FIBO
CLOUD
EDUCATION

AWS Solutions Architect

Serverless, DynamoDB, Decoupling

Н. Ганжигүүр
Fibo Cloud

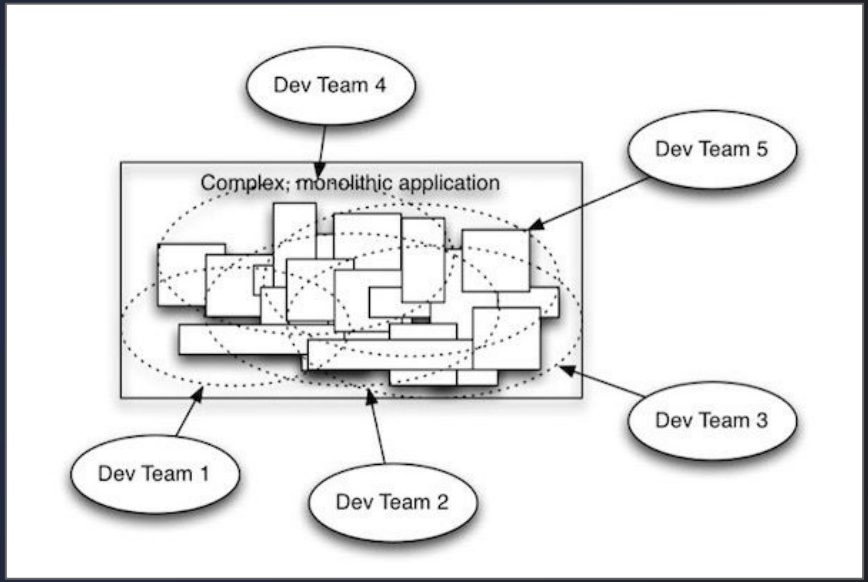
Monolith application

- Traditional method
- Assembled together (Tightly coupled)
- Single instance
- Easy to deploy (For ex: build one WAR deploy to Apache or Jetty)

Challenges of Monolithic

FIBO CLOUD EDUCATION

- Large & Complex
- Slow Development - IDE overload
- Hard to new developer
- Blocks CD
- Unscalable (Only X-axis)
- Unreliable - Single point failure
- Inflexible - Vendor lock-in



Challenges of Monolithic

FIBO CLOUD EDUCATION

- Large & Complex
- Slow Development - IDE overload
- Hard to new developer
- Blocks CD
- Unscalable (Only X-axis)
- Unreliable - Single point failure
- Inflexible - Vendor lock-in

Challenges of Monolithic

FIBO CLOUD EDUCATION

- Large & Complex
- Slow Development - IDE overload
- Hard to a new developer
- Blocks CD
- Unscalable (Only X-axis)
- Unreliable - Single point failure
- Inflexible - Vendor lock-in

Challenges of Monolithic

FIBO CLOUD EDUCATION

- Large & Complex
- Slow Development - IDE overload
- Hard to new developer
- Blocks CD
- Unscalable (Only X-axis)
- Unreliable - Single point failure
- Inflexible - Vendor lock-in

Challenges of Monolithic

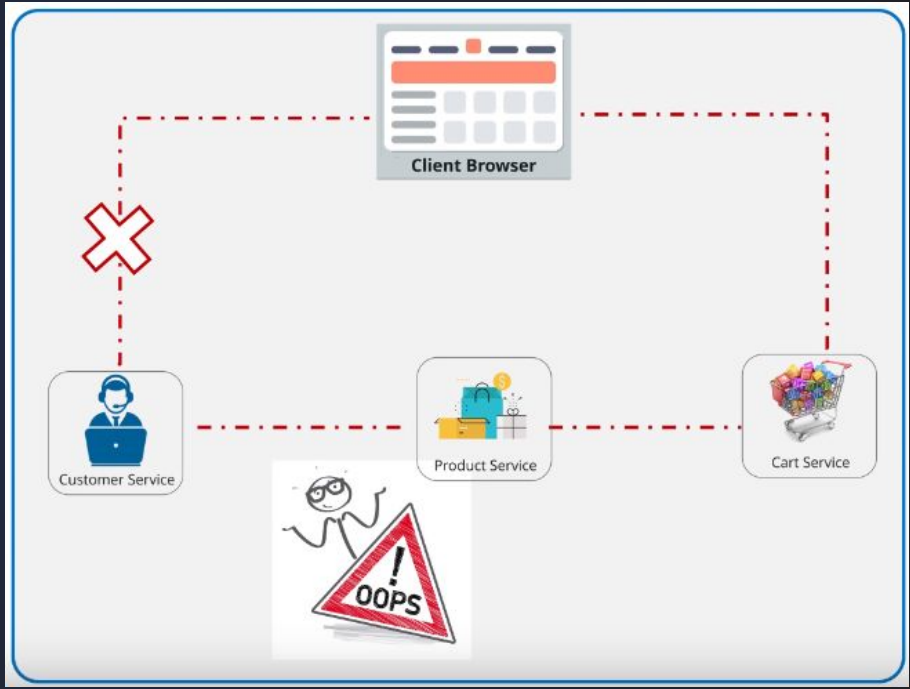
FIBO CLOUD EDUCATION

- Large & Complex
- Slow Development - IDE overload
- Hard to new developer
- Blocks CD
- Unscalable (Only Y-axis)
- Unreliable - Single point failure
- Inflexible - Vendor lock-in

Challenges of Monolithic

FIBO CLOUD EDUCATION

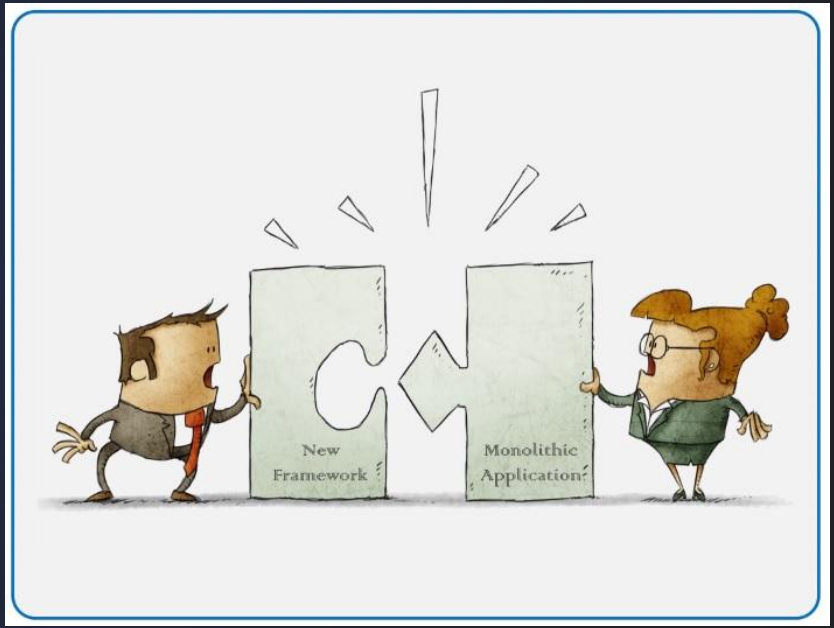
- Large & Complex
- Slow Development - IDE overload
- Hard to new developer
- Blocks CD
- Unscalable (Only X-axis)
- Unreliable - Single point failure
- Inflexible - Vendor lock-in



Challenges of Monolithic

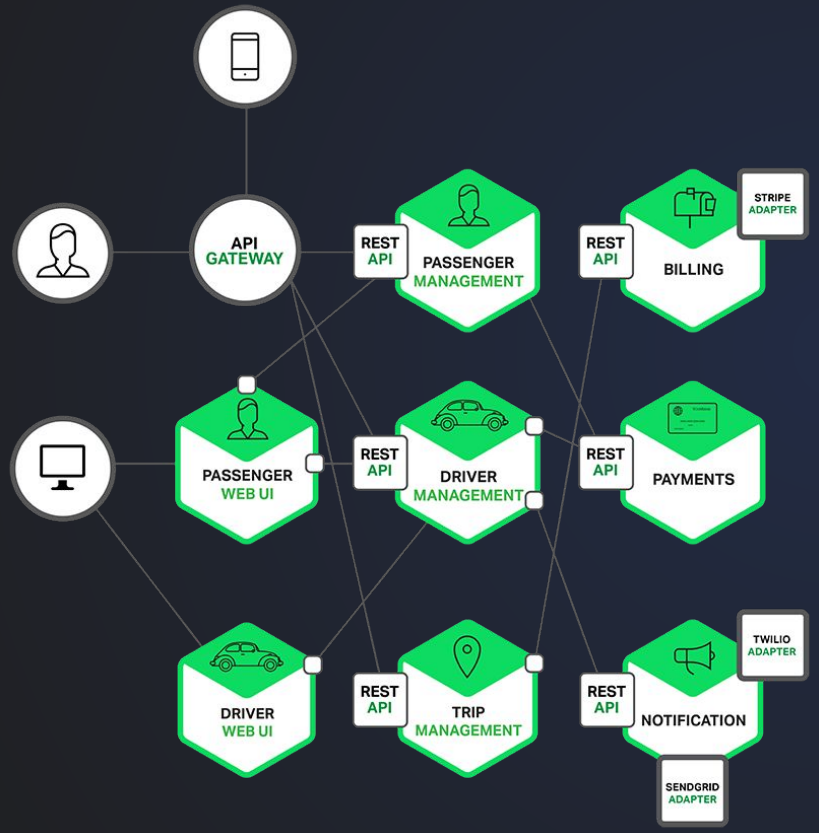
FIBO CLOUD EDUCATION

- Large & Complex
- Slow Development - IDE overload
- Hard to new developer
- Blocks CD
- Unscalable (Only X-axis)
- Unreliable - Single point failure
- Inflexible - Vendor lock-in

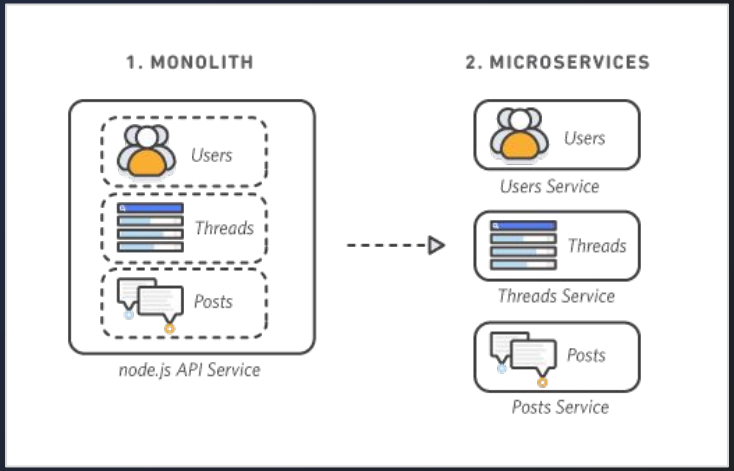


Microservices

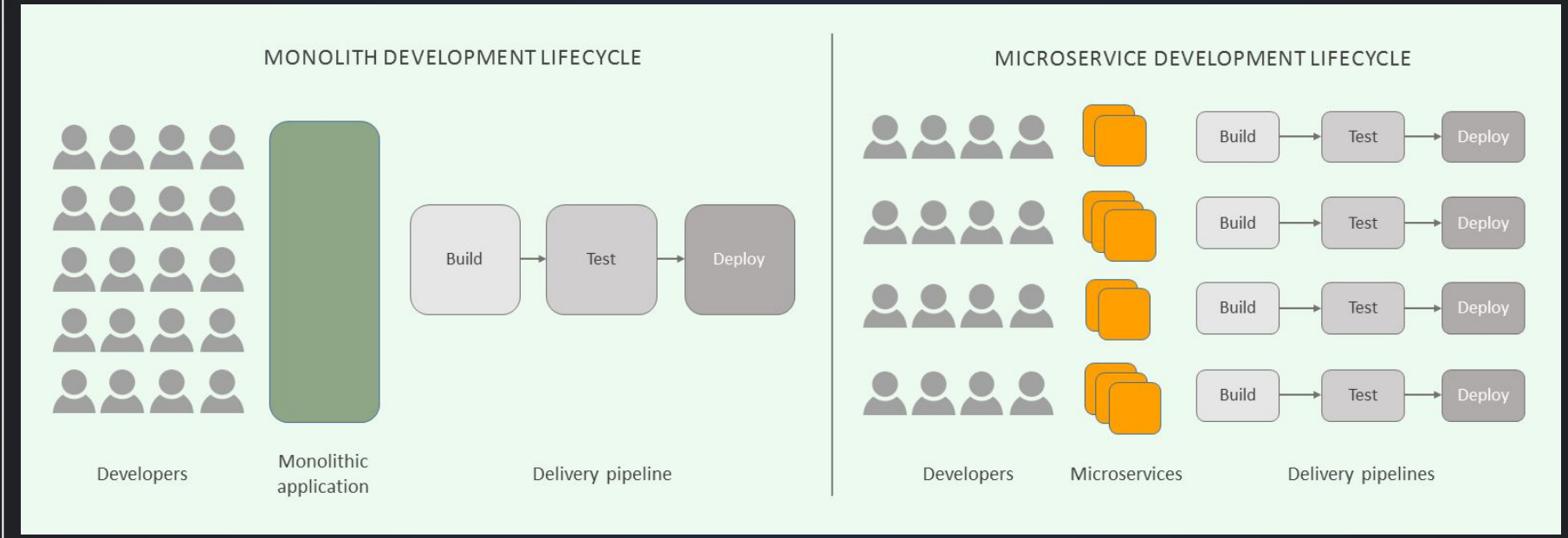
FIBO CLOUD EDUCATION



Is a self contained autonomous process provides a unique business capability



Microservices



- Small focused
- Development Deployment quick (Loosely coupled)
- Mixed technology
- Bounded context - privacy
- Easy to new developer
- Fault isolation
- Granular scaling (Independent scale)

Things on Microservices

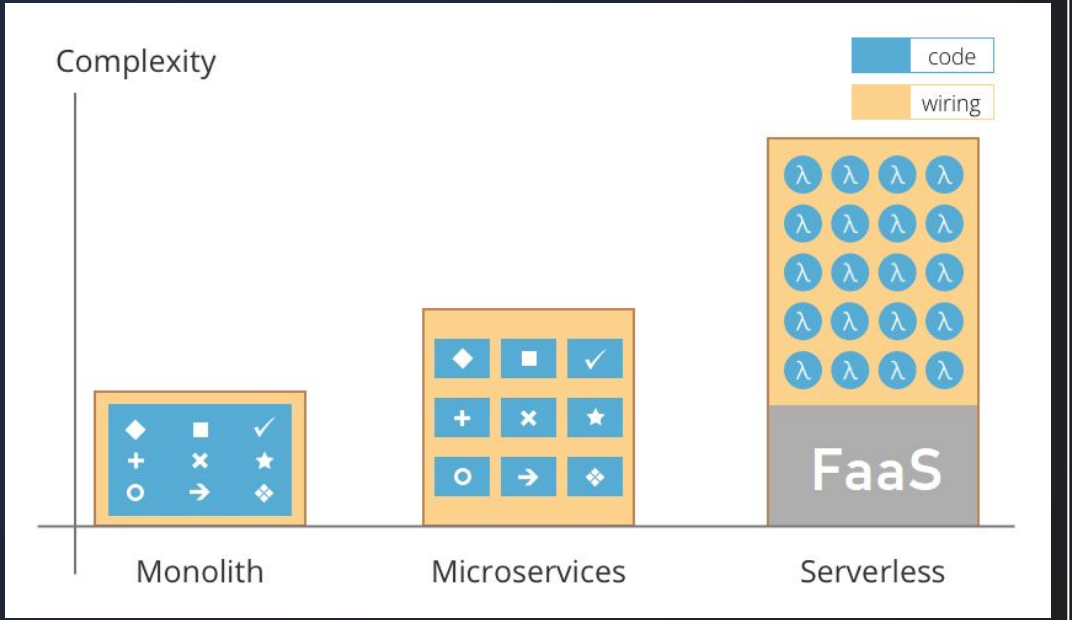
FIBO CLOUD EDUCATION

- API gateway
- Service discovery - Registrar
- Messaging (SQS, AQS, RabbitMQ...)
- Division size (Complexity)
- Downgrading call time
- Could be their own DB (more challenging to developers!)
- Testing & Hard to define bugs (Spider web)
- Deploying difficulty - High level of automation - Logical order (A-B-C A-C-B)
- Container - Microservices + Kubernetes + Docker = Best combination
- Must be well architected - Art of Engineering

Timeline

FIBO CLOUD EDUCATION

1. Do it all by yourselves
2. Virtualization
3. IaaS
4. PaaS
5. CaaS
6. FaaS (event driven apps)



Serverless

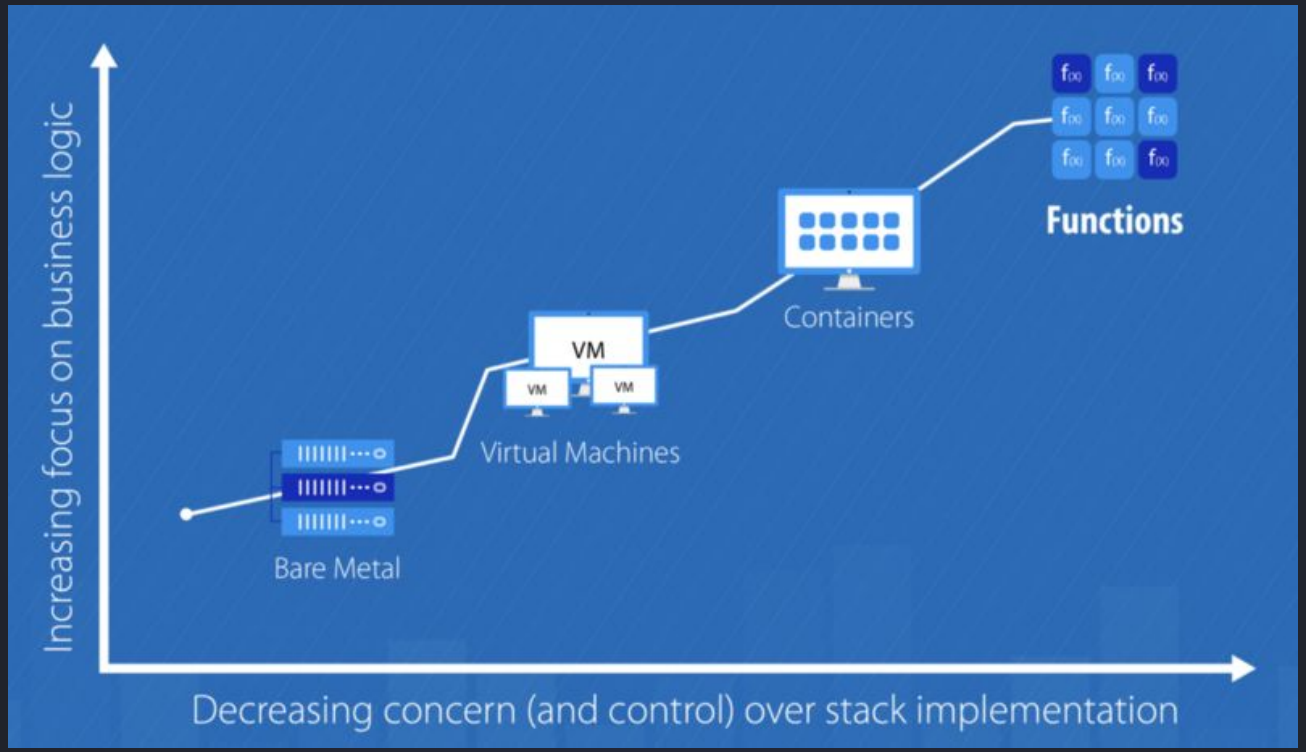
FIBO CLOUD EDUCATION

Traditional architecture

- Create setup servers
- Install OS
- Patch
- Software environment
- Architect for highly availability
- Architect for scalable

Serverless architecture

- No servers
- No OS
- And also no related softwares
- Efficient scalability
- High availability


Increasing focus on business logic
Functions
Containers
VM
VM
VM
Virtual Machines
Bare Metal
Decreasing concern (and control) over stack implementation

- Container easy to migrate to another
- Serverless - Size & memory constraint
- Full control with containers such as managing resources, setting policies and controlling security
- Testing , debugging, monitoring not easy on serverless
- Serverless **vendor lock-in !!!**
- Containers are more expensive than serverless
- Container should be monitored

Stateless function

FIBO
CLOUD
EDUCATION

AWS Serverless

Serverless stack

FIBO CLOUD EDUCATION

Service	Rate	Cost
Cognito	Free[1]	$0.00
API Gateway	$3.5/M reqs + $0.09/GB transfer	$2.20
Lambda	Free[2]	$0.00
DynamoDB	$0.0065/hr 10 write units, $0.0065/hr 50 read units[3]	$2.80
S3	$0.023/GB storage, $0.005/K PUT, $0.004/10K GET, $0.0025/M objects[4]	$0.24
CloudFront	$0.085/GB transfer + $0.01/10K reqs	$0.86
Route53	$0.50 per hosted zone + $0.40/M queries	$0.50
Certificate Manager	Free	$0.00
Total		**$6.10**

Benefits

FIBO
CLOUD
EDUCATION

- No servers
- Scaling automatically
- Super cheap

FIBO
CLOUD
EDUCATION

AWS Lambda

AWS Lambda is a compute service where you can upload your code and create a Lambda function. AWS Lambda takes care of provisioning and managing the servers that you use to run the code. You don't have to worry about OS, patching, scaling etc

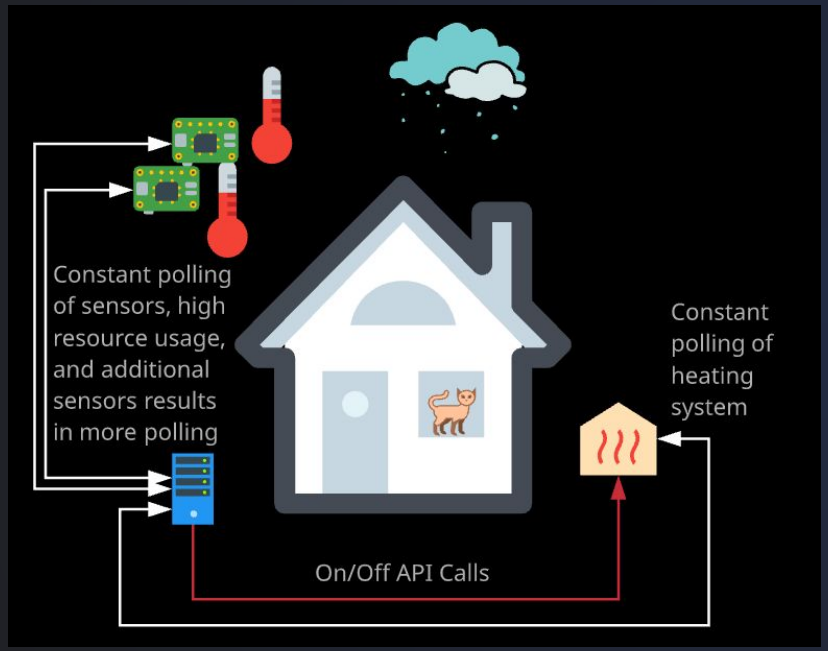
Abstraction layer
- Datacenter
- Hardware
- Assembly Code/ Protocols
- High level languages
- Operating systems
- Application layer/AWS APIs

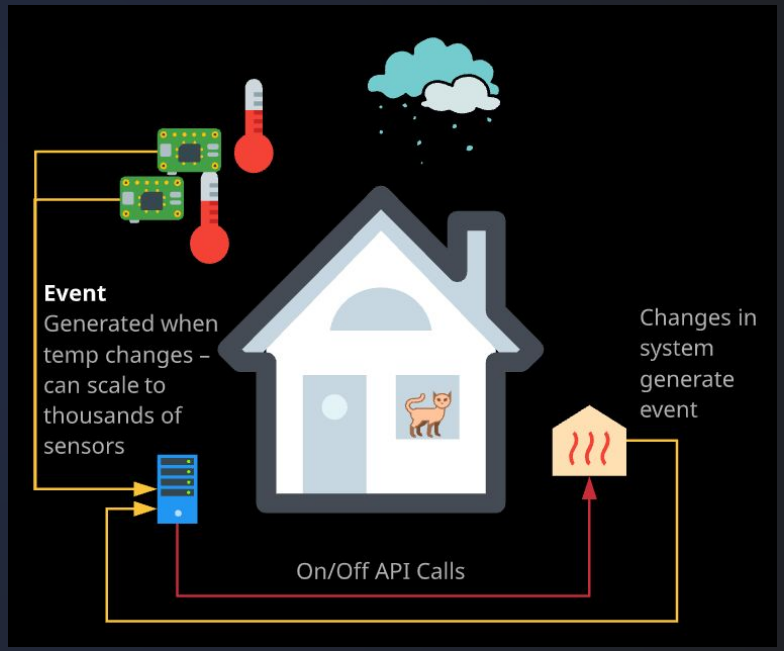
Function as a service
Only take care about our codes

Use following way:
- Event (S3, DynamoDB)
- HTTP (API gateway, AWS SDK)

Traditional vs Event driven

FIBO CLOUD EDUCATION



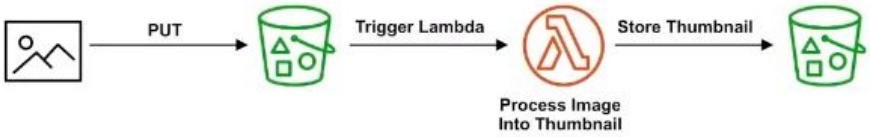


Typical architecture

Lambda is commonly used to **glue different services together** so the use cases are endless.

Processing Thumbnails

A web-service allows users to upload their profile photo. They are stored in an S3 bucket. We can setup an Event Trigger which will invoke a Lambda which will process the Profile Photo into a Thumbnail and the store it back in the bucket.



Contact Email Form

A company has a contact email form which submits form data via API Gateway Endpoint. That endpoint triggers a lambda which validates the form data and if valid will save the submission in DynamoDB and send and email notification via SNS to the company

AWS Lambda is a compute service that lets you run code **without provisioning** or **managing** servers.

Lambda executes your code only when needed and scales automatically to a few to a 1000 lambda functions concurrently is seconds.

.

You pay only for the compute time you consume - there is no charge when your code is not running.

Lambda is **Cheap**

Lambda is **Serverless**

Lambda **Scales Automatically**

Natively supports 7 runtimes languages:

1. Ruby
2. Python
3. Java
4. Go
5. Powershell
6. NodeJs
7. C#

You can also create your own **custom runtime** environments

Lambdas can be **invoked** via the AWS SDK or or trigger from other AWS Services.
(This is not a complete list)

API Gateway
api application-services aws serverless

AWS IoT
aws devices iot

Alexa Skills Kit
alexa iot

Alexa Smart Home
alexa iot

Application Load Balancer
aws load-balancing

CloudFront
aws cdn edge

CloudWatch Events
aws events management-tools

CloudWatch Logs
aws logging management-tools

CodeCommit
aws developer-tools git

Cognito Sync Trigger
authentication aws identity mobile-services sync

DynamoDB
aws database nosql

Kinesis
analytics aws streaming

S3
aws storage

SNS
aws messaging notifications pub-sub push

SQS
aws queue

FIBO CLOUD EDUCATION

Partner event sources (powered by Amazon EventBridge)

Datadog
Datadog is the essential monitoring platform for cloud applications.

OneLogin
OneLogin, the leader in Unified Access Management, connects people with technology through a simple and secure login, empowering organizations to access the world.

PagerDuty
PagerDuty helps AWS users automatically turn any signal into the right insight and action.

Saviynt
Saviynt enables enterprises to secure applications, data and infrastructure in a single platform for cloud and enterprise.

Segment
Segment provides the customer data infrastructure that businesses use to put their customers first.

SignalFx
SignalFx, the only real-time cloud monitoring platform for infrastructure, microservices, and applications, collects and analyzes metrics and traces across every component in your cloud environment.

SugarCRM
SugarCRM enables businesses to create extraordinary customer relationships with the most empowering, adaptable and affordable customer relationship management (CRM) solution on the market.

Whispir
Whispir is a cloud based platform that automates, personalises and layers communications using smart workflow technology.

Zendesk
Zendesk makes better customer service experiences for agents, admins, and customers.

FIBO CLOUD EDUCATION

First **1 million requests** per month are free.
There-after **$0.20** per additional 1 million requests

400,000 GB seconds free per month
Thereafter **$0.0000166667** for every GB second

**This price will vary on the amount of memory you allocate

128MB of Memory **X** 30M executed per month **X** 200ms run time per invocation = **$5.83**

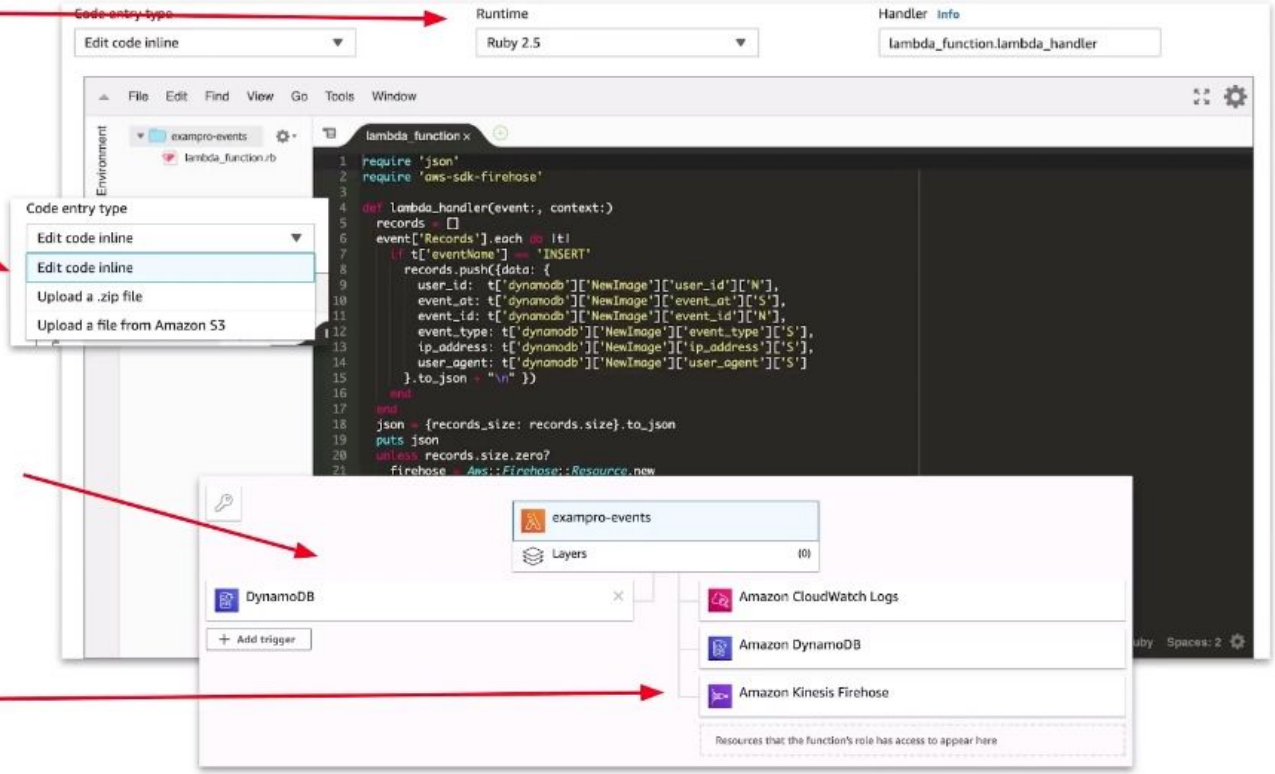
FIBO CLOUD EDUCATION

You choose you runtime

You upload Your code

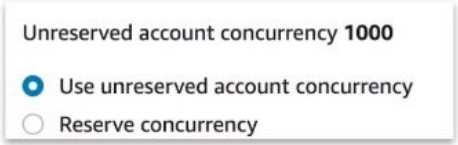
You choose your triggers

You grant permissions for outputs via an IAM Role



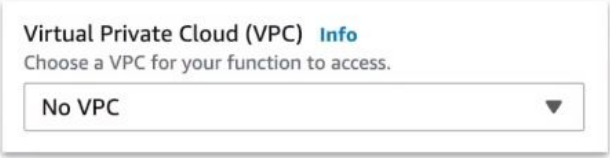
FIBO CLOUD EDUCATION

By Default you can have 1000 Lambda running concurrently (Ask AWS Support for Limit Increase)

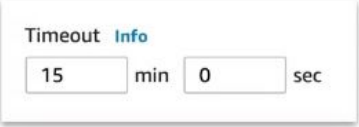


/tmp directory can contain up to **500MB**

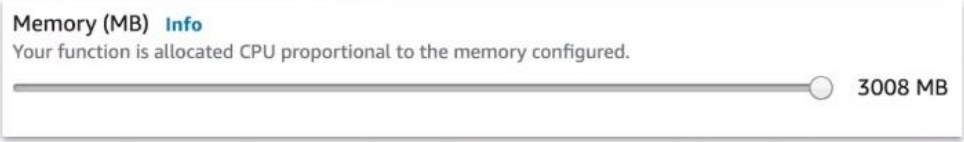
By Default Lambda run in No VPC. You can set them to by in your own VPC but your lambda will lose internet access



You can set timeout to be a maximum of **15 minutes**



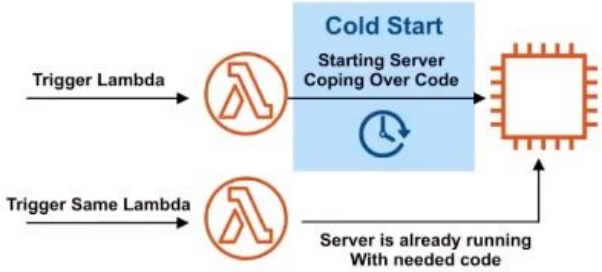
Memory can be set between **128MB** to a Maximum of **3008MB** at an increment of **64MB**



AWS has servers preconfigured (just sitting around turned off) for your runtime environment.
When a Lambda is invoked these servers need to be turned on and your code needs to be copied over.

During the time there will be a delay when the function will initially run which is called a **Cold Start**

If the same Lambda is invoked and the server is still running it will use that server again, so there will be little to delay to running that function. This what we call a **Warm Server**



Serverless functions are **cheap** but everything comes with a trade off. Serverless functions Cold Starts can **cause delays in the User Experience**. If your web-application relies on being very responsive, than you want to reconsider Serverless functions.

There are strategies around Cold Starts such as **Pre Warming** which keep servers continuously running. Cloud Providers are always looking for ways to reduce cold starts.

Cold starts

FIBO CLOUD EDUCATION

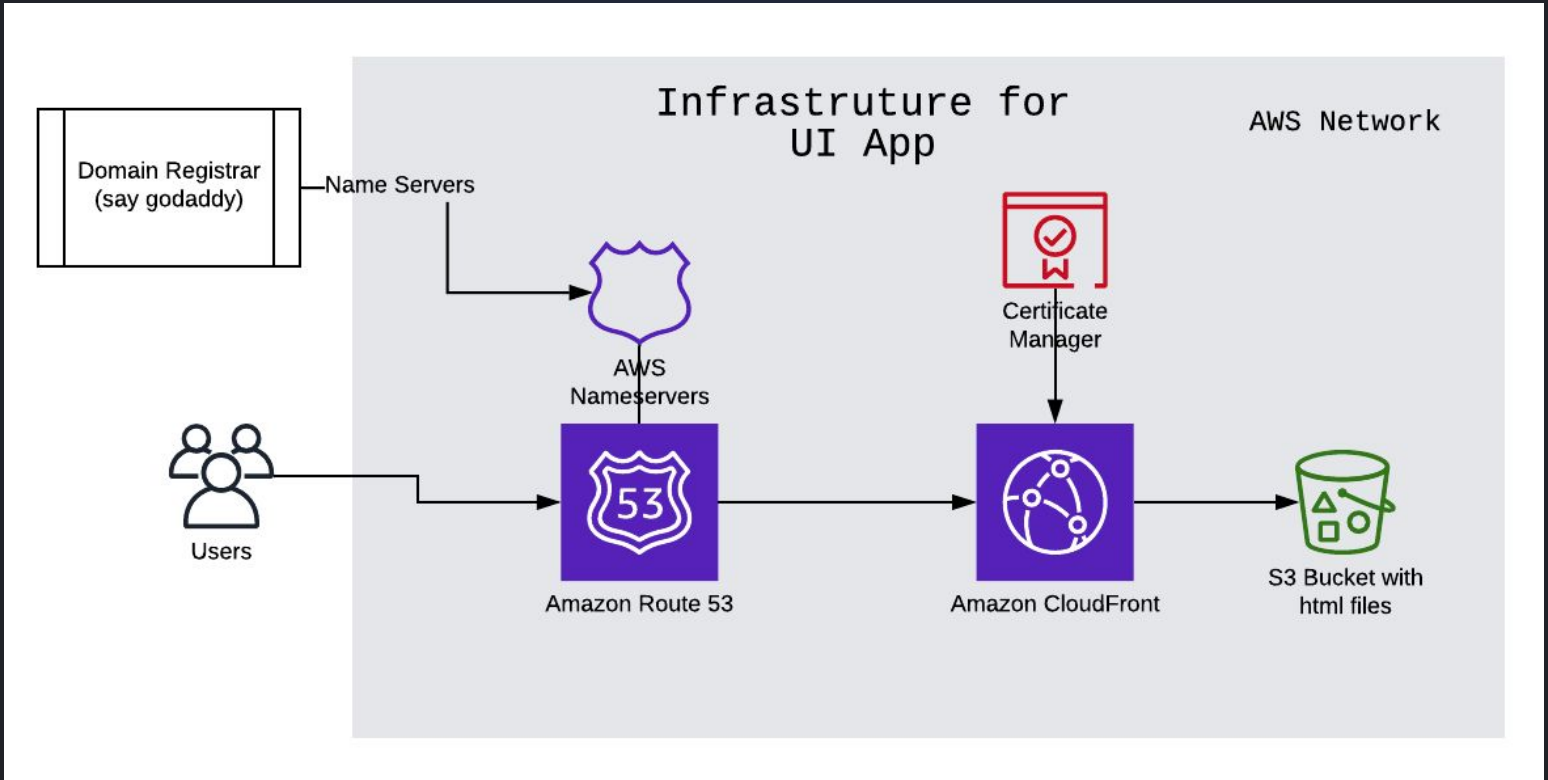
#	Resource	Timeline
1	<default>	809ms
2	<default>	369ms
3	<default>	130ms
4	<default>	151ms
5	<default>	151ms
6	<default>	152ms
7	<default>	132ms
8	<default>	153ms
9	<default>	133ms
10	<default>	136ms
11	<default>	139ms
12	<default>	139ms
13	<default>	129ms
14	<default>	170ms
15	<default>	245ms
16	<default>	167ms
17	<default>	123ms
18	<default>	129ms
19	<default>	134ms
20	<default>	124ms
21	<default>	158ms
22	<default>	139ms
23	<default>	187ms
24	<default>	200ms
25	<default>	134ms
26	<default>	148ms
27	<default>	136ms
28	<default>	157ms
29	<default>	136ms
30	<default>	160ms

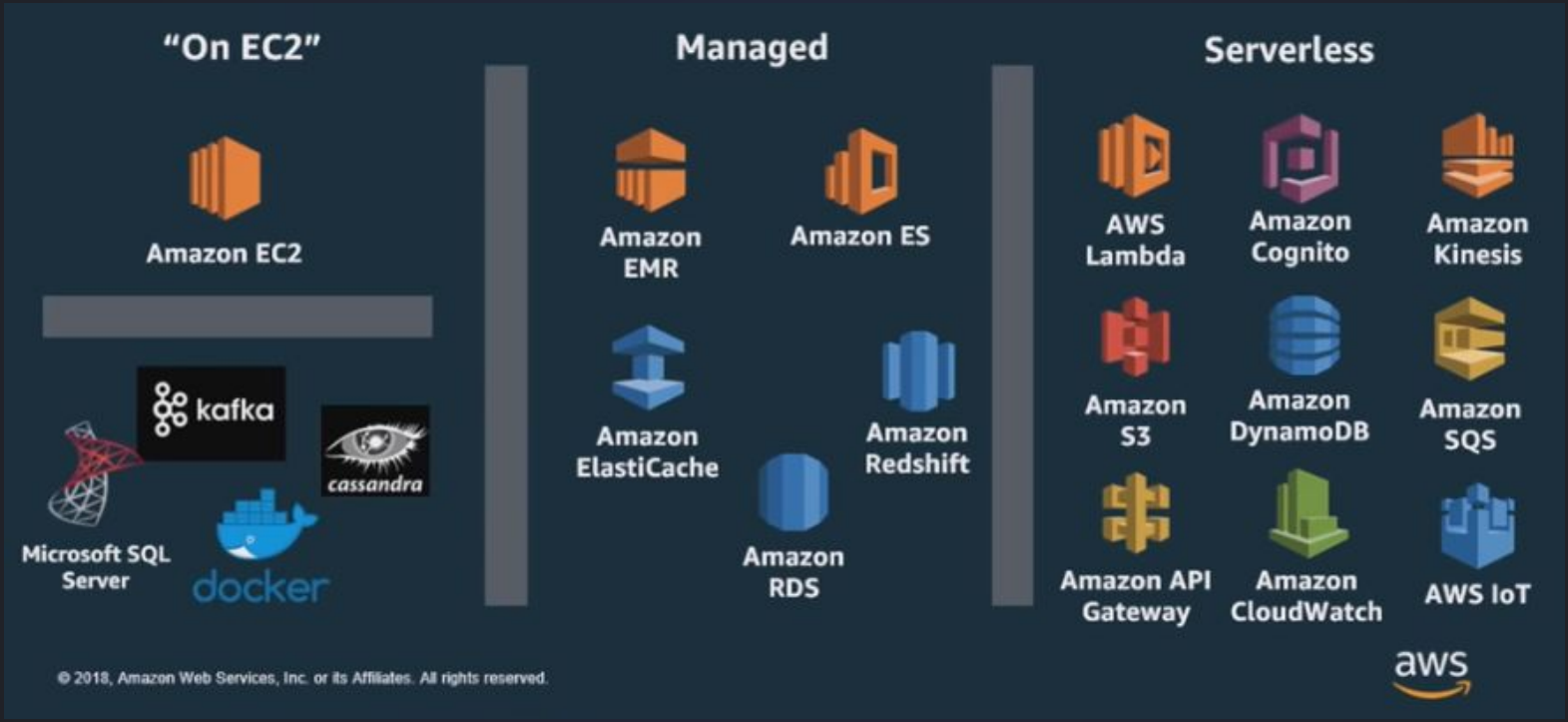
Timeline | Sizes | Duration | Types

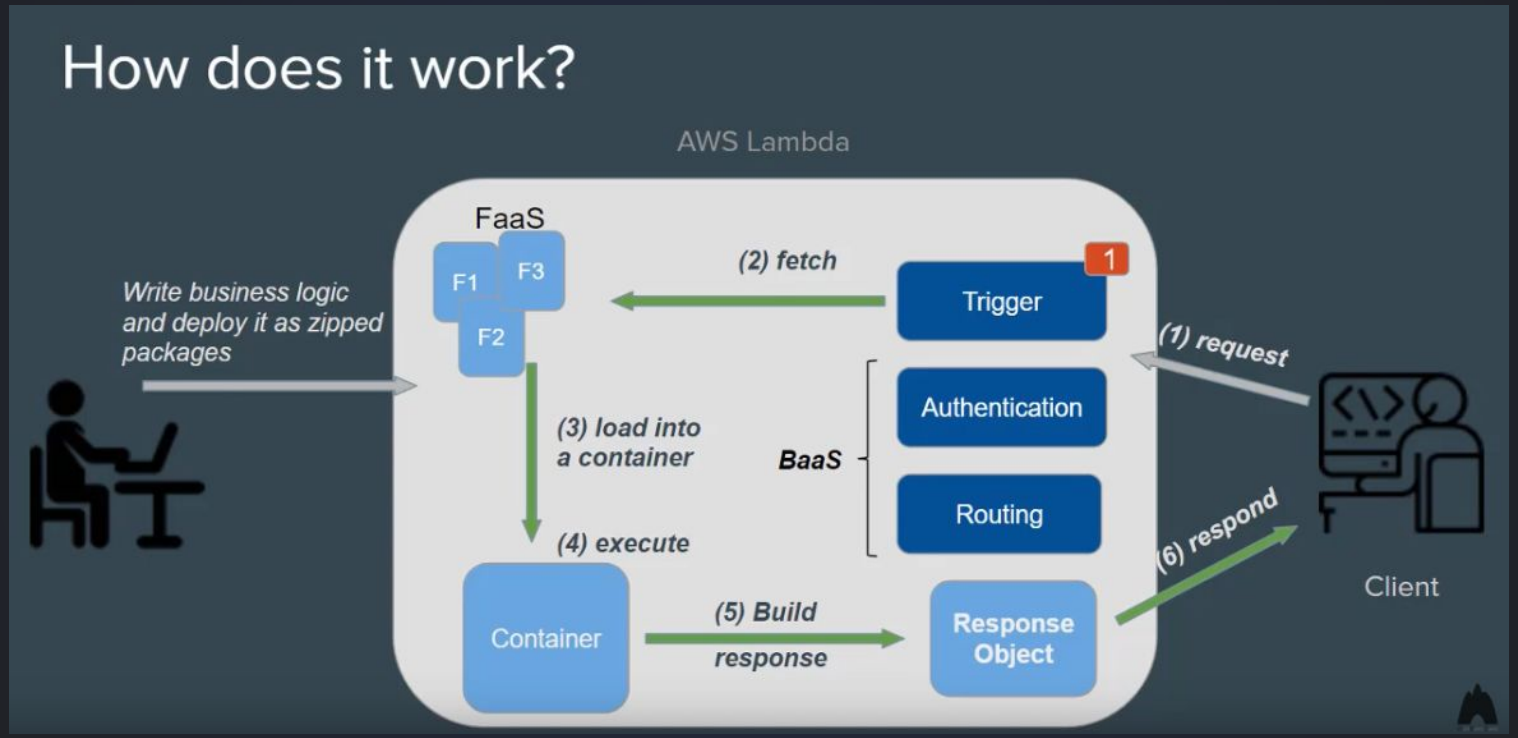
FIBO
CLOUD
EDUCATION

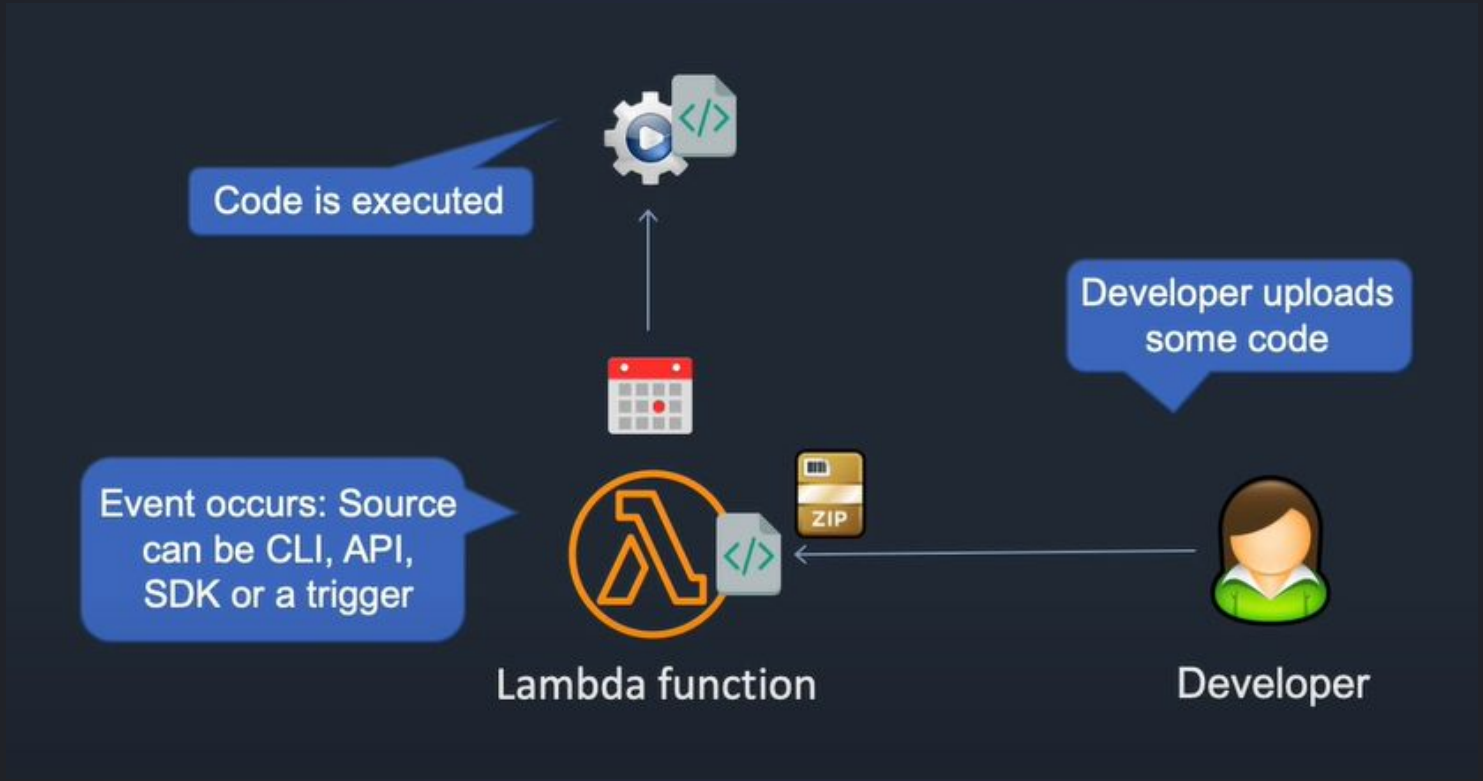
S3 static website

S3 static website




"On EC2"
Amazon EC2
kafka
cassandra
Microsoft SQL Server
docker
Managed
Amazon EMR
Amazon ES
Amazon ElastiCache
Amazon Redshift
Amazon RDS
Serverless
AWS Lambda
Amazon Cognito
Amazon Kinesis
Amazon S3
Amazon DynamoDB
Amazon SQS
Amazon API Gateway
Amazon CloudWatch
AWS IoT
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws


How does it work?
AWS Lambda
FaaS
F1
F3
F2
Write business logic and deploy it as zipped packages
(2) fetch
Trigger
1
(1) request
Authentication
BaaS
Routing
(3) load into a container
(4) execute
Container
(5) Build response
Response Object
(6) respond
Client


Code is executed
Developer uploads some code
Event occurs: Source can be CLI, API, SDK or a trigger
ZIP
Lambda function
Developer

API Gateway

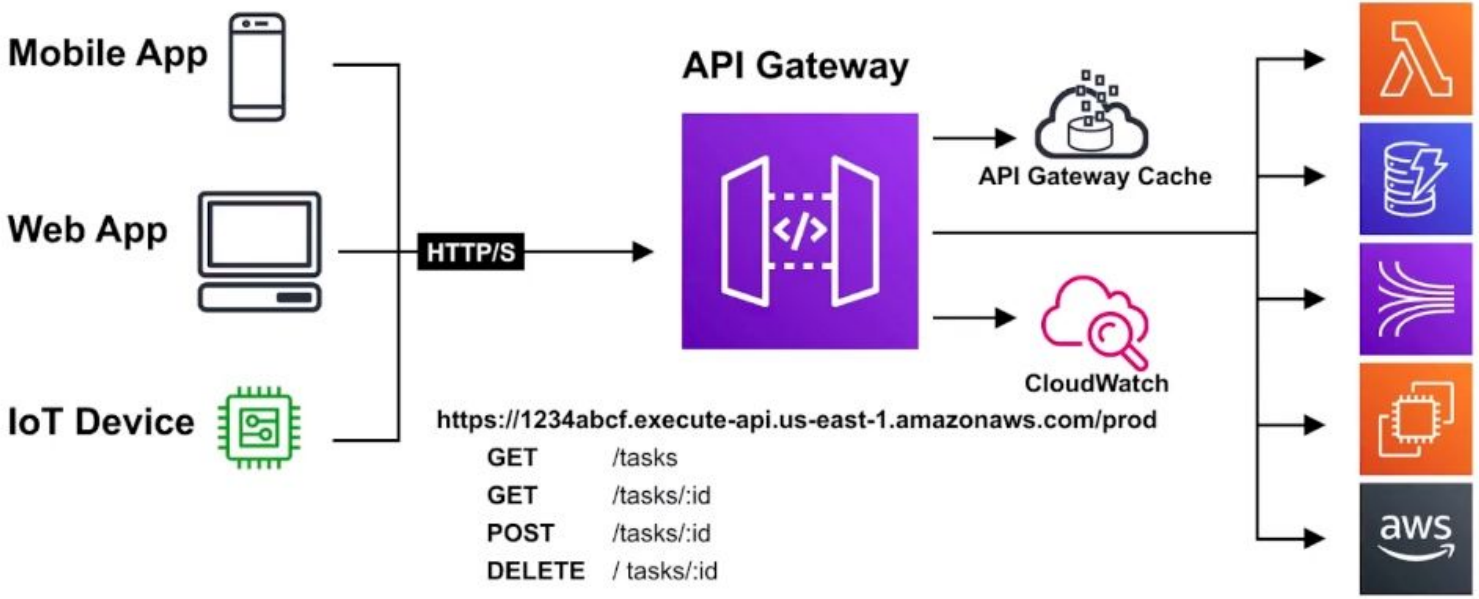
FIBO
CLOUD
EDUCATION

What is an API Gateway?

FIBO CLOUD EDUCATION

API Gateway is a solution for **creating secure APIs** in your cloud environment at **any scale.**

Create APIs that act as a **front door** for applications to **access data, business logic,** or **functionality from back-end services.**



API Gateway handles all the tasks involved in accepting and processing **up to hundreds of thousands of concurrent API calls**, including traffic management, authorization, and monitoring.

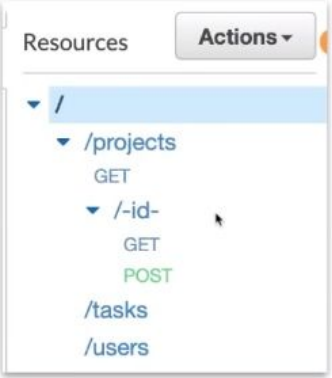
Allows you to **track** and **control any usage** of the API. **Throttle** requests to help **prevent attacks**.

Expose HTTPS endpoints to define a **RESTful API.**

Highly scalable (**happens automatically**) and cost effective.

Send each API endpoint to a different target.

Maintain **Multiple Versions** of your API.

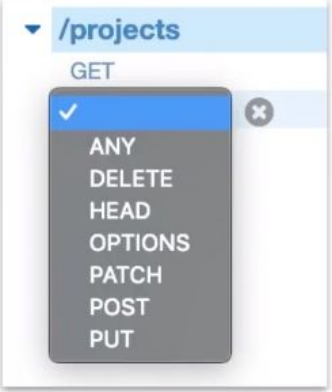


Resources

When you create an API you need to also create multiple **Resources.**

Resources are the urls you define eg. /projects

Resources can have child resources eg. /projects**/-id-/edit**



Methods

You need to define **Methods** on Resources

You can define multiple **Methods** on a Resource

Methods allow you to make API calls that resource url with that protocol

eg.

GET /projects/-id-

POST /projects/-id-

Stages

In order to use your API you need to Deploy it to Stages

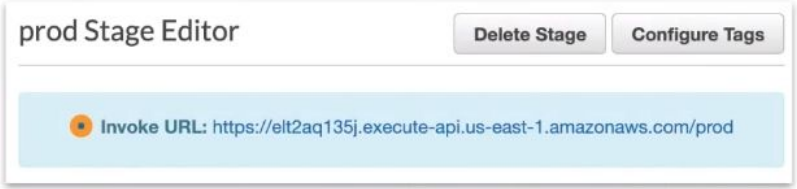
Stages are versions of your API

Invoke URL

For each stage AWS provides you a Invoke URL

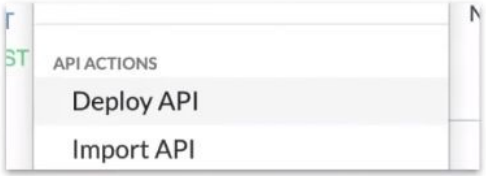
This is where you'll make your API calls.

It is possible to use a custom domain for your Invoke URL



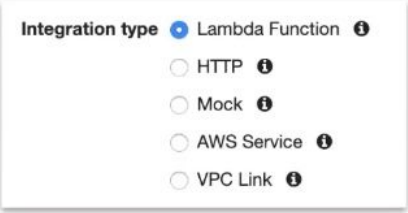
Deploy API

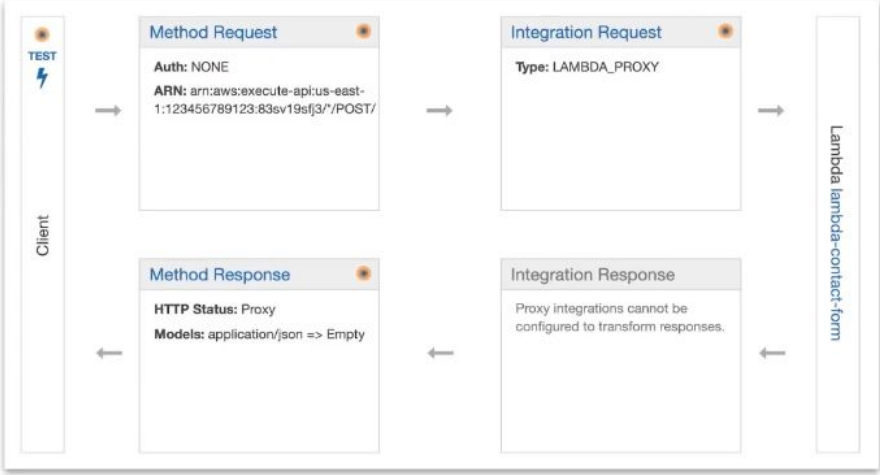
Everytime you make a change to your API you need to Deploy it via the **Deploy API** action. When you deploy you choose the stage



When you create an API Method on a resource you need to choose the Integration type.

The most common Integration type is **Lambda**



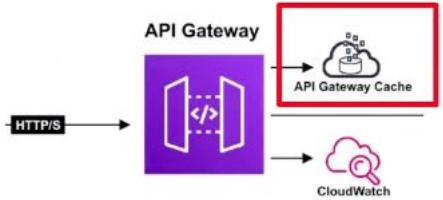


You have fine tune control over the **Request** and **Response** for the Method Execution.

API Gateway caching

FIBO CLOUD EDUCATION

API Caching can be enabled to cache your endpoints response to API calls.



- When enabled on a stage, API Gateway caches responses from your endpoint for a specified **time-to-live (TTL)** period.
- API Gateway responds to requests by **looking up the response from the cache**. (instead of making a request to the endpoint)

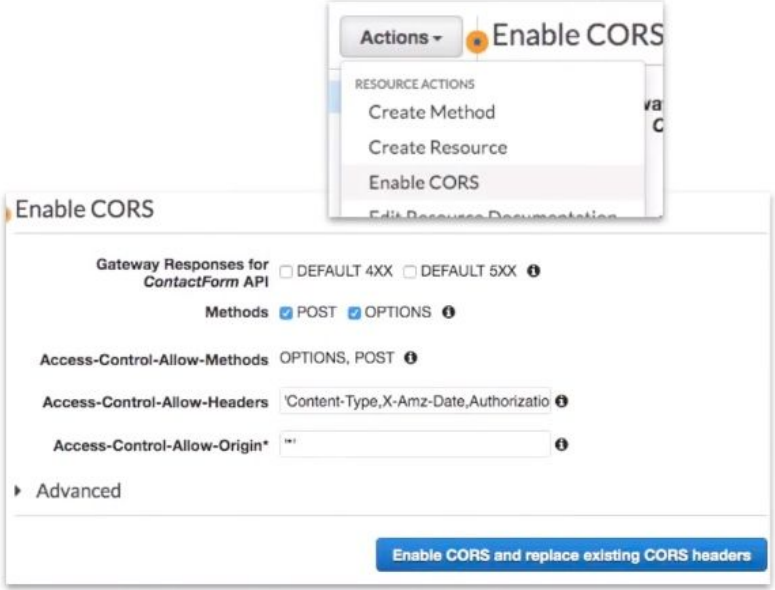
Reduces the number of calls made to your endpoint.

Improves latency of the requests made to your API.

Cors

FIBO CLOUD EDUCATION

Cross-Origin Resource Sharing (CORS) is a way that the server at the other end (not client code in the browser) can relax a same-origin policy.



Allows restricted resources (ie Fonts) on a webpage to be requested from a **different domain than the initial resource** that it came from.

Should always be enabled if using Javascript/AJAX that uses multiple domains with an API gateway.

CORS is **always** enforced by the client.

Same origin policy

FIBO CLOUD EDUCATION

Same Origin Policy is a concept in the application security model, where a web browser permits **scripts contained in a first web page,** to access **data in a second webpage.**

- Same Origin Policies are used to help prevent Cross-Site Scripting (XSS) attacks.
- They only works if both web pages have the same origin
- They are enforced at the web browser level
- They ignore tools such as Postman or Curl

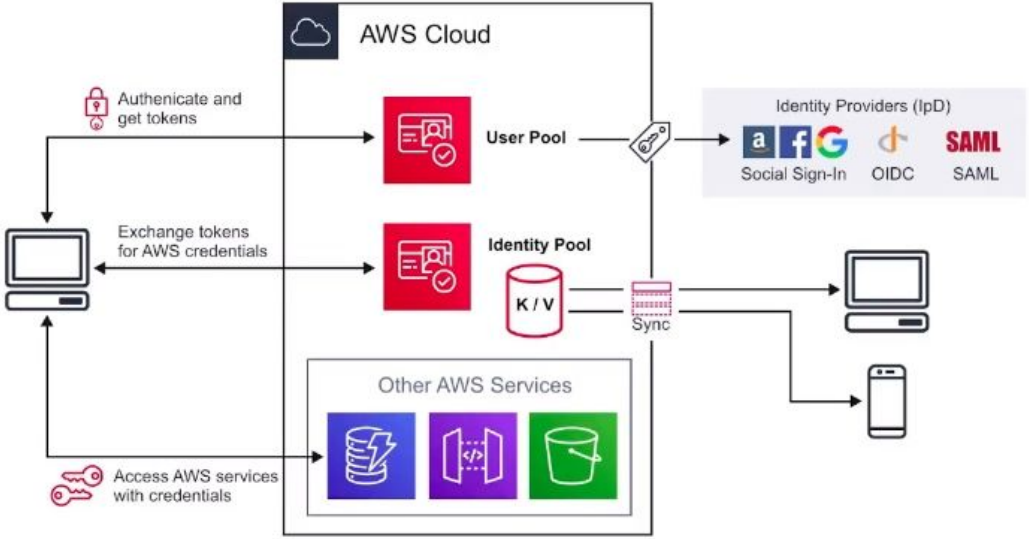
FIBO
CLOUD
EDUCATION

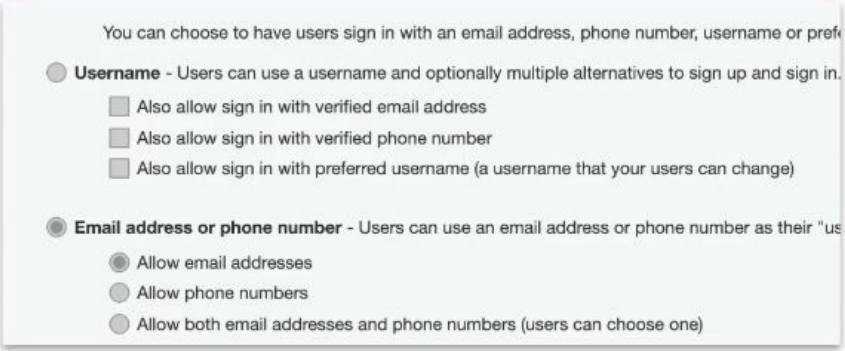
AWS Cognito

Cognito User Pools
User directory with authentication to IpD to grant access to your app

Cognito Identity Pools
Provide temporary credentials for users to access AWS Services

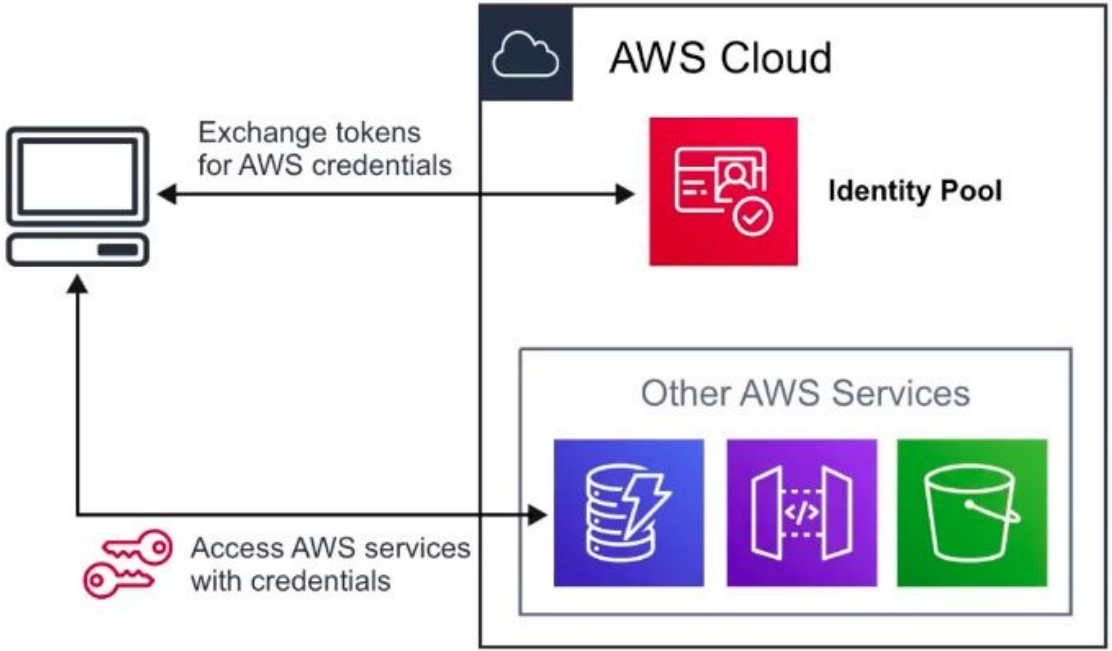
Cognito Sync
Syncs user data and preferences across all devices





- Choose what attributes
- Choose password requirements
- Apply MFA
- Restrict whether users are allow to sign up on their own or need admin verification
- Analytics with PinPoint for user campaigns
- Trigger custom log via Lambdas after actions such as after signup

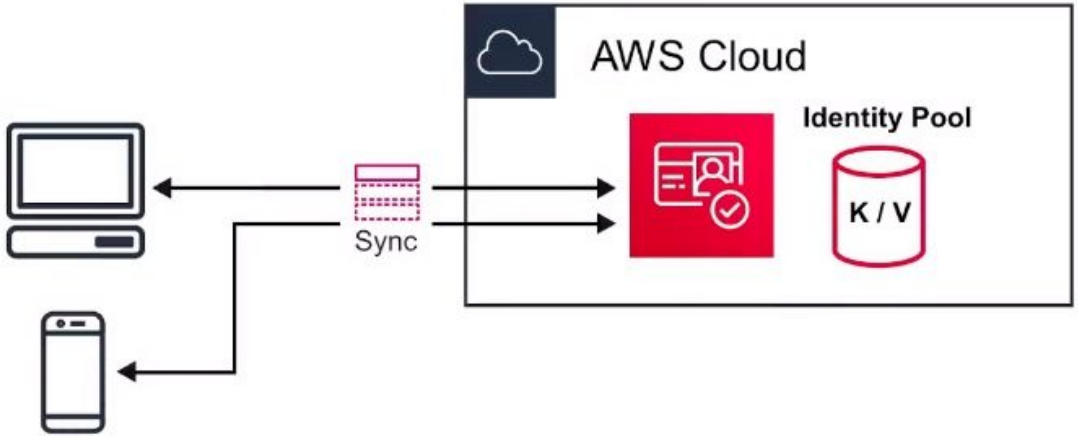
Identity Pools provide **temporary AWS credentials** to access services eg. S3, DynamoDB
Identity Pools can be thought of as the actual mechanism authorizing access to the AWS resources.



Sync **user data** and **preferences** across devices with one line of code

Cognito uses **push synchronization** to push updates and synchronize data

Uses Simple Notification Service (SNS) to send notifications to all user devices when data in the cloud changes.



FIBO
CLOUD
EDUCATION

Dynamo DB

FIBO CLOUD EDUCATION

What is NoSQL

NoSQL is database which is neither relational and does not use SQL to query the data for results

What is a Key/Value Store?

A form of data storage which has a key which references a value and nothing more

```
{ Title: 'S01E019 DS9 Duet' }
```

What is a Document Store?

A form of data storage which a nested data structure

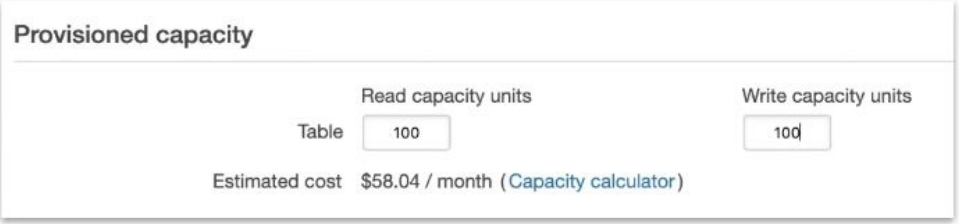
```
{
  Series: 'DS9'
  Episodes: [
    {
      Season: 1,
      Epsisode: 19,
      Title: 'Duet'
    }
  ]
}
```

DynamoDB is a NoSQL **key/value** and **document** database for internet-scale applications.

Features

- Fully managed
- Multiregion
- Multimaster
- Durable database
- Built-in security
- Backup and restore
- In-memory caching

Specific your read and write capacity per second, it just works at **whatever capacity you need** without you tweaking anything.



Provides

- Eventual Consistent Reads (default)
- Strongly Consistent Reads

All data is stored on **SSD storage** and is spread across **3 different regions**.

FIBO
CLOUD
EDUCATION

Primary Key

Partition Key | Sort Key

IMDB ID	Year	Title	Box Office
tt0079945	1979	Star Trek: The Motion Picture	139000000
tt0084726	1982	Star Trek II: The Wrath of Khan	97000000
tt0088170	1984	Star Trek III: The Search for Spock	87000000
tt0092007	1986	Star Trek IV: The Voyage Home	133000000
tt0098382	1989	Star Trek V: The Final Frontier	63000000
tt0102975	1991	Star Trek VI: The Undiscovered Country	96900000
tt0111280	1994	Star Trek Generations	118000000
tt0117731	1996	Star Trek: First Contact	146000000
tt0120844	1998	Star Trek: Insurrection	117800000
tt0253754	2002	Star Trek: Nemesis	67300000

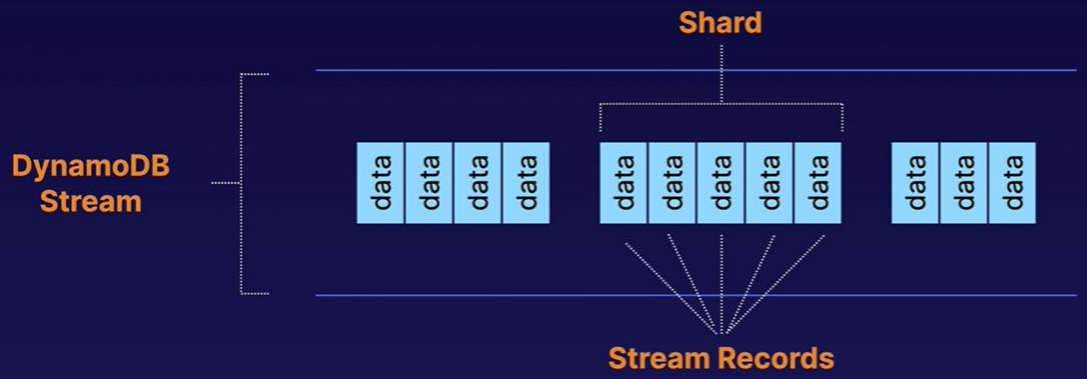
Item

Attribute

DynamoDB Streaming

FIBO CLOUD EDUCATION

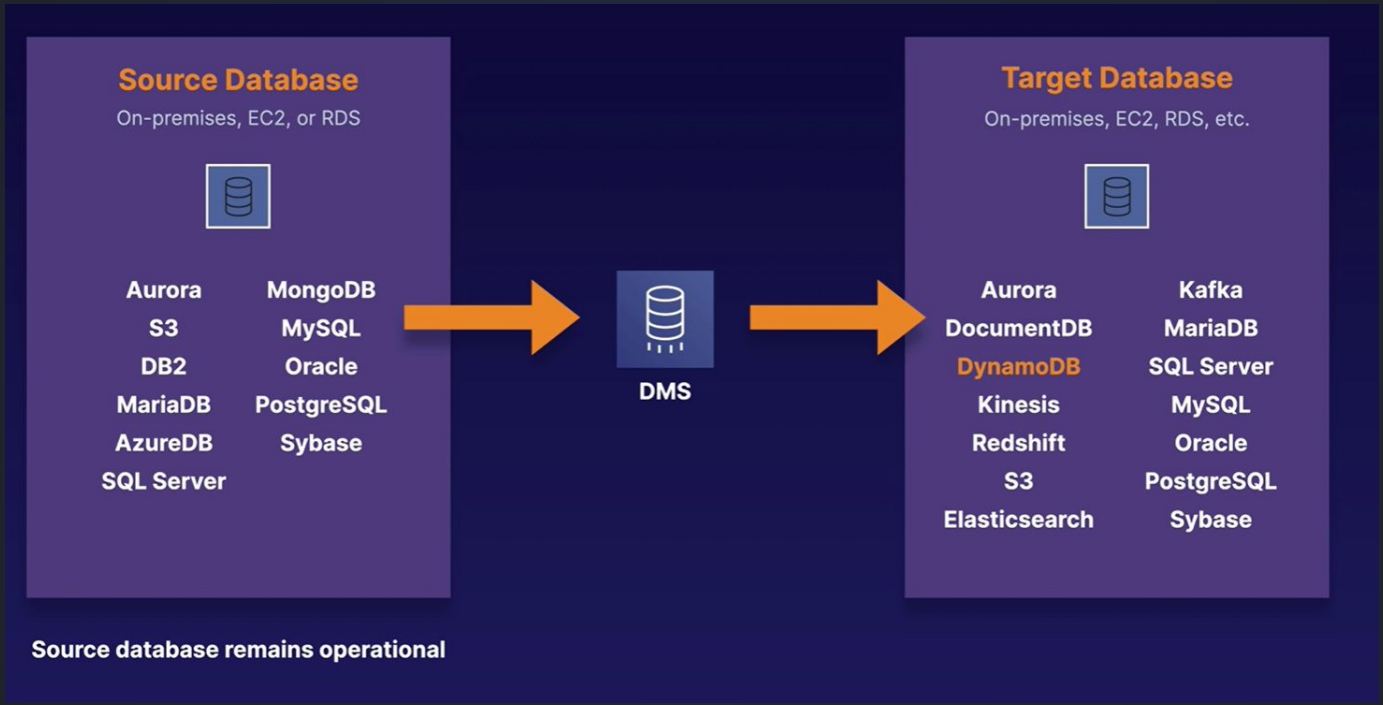
- Time-ordered sequence of item-level changes in a table



- Stored for **24 hours**
- Inserts, updates, and deletes

Database Migration Service

FIBO CLOUD EDUCATION



Features

FIBO CLOUD EDUCATION

- DAX
- On-demand vs Provisioned capacity
- Transaction
  - 25 items or 4MB of data
  - Prepare/Commit
- Full backups at any time
- Point in time recovery
- Global table

To Consider

FIBO
CLOUD
EDUCATION

Lambda X-Ray
Event bridge
Pricing

https://serverless-stack.com/
https://www.serverless.com/

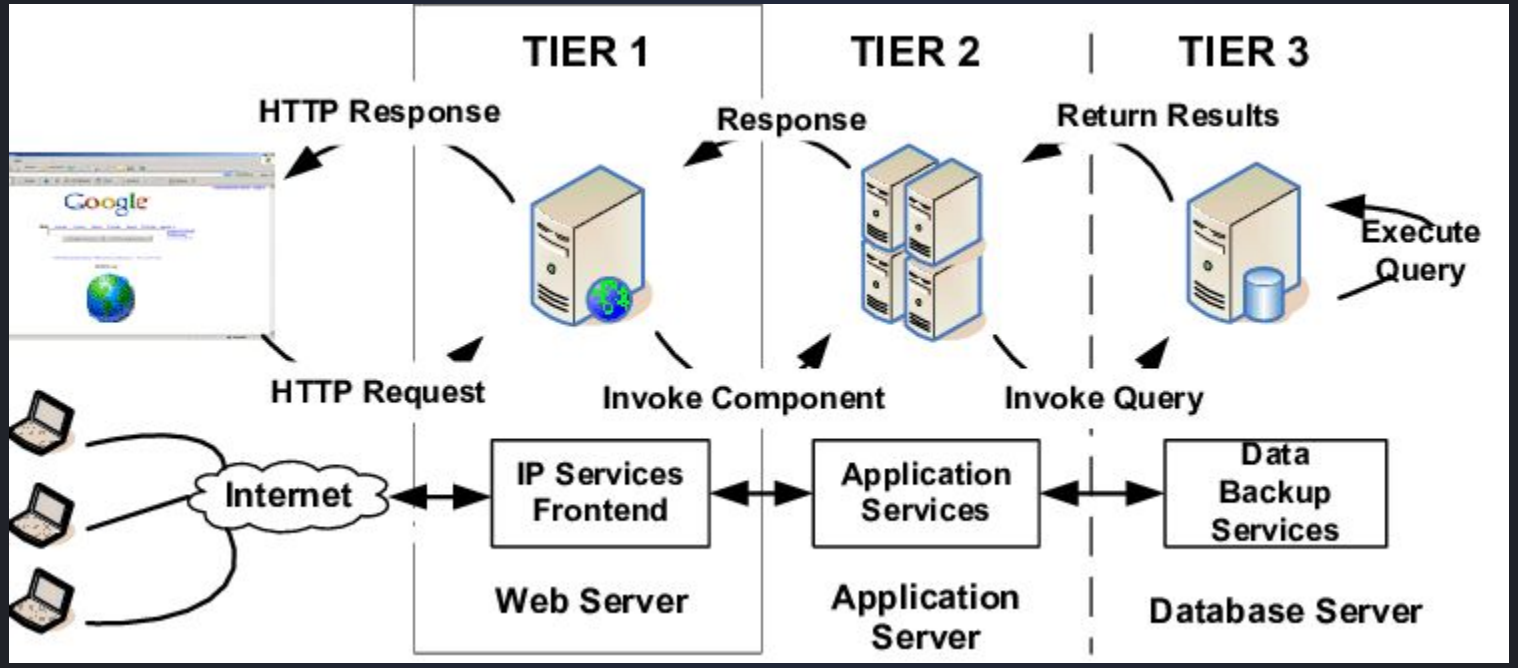
Firebase
AWS Amplify
React, Node

https://gitlab.com/fibo-aws-course/serverless-text-to-speech
https://tutorialsdojo.com/aws-lambda/

FIBO
CLOUD
EDUCATION

Decoupling

Typical architecture

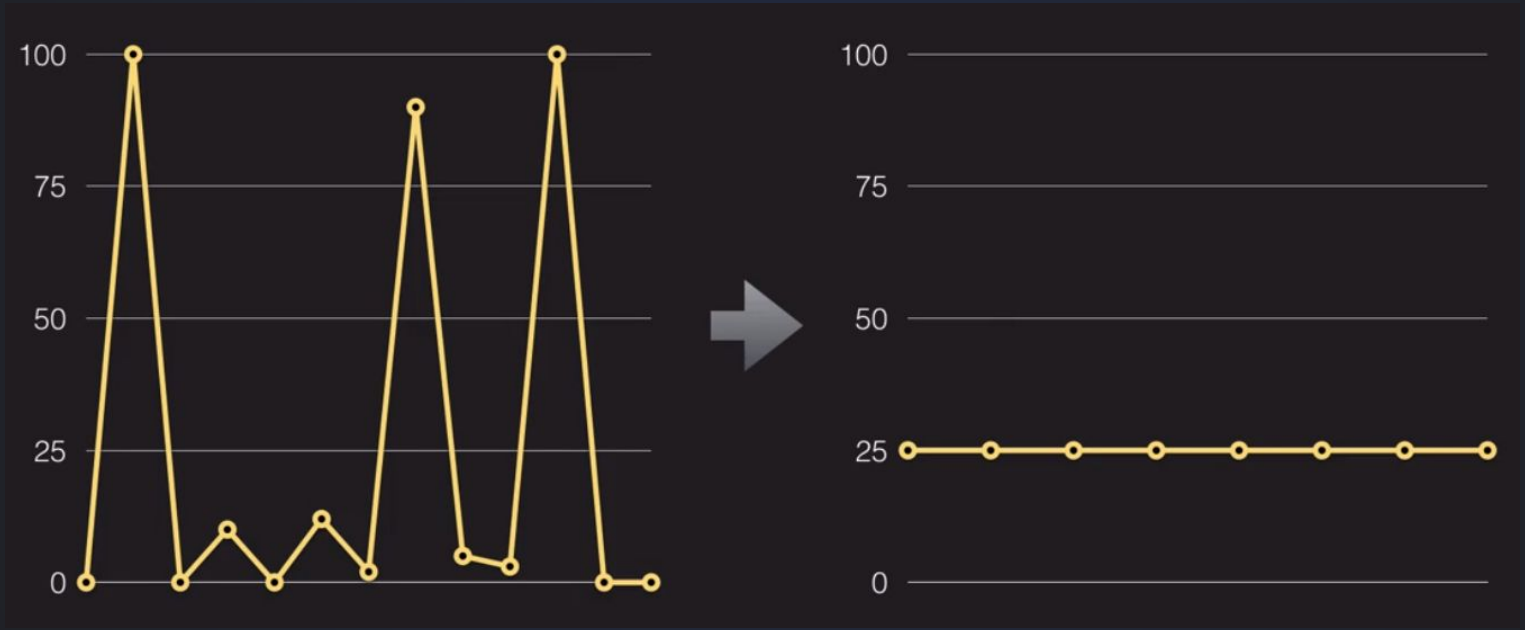


FIBO
CLOUD
EDUCATION

SQS

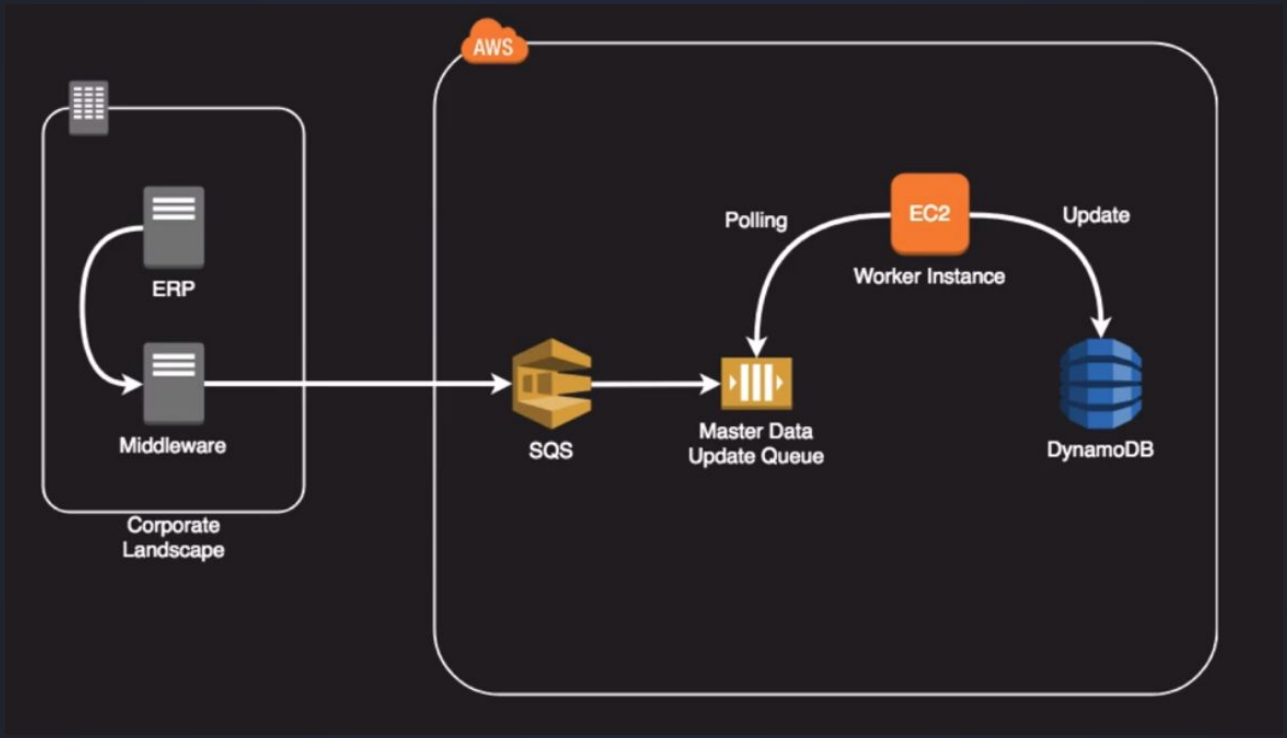
Loosely coupled

FIBO
CLOUD
EDUCATION



Loosely coupled

FIBO CLOUD EDUCATION



What is a Messaging System?

Used to provide asynchronous communication and decouple processes via messages / events
From a sender and receiver (producer and consumer)

Queueing

Generally will delete messages once they are consumed. Simple communication. **Not Real-time**
Have to pull. Not reactive.

VS

Streaming

Multiple consumers can **react** to events (messages), Event live in the stream for long periods of time, so complex operations can be applied. **Real-time**

SQS

Sidekiq

RabbitMQ

Kinesis

APACHE kafka

NATS

Simplified example because there are lots of ifs and buts

SQS is for **Application Integration**.

AWS SQS is a solution for the distributed **queuing** of messages generated by your application. It connects isolate applications together by passing along message to one another.

Application Integration
- Step Functions
- Amazon EventBridge
- Amazon MQ
- Simple Notification Service
- Simple Queue Service
- SWF

A **queue** is a temporary repository for messages that are awaiting processing.

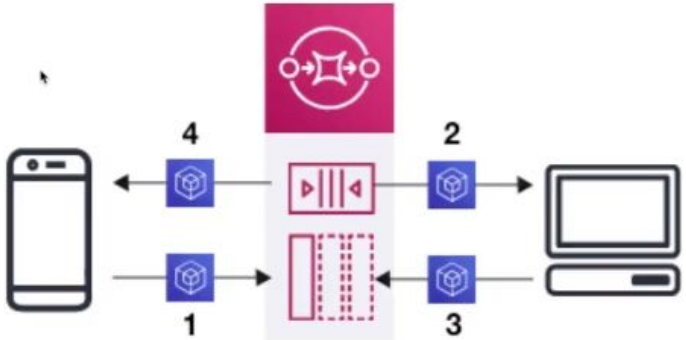
Using the **AWS SDK** you write code which publishes messages onto the queue or you pull the queue for messages.

SQS is pull based
Not push based!

SQS usage

FIBO CLOUD EDUCATION

1. App publishes messages to the queue
2. Other app pulls the queue and find the message and does something
3. Other app reports that they completed their task and marks the message for completion
4. Original app pulls the queue and sees the message is no longer in the queue.



Both apps are using the **AWS SDK** to push messages and pull the queue.

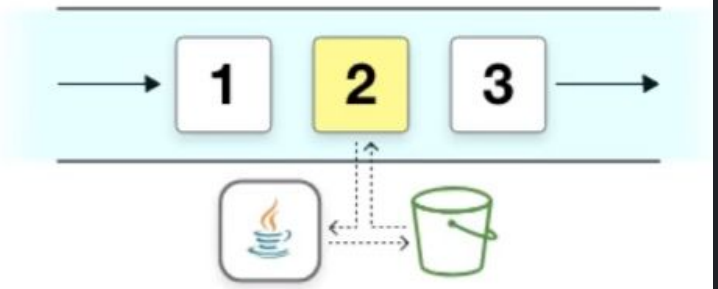
SQS limit - Message size

FIBO CLOUD EDUCATION

Message Size

The size of a message can be between **1 byte and 256 KB**

Amazon SQS Extended Client Library for Java lets you send messages **256KB to 2GB** in size. The message will be stored in S3 and library will reference the S3 object.

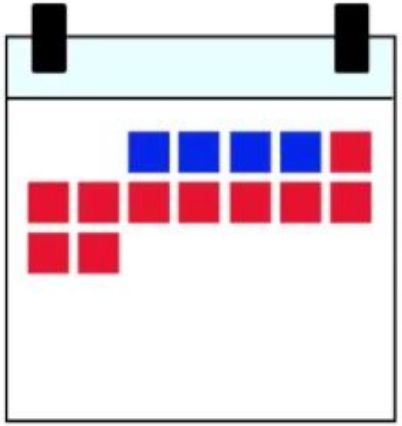


SQS limit - Message retention

FIBO CLOUD EDUCATION

Message Retention

how long SQS will hold onto a message in the queue before dropping it from the queue (deleting it)

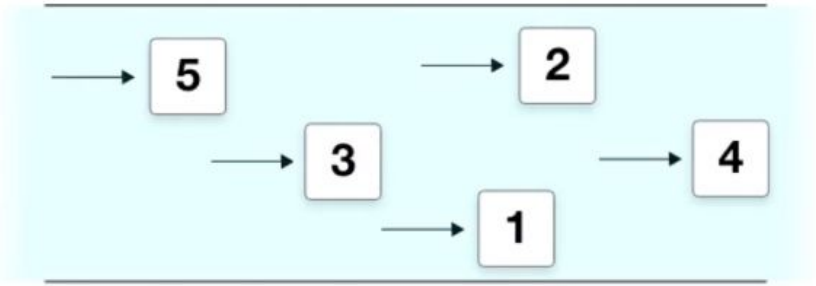


Message retention by default is **4 days**.

Message retention can be adjusted from a minimum of **60 seconds** to a max **of 14 Days**

Standard vs FIFO Queue

FIBO CLOUD EDUCATION



AWS SQS **Standard Queues** allow you a **nearly-unlimited** number of transactions per second.

Guarantees that a message will be delivered **AT LEAST once**.

More than one copy of a message could be potentially delivered **out of order**.

Provides **best-effort ordering** that helps ensure a message is generally delivered in the same order that it was sent.

Standard vs FIFO Queue

FIBO CLOUD EDUCATION

AWS SQS **First-In-First-Out** queues support multiple ordered message groups within a single queue.

Limited to 300 transactions per second.

SWS FIFO queues have all the same capabilities of a Standard Queue

Visibility Timeout

FIBO CLOUD EDUCATION

How do we prevent another app from reading a message while another one Is busy with that message? (Avoid someone doing the same task)

A **visibility time-out** is the period of time that messages are **invisible in the SQS queue**, after a reader picks up that message.

Messages will be **deleted** from the queue **after a job has processed**. (before the visibility timeout expires)

If a job is **NOT** processed before the visibility time-out period, the message will **become visible again** and another reader will process it.

This can result in the same message being delivered **twice**!

30 seconds (default)
0 seconds minimum
12 hours maximum

Short vs Long polling

FIBO CLOUD EDUCATION

Polling is the method in which we retrieve messages from the queues.

Short polling (default) returns messages immediately, even if the message queue being polled is empty.

When you need a message **right away**. shorting polling is what you want to use.

Long polling waits until message **arrives in the queue**, or the **long poll timeout expires**.

Long polling makes it **inexpensive to retrieve messages** from your queue as soon as the messages are available.

Using long polling will reduce the cost because you can **reduce the number of empty receives**.

Most use-cases you want to use Long Polling

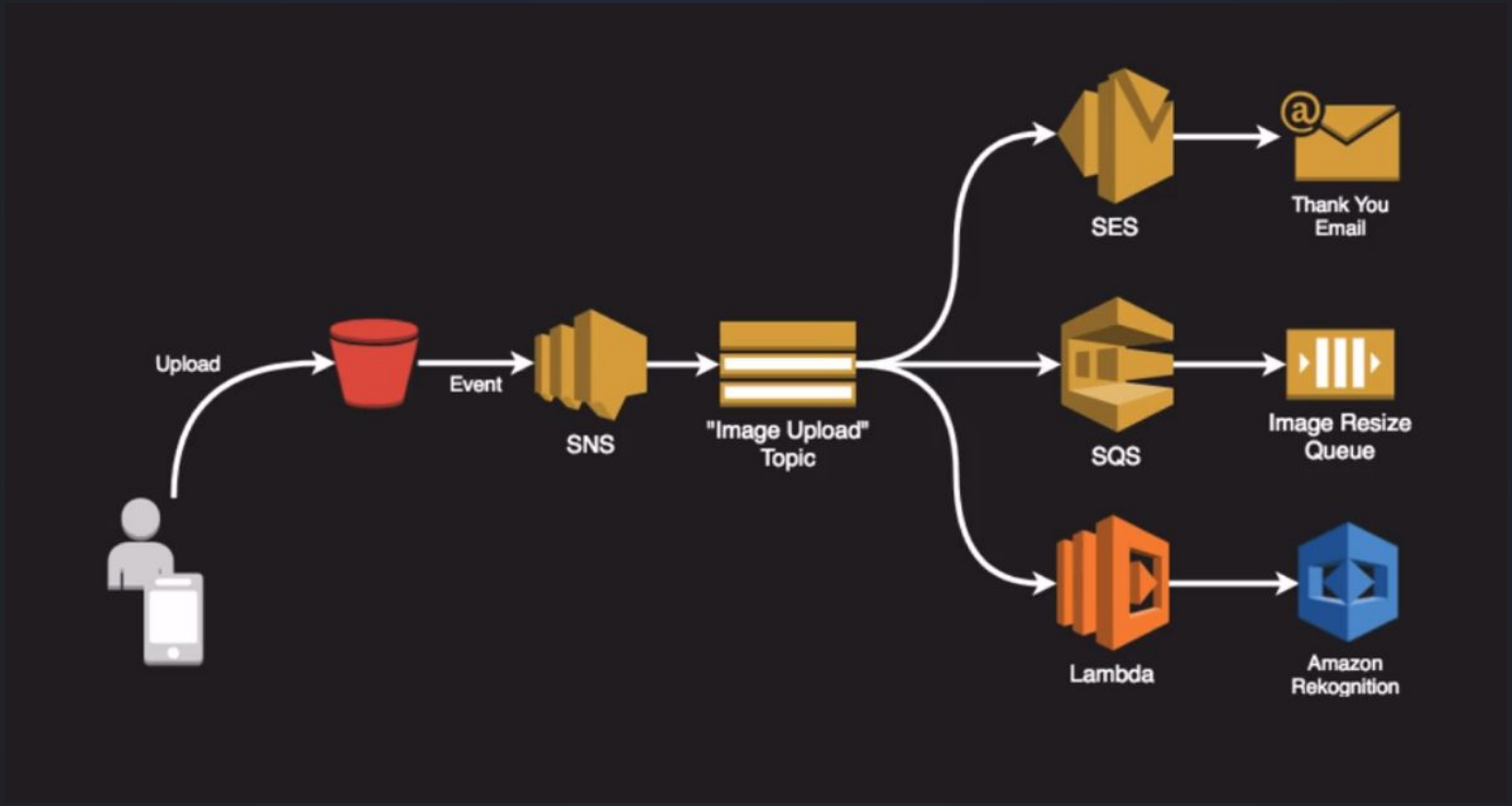
You can enable long polling when receiving a message by setting the wait time in seconds on the **ReceiveMessageRequest**

```
ReceiveMessageRequest receive_request = new ReceiveMessageRequest()
.withQueueUrl(queue_url)
.withWaitTimeSeconds(40);
sqs.receiveMessage(receive_request);
```

FIBO
CLOUD
EDUCATION

SNS

Fan out architecture



Pub/sub

FIBO CLOUD EDUCATION

What is Pub/Sub?

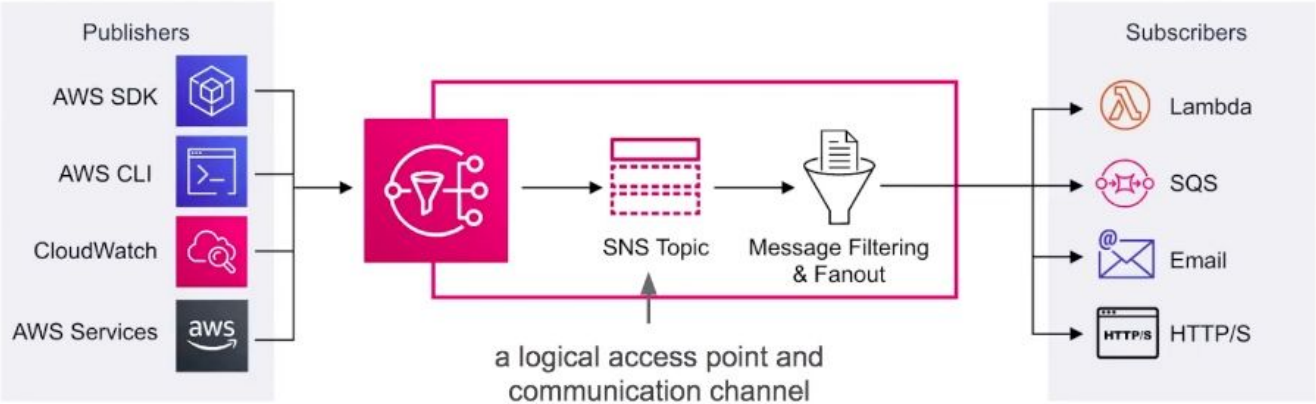
Publish–subscribe pattern commonly implemented in **messaging systems**. In a pub/sub system the sender of messages (**publishers**) do not send their messages directly to receivers. They instead send their messages to an **event bus**. The event bus categorizes their messages into groups. Then receivers of messages (**subscribers**) subscribe to these groups. Whenever new messages appear within their subscription the messages are immediately delivered to them.

- Publisher have no knowledge of who their subscribers are.
- Subscribers do **not pull** for messages.
- Messages are instead automatically and immediately **pushed** to subscribers.
- Messages and events are interchangeable terms in pub/sub

Simple Notification Service (SNS) is a highly available, durable, secure, fully managed **pub/sub messaging** service that enables you to **decouple** microservices, distributed systems, and serverless applications.

Application Integration!

- Publishers **push** events to an SNS Topic
- Subscribers subscribe to SNS Topic to have events **pushed** to them



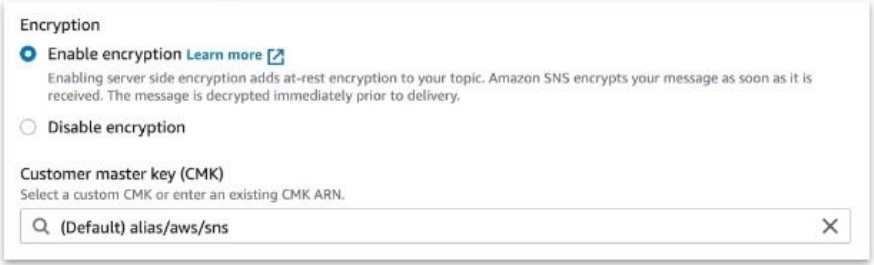
Publishers don't care about the subscribers protocol

Topics allow you to group multiple subscriptions together.

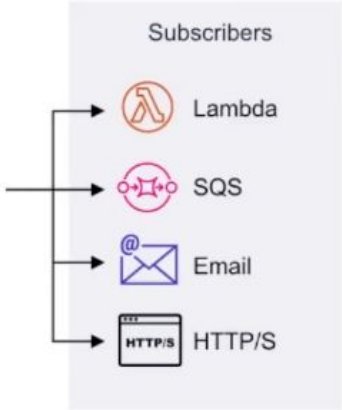
A topic is able to deliver to multiple protocols at once eg. email, text message, http/s

When topics deliver messages to subscribers it will automatically format your message according to the subscriber's chosen protocol

You can encrypt Topics via **KMS**

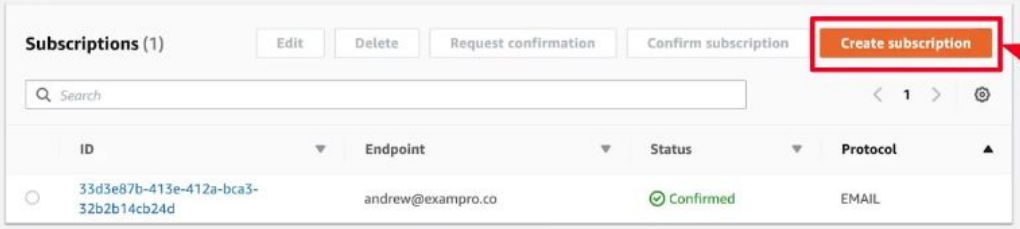


Subscribers listen for incoming messages



Subscriptions

FIBO CLOUD EDUCATION

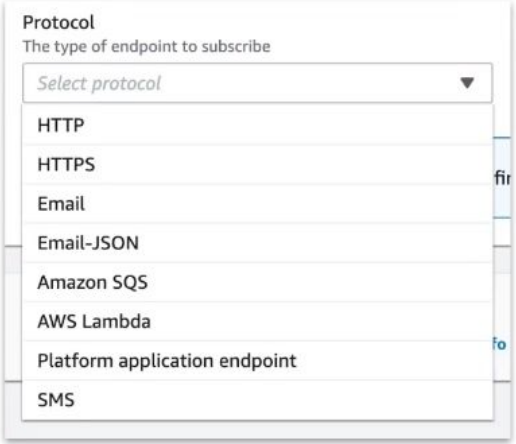


To receive messages from a topic you need to create a Subscription.

A subscription can only subscribe to one protocol and one topic.

The following protocols:

HTTPs and HTTPs create webhooks into your web-application
Email good for internal email notifications (only supports **plain text**)
Email-JSON sends you json via email
Amazon SQS place SNS message into SQS queue
AWS Lambda triggers a lambda function
SMS send a text message
Platform application endpoints Mobile Push

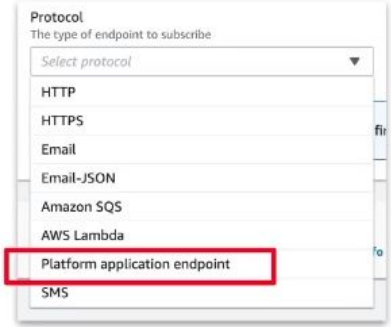


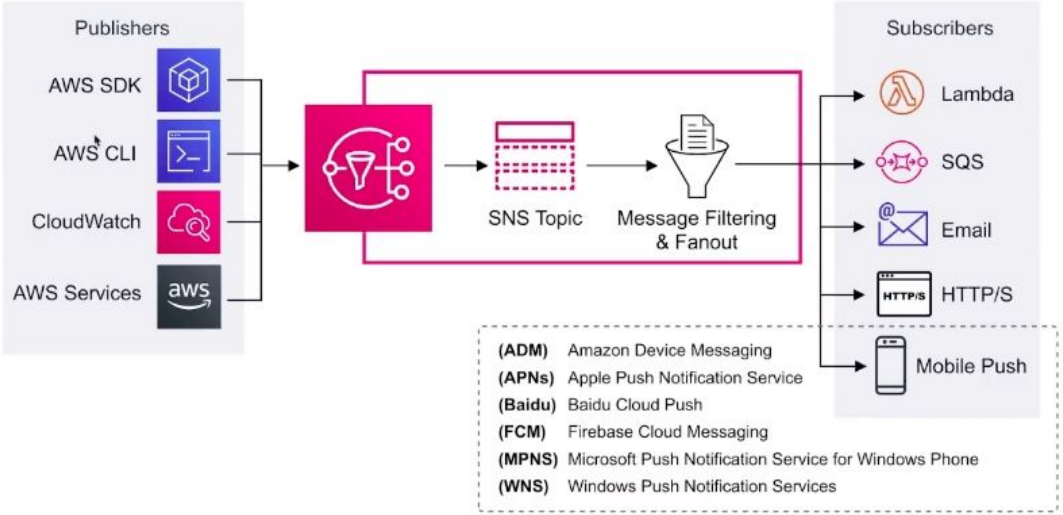
Application as subscribers

FIBO CLOUD EDUCATION

Send push notification messages directly to apps on **mobile devices**.

Push notification messages sent to a mobile endpoint can appear in the mobile app as message alerts, badge updates, or even sound alerts.





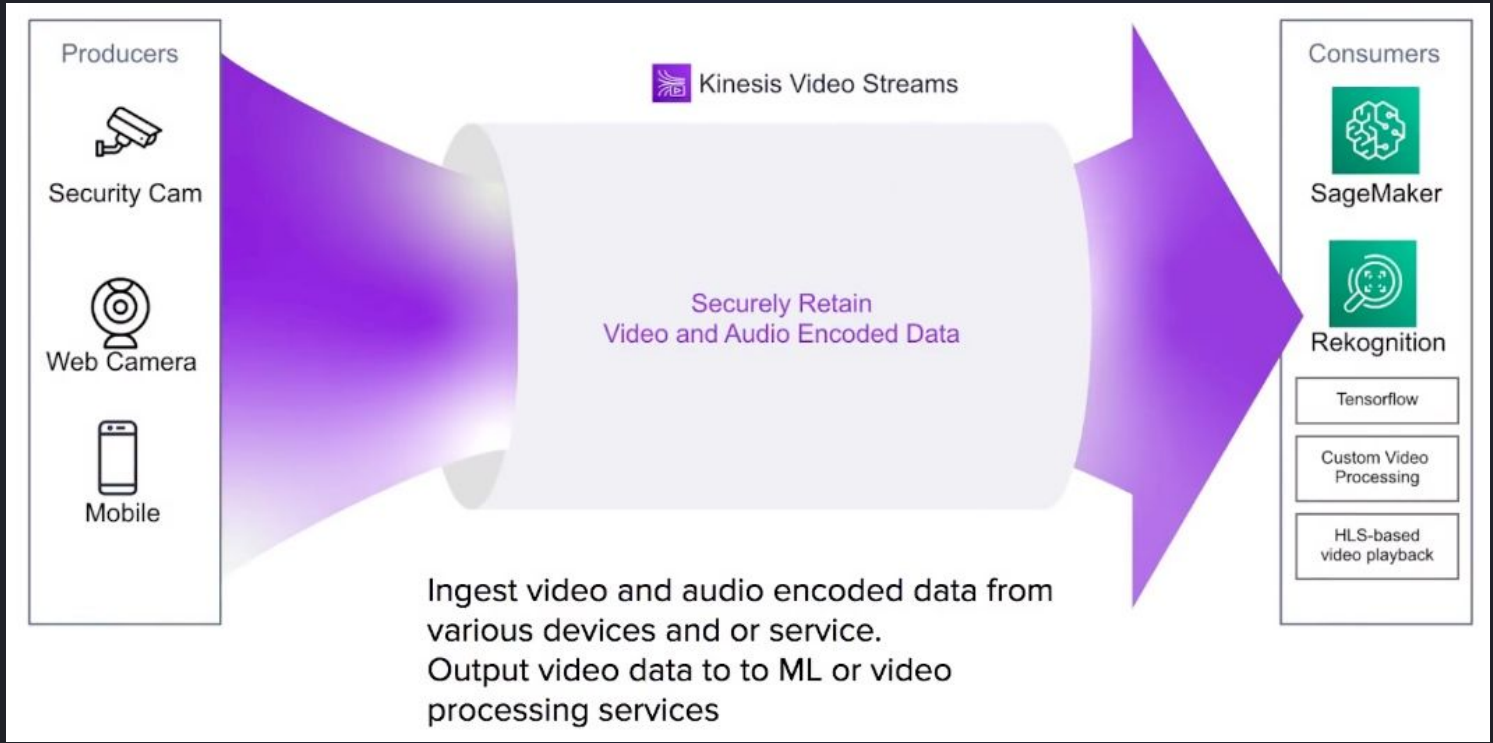
SNS vs SQS?

- Both Messaging Services in AWS
- SNS - Push
- SQS - Polls (Pulls)

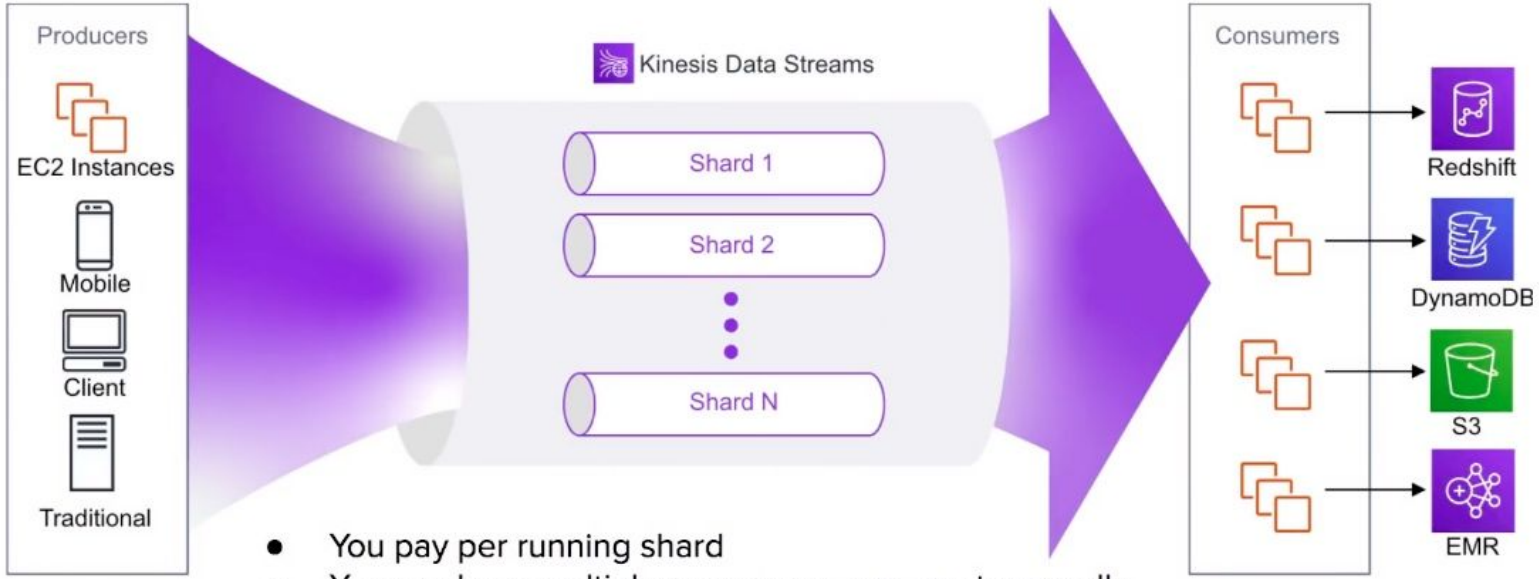
FIBO
CLOUD
EDUCATION

Kinesis

Kinesis Video Streams

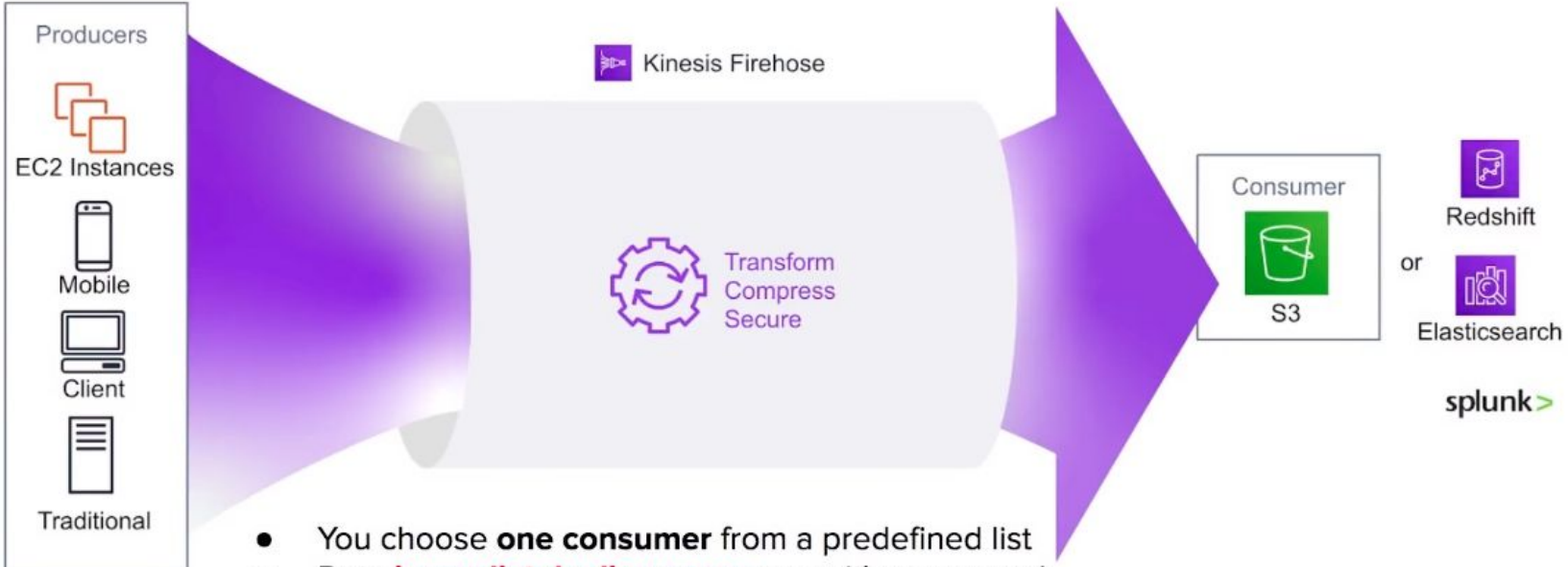


Kinesis Data Streams



- You pay per running shard
- You can have multiple consumers, you must manually configure your consumers.
- Data can be persist from **24 hours (default)** to 168 hours before it disappears from the stream

Kinesis Firehouse

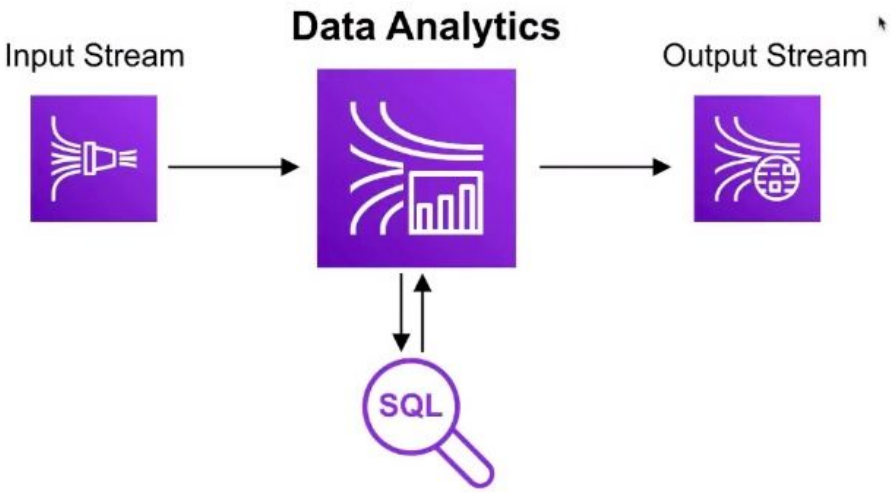


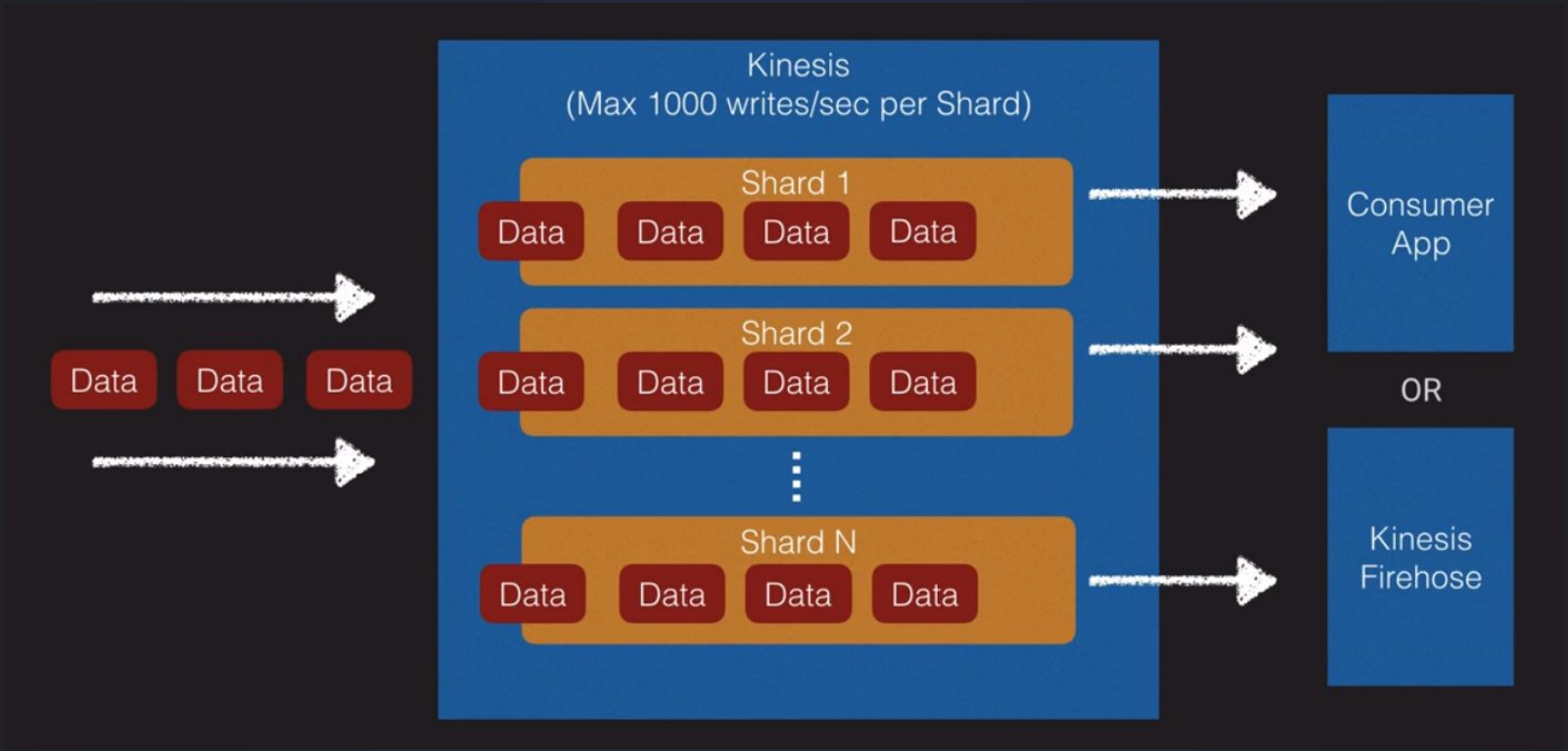
- You choose **one consumer** from a predefined list
- Data **immediately disappears** once it's consumed
- You can convert incoming data to other to a few files formats, compress and secure data.
- You pay only for data that is ingested

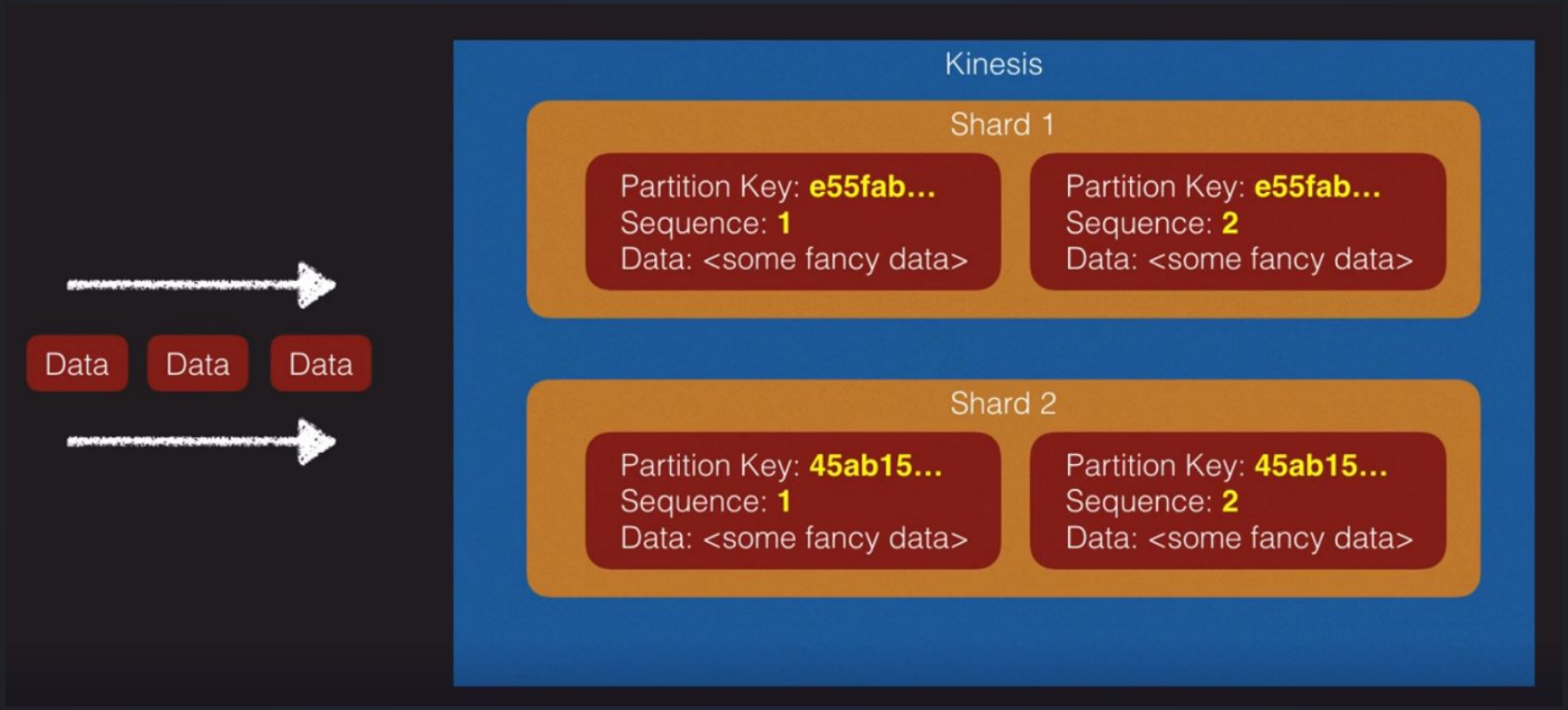
Kinesis Data Analytics

FIBO CLOUD EDUCATION

You can specific Firehose or Data Streams as an input and an output. Data that pass through Data Analytics is run through **custom SQL** you provide and the results are then output. This allows for real-time analytics of your data.

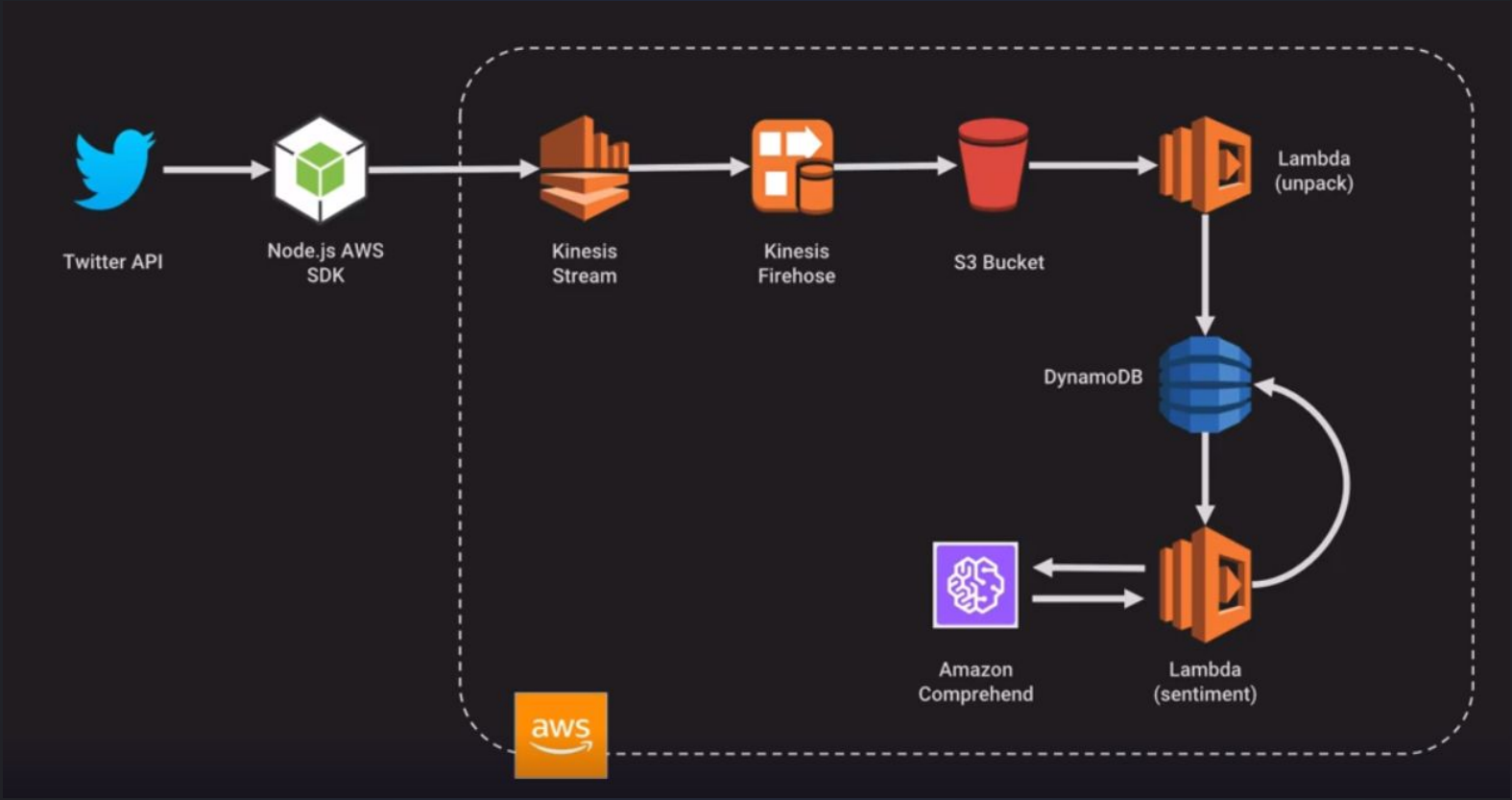



Kinesis
(Max 1000 writes/sec per Shard)
Shard 1
Data
Data
Data
Data
Shard 2
Data
Data
Data
Data
Shard N
Data
Data
Data
Data
Data
Data
Data
Consumer
App
OR
Kinesis
Firehose


Data
Data
Data
Kinesis
Shard 1
Partition Key: e55fab...
Sequence: 1
Data: <some fancy data>
Partition Key: e55fab...
Sequence: 2
Data: <some fancy data>
Shard 2
Partition Key: 45ab15...
Sequence: 1
Data: <some fancy data>
Partition Key: 45ab15...
Sequence: 2
Data: <some fancy data>

Architecture

FIBO CLOUD EDUCATION



Kinesis vs Amazon SQS

FIBO CLOUD EDUCATION

Kinesis Data Streams	Amazon SQS
Routing related records to the same record processor	Messaging semantics such as message-level ack/fail and visibility timeout
Maintaining the order of records	Individual message delay of up to 15 minutes
Connecting multiple consumers to a stream concurrently	Seamless and automatic scalability (Kinesis requires planning and provisioning shards)
Store records for up to 7 days and then consume whilst maintaining order	