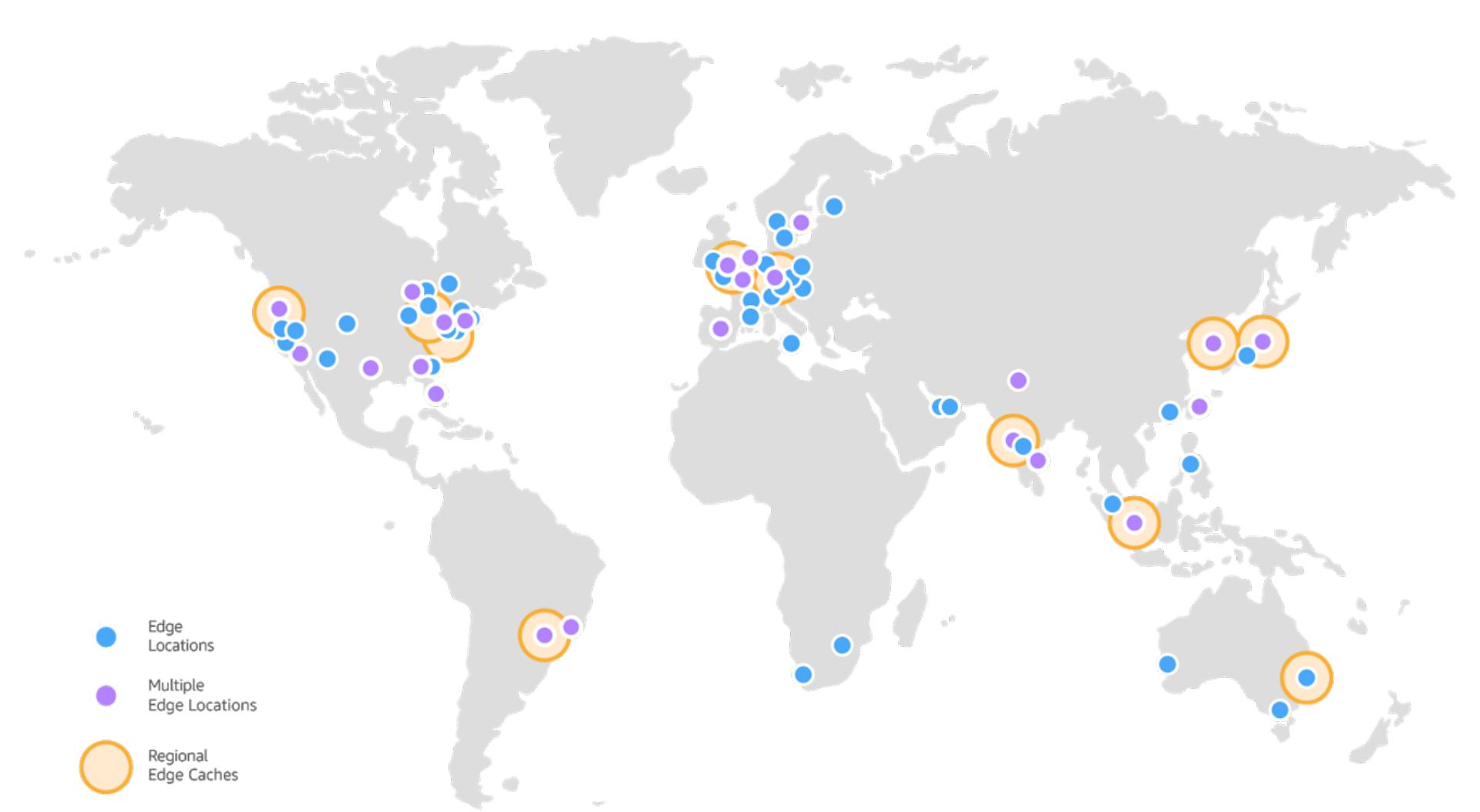
What is
Lambda@Edge?



Cloudfront

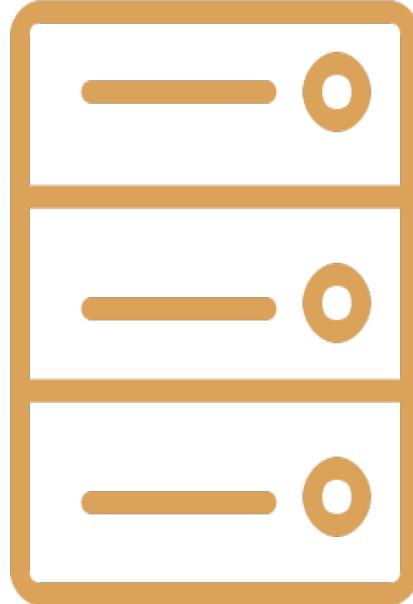


www.



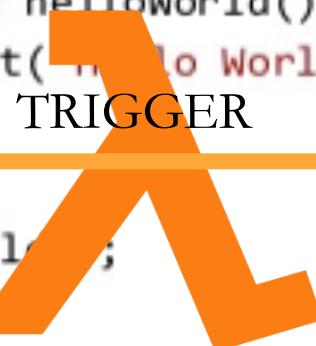
AWS Lambda

- Provisioning or managing servers
 - Managing scaling up and down
 - Security patches
- ```
Our main function Hello World!
function helloWorld() {
 alert('Hello World!');
}
helloWorld();
```





```
// our main function Hello World
function helloWorld() {
 alert('Hello World!');
}
helloWorld();
</script>
```





TRIGGER



TRIGGER



TRIGGER

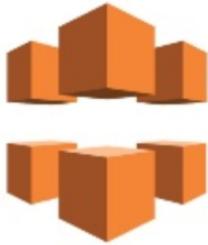




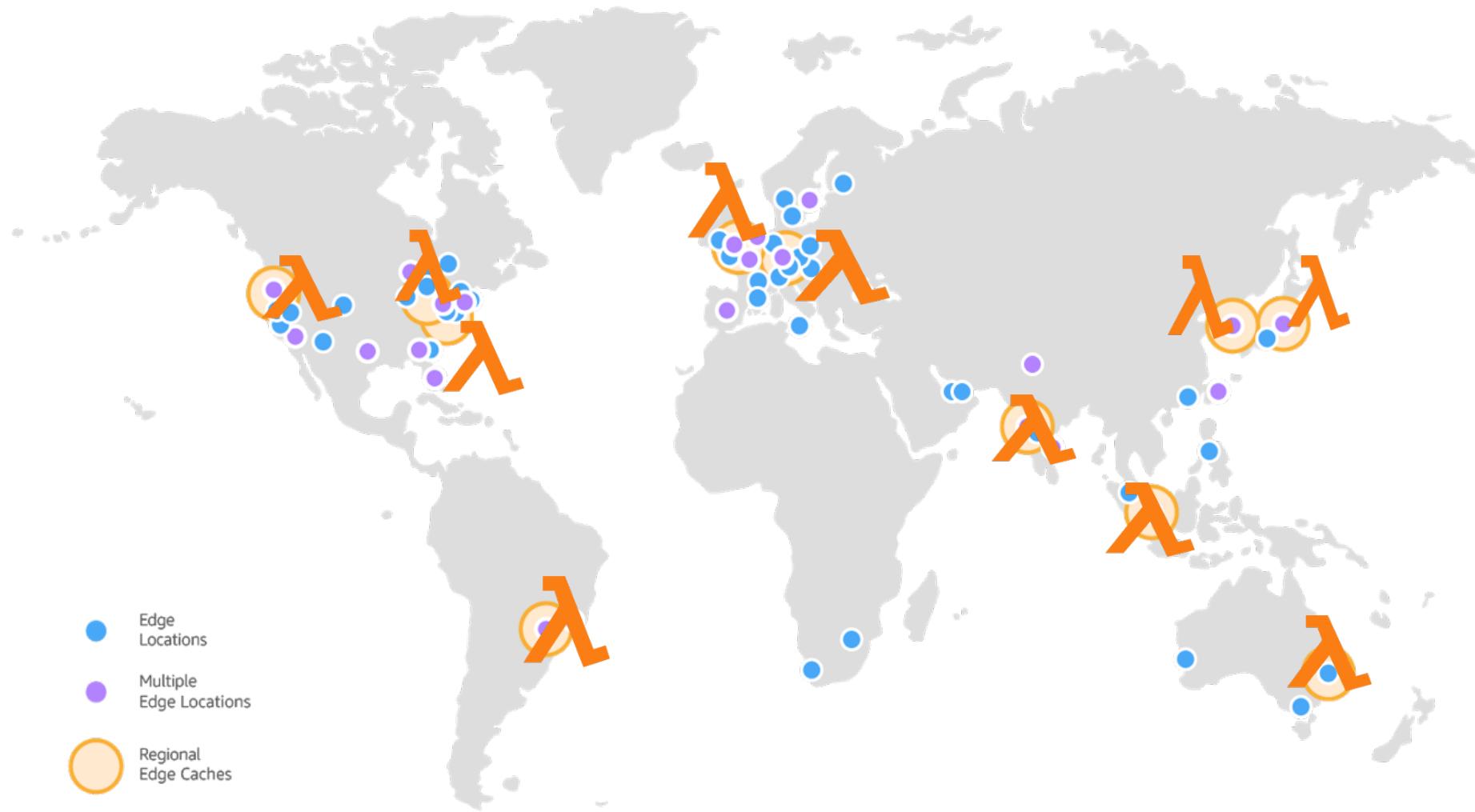
AWS Lambda



Amazon CloudFront



Lambda@Edge



- Intelligent processing of HTTP requests
- No need to worry about scaling up and down
- Handle thousands of requests per second (10,000 (in each region))
- Don't pay for down time

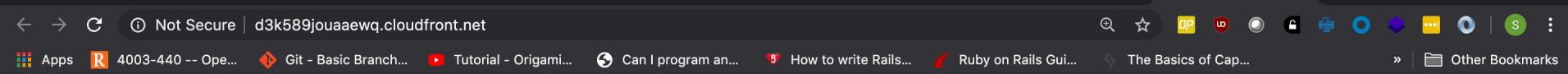
Why?



# What can Lambda@Edge be used for?

- A/B testing
- Image resizing
- App Personalisation
- HTTP Redirects and URL rewrites
- Bot Detection@edge
- Authorisation and Authentication @edge

# How To Use It



# Hello World from the Edge!

# 1. Create a Lambda Function



```
↳ hello-world
↳ hello-world
↳ hello-world
↳ hello-world |
```



## **HANDLER**

- AWS Lambda invokes your Lambda function via a handler object.

## **CONTEXT**

- When Lambda runs your function, it passes a context object to the handler

## **CALLBACK**

- Used to return information back to the caller.

## **EVENT**

- A JSON object you can manipulate from the HTTP request





# Cloudfront Request Event Structure

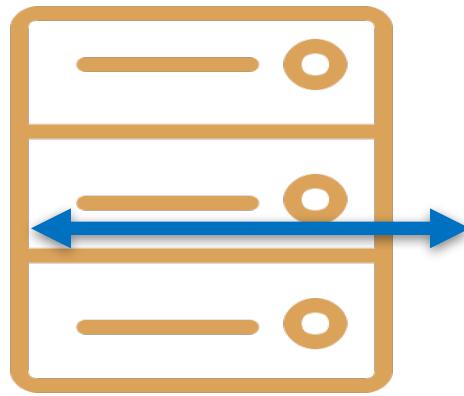
- Headers
- Querystring
- uri (read/write)
- data (read/write)
- customHeaders
- readTimeout
- sslProtocols

```
{
 "Records": [
 {
 "cf": {
 "config": {
 "distributionDomainName": "d123.cloudfront.net",
 "distributionId": "EDFDVNBGZEXAMPLE",
 "eventType": "viewer-request",
 "requestId": "MRVMF7KydivxMwfJiglgwHQwZsbG2IhRJ07sn9AkKUFSHS9EXAMPLE=="
 },
 "request": {
 "body": {
 "action": "read-only",
 "data": "eyJlc2VybmttZSI6IkxhbWJKYUBFZGdIiwiY29tZWVuDCI6IlRoaxXMcgvXdxWzdCBib2R5In0==",
 "encoding": "utf-8",
 "inputTruncated": false
 },
 "clientIp": "2001:0db8:85a3:0:0:8a2e:0370:7334",
 "queryString": "size=large",
 "uri": "/picture.jpg",
 "method": "GET",
 "headers": {
 "host": [
 {
 "key": "Host",
 "value": "d11111abcdef8.cloudfront.net"
 }
],
 "user-agent": [
 {
 "key": "User-Agent",
 "value": "curl/7.51.0"
 }
]
 },
 "origin": {
 "custom": {
 "customHeaders": {
 "my-origin-custom-header": [
 {
 "key": "My-Origin-Custom-Header",
 "value": "Test"
 }
]
 }
 },
 "domainName": "example.com",
 "keepaliveTimeout": 5,
 "path": "/custom_path",
 "port": 443,
 "protocol": "https",
 "readTimeout": 5,
 "sslProtocols": [
 "TLSv1.1"
],
 "s3": {
 "authMethod": "origin-access-identity",
 "customHeaders": {
 "my-origin-custom-header": [
 {
 "key": "My-Origin-Custom-Header",
 "value": "Test"
 }
]
 },
 "domainName": "my-bucket.s3.amazonaws.com",
 "path": "/s3_path",
 "region": "us-east-1"
 }
 }
 }
 }
 }
]
}
```

# Cloudfront Response Event Structure

```
{
 "Records": [
 {
 "cf": {
 "config": {
 "distributionDomainName": "d123.cloudfront.net",
 "distributionId": "EDFVBDEEXAMPLE",
 "eventType": "viewer-request",
 "requestId": "MRVMF7KydivxMWFJiglgwHQwZsbG2IhRJ07sn9AkKUFSHS9EXAMPLE=="
 },
 "request": {
 "body": {
 "action": "read-only",
 "data": "eyJlc2VybmcPtZSI6IkxhbWJKYUBFZGd1IiwiY29tbWVudCI6IlRoaxMgaXMgcmVxdWVzdCBib2R5In0=",
 "encoding": "base64",
 "inputTruncated": false
 },
 "clientIp": "2001:0db8:85a3:0:0:8a2e:0370:7334",
 "contentType": "image/jpeg",
 "url": "/picture.jpg",
 "method": "GET",
 "headers": {
 "host": [
 {
 "key": "Host",
 "value": "d11111abcdef8.cloudfront.net"
 }
],
 "user-agent": [
 {
 "key": "User-Agent",
 "value": "curl/7.51.0"
 }
]
 },
 "origin": {
 "custom": {
 "customHeaders": {
 "my-origin-custom-header": [
 {
 "key": "My-Origin-Custom-Header",
 "value": "Test"
 }
],
 "domainName": "example.com",
 "keepaliveTimeout": 5,
 "path": "/custom_path",
 "port": 443,
 "proto": "https",
 "readTimeout": 5,
 "sslProtocols": [
 "TLSv1",
 "TLSv1.1"
],
 "s3": {
 "authMethod": "origin-access-identity",
 "customHeaders": {
 "my-origin-custom-header": [
 {
 "key": "My-Origin-Custom-Header",
 "value": "Test"
 }
],
 "domainName": "my-bucket.s3.amazonaws.com",
 "path": "/s3_path",
 "region": "us-east-1"
 }
 }
 }
 }
 }
 }
 }
 }
]
}
```

# 2. Make A Cloudfront Distribution



Origin

# Cloudformation

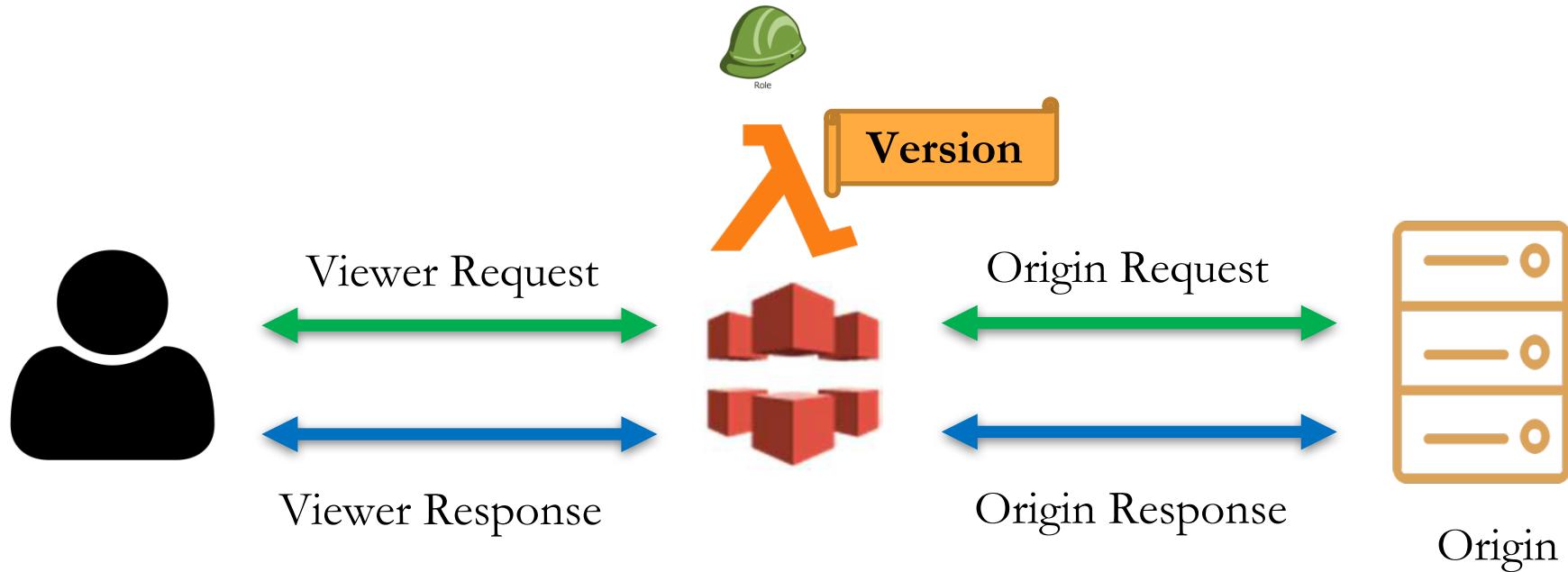
A common language for you  
to describe and provision all  
the infrastructure resources in  
your cloud environment



# 3. Link Them Together

# How do I link my Lambda Function to Cloudfront?

1. Pick an Event Type
2. A Version of your Lambda Function
3. IAM Role for your Lambda Function



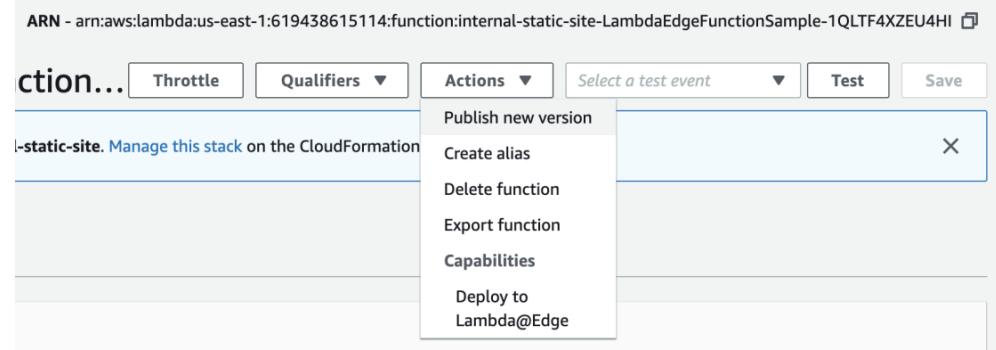
# Lambda Version

**VersionedLambdaFunction:**

Type: 'AWS::Lambda::Version'

Properties:

FunctionName: !Ref LambdaFunction



A new version will **not** be created  
when the Lambda function code  
changes!!!

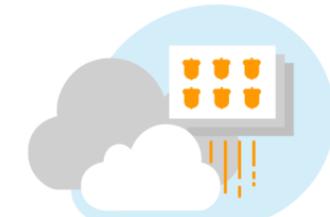
# AWS Serverless Application Model (SAM)



MEET SAM.

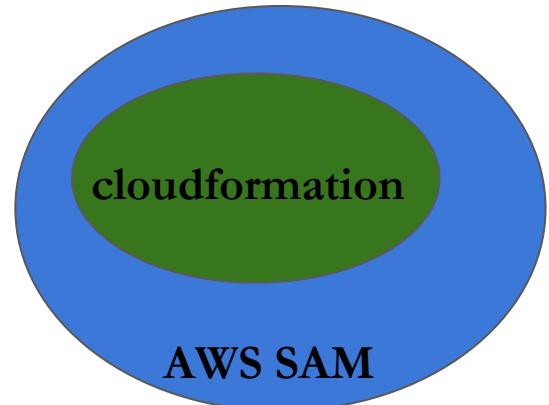


USE SAM TO BUILD TEMPLATES THAT DEFINE  
YOUR SERVERLESS APPLICATIONS.



DEPLOY YOUR SAM TEMPLATE  
WITH AWS CLOUDFORMATION.

- Extends AWS CloudFormation
- Simplify packaging and deploying
- Local debugging and testing with AWS SAM CLI



Hey Cloudfront!

The Lambda Code  
has changed.

I have a new Lambda  
version for you!



# Install SAM CLI

## Mac or Linux

Install SAM CLI using Brew

```
brew tap aws/tap
brew install aws-sam-cli
```

Requires Docker and the AWS CLI. [See installation instructions](#)

## Windows

Install SAM CLI using an MSI

- [64 bit](#)
- [32 bit](#)

Requires Docker and the AWS CLI. [See installation instructions](#)

## Install using pip

```
pip install --user aws-sam-cli
```

Requires Docker and the AWS CLI. [See installation instructions](#)

# AWS Serverless Application Model (SAM)

```
CFDistribution:
 Type: AWS::CloudFront::Distribution
 Properties:
 DistributionConfig:
 WebACLId: !Ref InternalWebACL
 Aliases:
 - !Ref 'CloudFrontDomainName'
 Enabled: 'true'
 Comment: 'CI/CD for Lambda@Edge'
 Origins:
 - Id: S3Origin
 DomainName: !GetAtt S3Bucket.DomainName
 S3OriginConfig:
 OriginAccessIdentity: !Join ["", ["origin-access-identity/cloudfront/",
 !Ref CloudFrontOriginAccessIdentity]]
 DefaultCacheBehavior:
 TargetOriginId: S3Origin
 LambdaFunctionAssociations:
 -
 EventType: origin-request
 LambdaFunctionARN: !Ref LambdaEdgeFunctionSample.Version
 ForwardedValues:
 QueryString: 'false'
 Headers:
 - Origin
 Cookies:
 Forward: none
 ViewerProtocolPolicy: allow-all
```



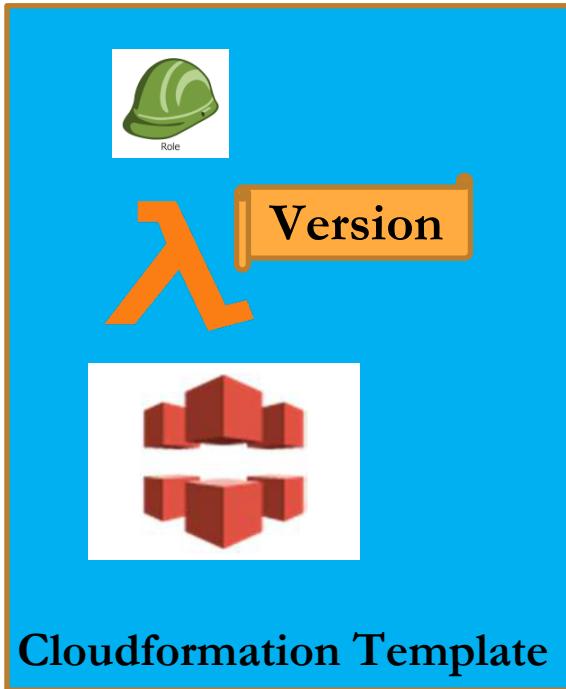
```
LambdaEdgeFunctionSample:
 Type: AWS::Serverless::Function
 Properties:
 CodeUri: ../src/
 Role: !GetAtt LambdaEdgeFunctionRole.Arn
 Runtime: nodejs6.10
 Handler: index.handler
 Timeout: 5
 AutoPublishAlias: live
```





# 4. Deploy It

# What Does AWS expect?



Lambda  
Code

Packaged Code



S3



# 1. Package It Up

```
aws cloudformation package --template-file /path_to_template/template.json --s3-bucket bucket-name --output-template-file packaged-template.json
```



```
→ static-webinator git:(hello-world-voxxed) ✘ aws \
--profile bear cloudformation package \
--template-file template.yml \
--output-template-file packaged.yaml \
--s3-bucket webinator-lambda-bucket-1234 \
--region us-east-1 ┌
```

## 2. Deploy It

```
aws cloudformation deploy --template-file /path_to_template/template.json --stack-name my-new-stack --parameter-overrides Key1=Value1 Key2=Value2 --tags Key1=Value1 Key2=Value2
```

```
→ static-webinator git:(hello-world-voxxed) ✘ aws \
--profile bear \
cloudformation deploy \
--template-file packaged.yaml \
--stack-name lambda-edge-sarah \
--capabilities CAPABILITY_IAM \
--region us-east-1
```

```
→ static-webinator git:(hello-world-voxxed) ✘ aws \
--profile bear \
cloudformation deploy \
--template-file packaged.yaml \
--stack-name lambda-edge-sarah \
--capabilities CAPABILITY_IAM \
--region us-east-1
```

Waiting for changeset to be created..

Waiting for stack create/update to complete

Successfully created/updated stack - lambda-edge-sarah

```
→ static-webinator git:(hello-world-voxxed) ✘ ┌
```

## Cloudfront Domain Name

http://d12p9yy4z18nsa.cloudfront.net/

# What Did I Use Lambda@Edge for?

# **My Problem**

1. Static website
2. Custom domain name
3. Secure
4. Index Document support

# **Index Document Support**

<http://example.com/books/index.html>

# Solutions



S3 Bucket

# Public S3 Bucket

1. Static website



2. Custom domain name



3. Index Document support



4. Secure



HTTP



Public S3  
Bucket

# Private S3 Bucket

1. Static website



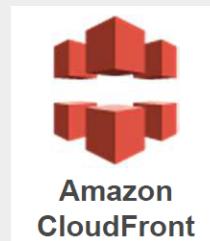
2. Custom domain name



3. Index Document support



4. Secure



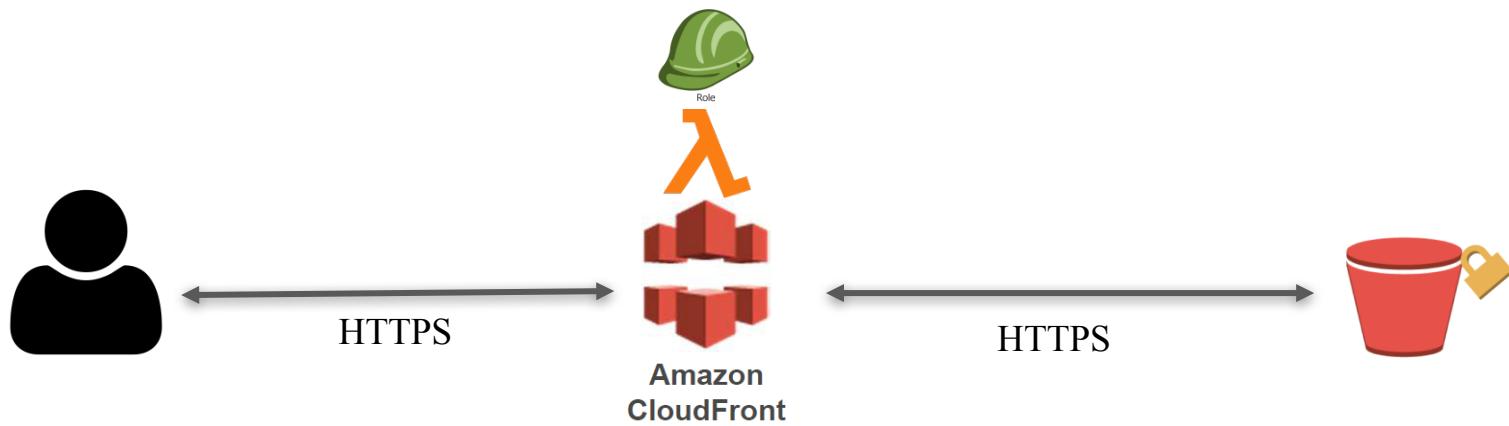
HTTPS

REST API



mydemo.net/mydemo/index.html  
**NOT**  
mydemol.net/mydemo

# THE Solution



URL redirects and  
rewrites at the  
Edge



[Buy](#)[Rent](#)[Invest](#)[Sold](#)[Share](#)[New homes](#)[Find agents](#)[Lifestyle](#)[News](#)[Commercial](#)[Sign in](#)[Join](#)

## Search properties for sale

[Buy](#)[Rent](#)[Sold](#)[Property value](#)[Find agents](#)[Search](#)[All property types](#) ▾[Beds \(min\)](#) ▾[Beds \(max\)](#) ▾[Price \(min\)](#) ▾[Price \(max\)](#) ▾ Surrounding suburbs

DOWNLOAD OUR APP

 [iPhone & iPad](#) [Android](#)[Learn more](#)

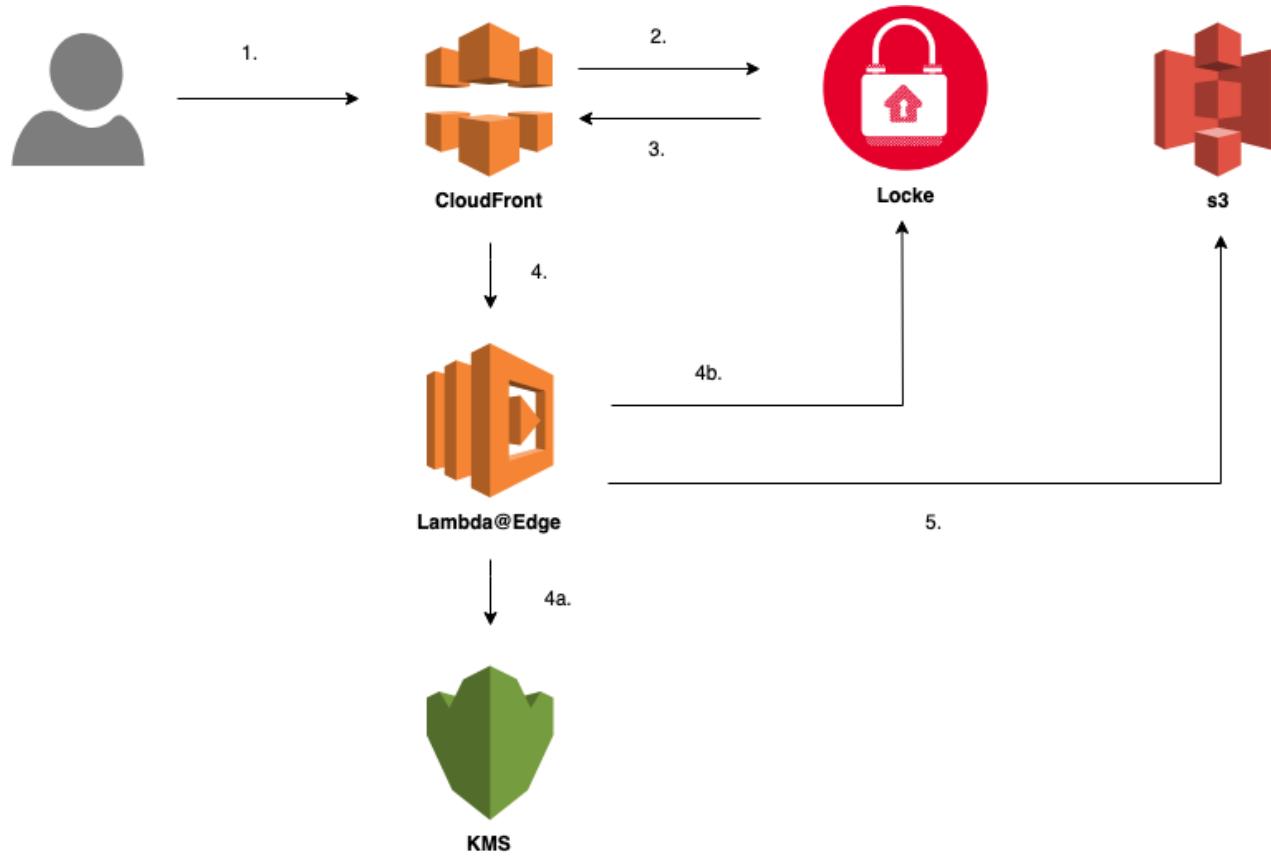
## Home Loans

Hundreds of home loans from lenders you know.

What Else is it Being  
used for?

# Improved Security

- Authentication@edge - better way to share content in the company (no more IP whitelists)
- No more public buckets



Thank You :)



Here's Some  
Code :D

[https://github.com/  
Narnie](https://github.com/Narnie)

# Beginner Pain

1. Logs end up in different regions 
2. Can only deploy your Lambda@Edge in **us-east-1**  
(all other regions replicate from this one.)
3. Lambda version update - use SAM or remember to  
update lambda function version manually

# Index Document Support

If uri ends with `/`

URL rewrite `uri/index.html`

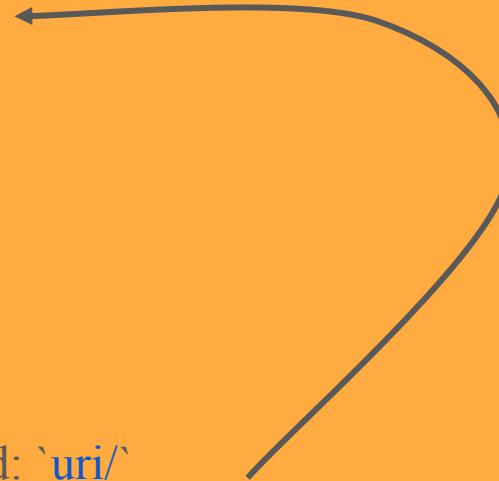
If uri does not end with `/`

If `uri/index.html` exists

302 redirect with `/` appended: `uri/`

Else

Pass original request to origin



## Limits that differ by event-type

| Entity                                                                                                                                                 | Origin request and response event limits | Viewer request and response event limits |
|--------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------|------------------------------------------|
| Function resource allocation                                                                                                                           | Same as Lambda limits                    | 128 MB                                   |
| Function timeout. The function can make network calls to resources such as Amazon S3 buckets, DynamoDB tables, or Amazon EC2 instances in AWS Regions. | 30 seconds                               | 5 seconds                                |
| Size of a response that is generated by a Lambda function, including headers and body                                                                  | 1 MB                                     | 40 KB                                    |
| Maximum compressed size of a Lambda function and any included libraries                                                                                | 50 MB                                    | 1 MB                                     |

# Lambda@Edge Pricing Details

There is no free tier for Lambda@Edge at this time.

You are charged for the total number of requests across all your functions. Lambda@Edge counts a request each time it starts executing in response to a CloudFront event globally.

Request pricing is **\$0.60 per 1 million requests** (\$0.0000006 per request).

Duration is calculated from the time your code begins executing until it returns or otherwise terminates. You are charged \$0.00005001 for every GB-second used. For instance, if you allocate 128MB of memory available per execution with your Lambda@Edge function, then your duration charge will be **\$0.00000625125** for every **128MB-second** used. Note that Lambda@Edge functions are metered at a granularity of 50ms.

## Requests

**\$0.60 PER 1M REQUESTS**

*\$0.0000006 per request*

## Duration

**\$0.00000625125 FOR EVERY 128MB-SECOND**

*Lambda@Edge functions are metered at a granularity of 50ms*

| General              | Origins and Origin Groups | Behaviors | Error Pages | Restrictions | Invalidations | Tags |
|----------------------|---------------------------|-----------|-------------|--------------|---------------|------|
| <a href="#">Edit</a> |                           |           |             |              |               |      |

|                                        |                                                              |
|----------------------------------------|--------------------------------------------------------------|
| <b>Distribution ID</b>                 | E3TE48J2FLRA17                                               |
| <b>ARN</b>                             | arn:aws:cloudfront::619438615114:distribution/E3TE48J2FLRA17 |
| <b>Log Prefix</b>                      | -                                                            |
| <b>Delivery Method</b>                 | Web                                                          |
| <b>Cookie Logging</b>                  | Off                                                          |
| <b>Distribution Status</b>             | Deployed                                                     |
| <b>Comment</b>                         | CI/CD for Lambda@Edge                                        |
| <b>Price Class</b>                     | Use All Edge Locations (Best Performance)                    |
| <b>AWS WAF Web ACL</b>                 | -                                                            |
| <b>State</b>                           | Enabled                                                      |
| <b>Alternate Domain Names (CNAMEs)</b> | -                                                            |
| <b>SSL Certificate</b>                 | Default CloudFront Certificate (*.cloudfront.net)            |
| <b>Domain Name</b>                     | d12p9yy4z18nsa.cloudfront.net                                |
| <b>Custom SSL Client Support</b>       | -                                                            |
| <b>Security Policy</b>                 | TLSv1                                                        |
| <b>Supported HTTP Versions</b>         | HTTP/1.1, HTTP/1.0                                           |
| <b>IPv6</b>                            | Enabled                                                      |
| <b>Default Root Object</b>             | -                                                            |
| <b>Last Modified</b>                   | 2019-05-12 23:07 UTC+10                                      |
| <b>Log Bucket</b>                      | -                                                            |

## COSTS:

<https://medium.com/@zackbloom/serverless-pricing-and-costs-aws-lambda-and-lambda-edge-169bfb58db75>

# Examples



<https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/lambda-examples.html>

