



PUBLIC SECTOR
SUMMIT
MEXICO CITY

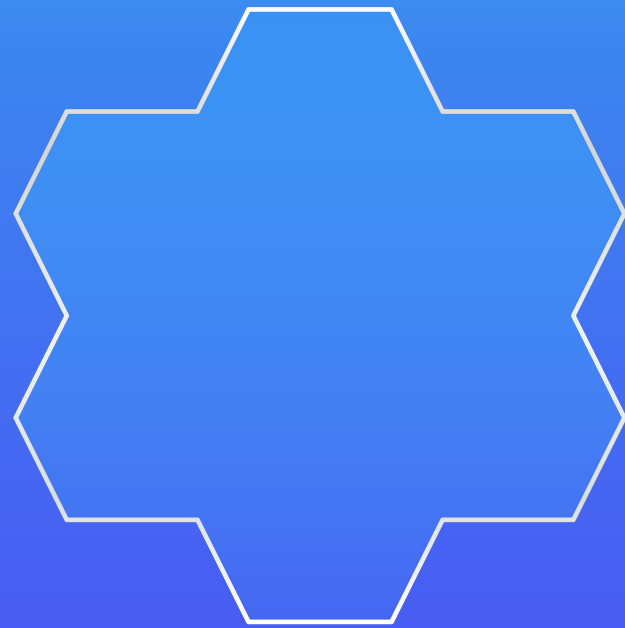
T E C H 2 0 2

El camino hacia aplicaciones sin servidores

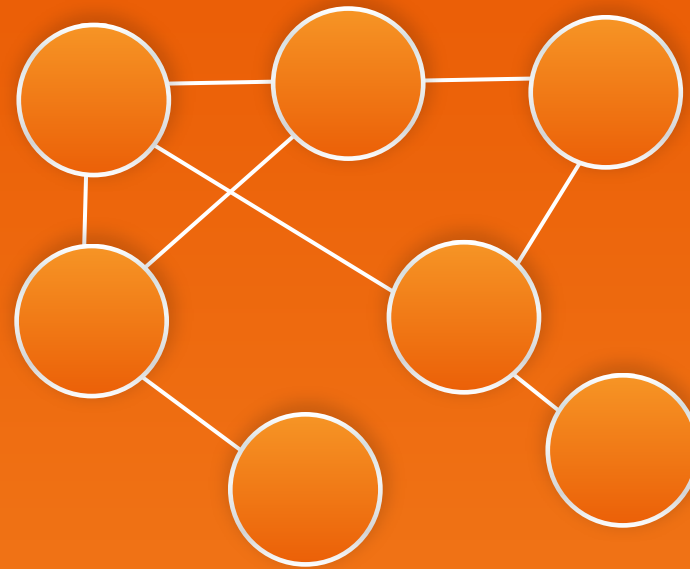
Eduardo Patiño Balaguera
Arquitecto de Soluciones
AWS

¿Por qué estamos aquí?

Evolución de la lógica del negocio



**Aplicación
monolítica**

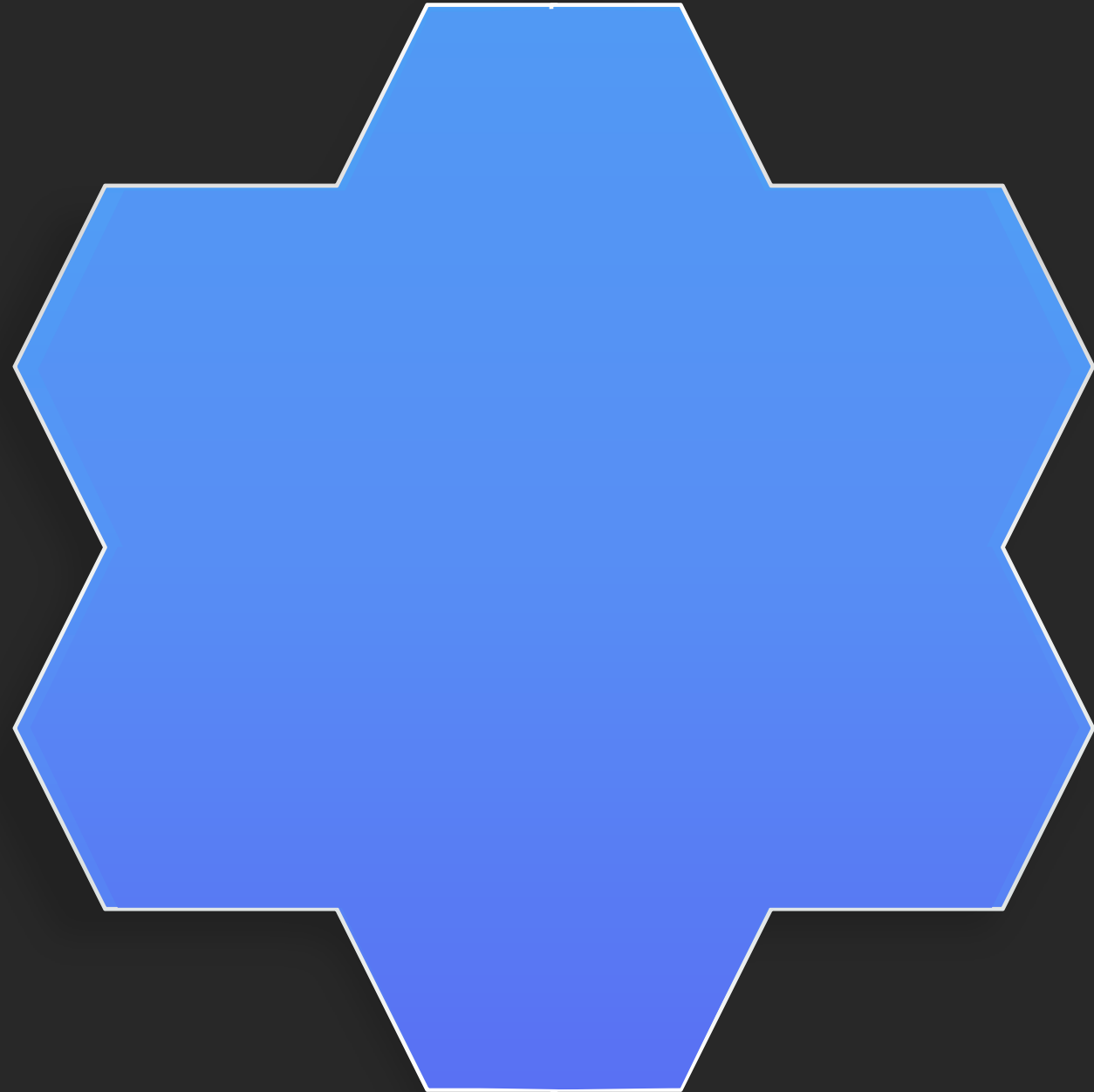


Microservicios



Funciones

**Separación de
apps
monolíticas
hace 10 años**



Separación de apps monolíticas hace 10 años

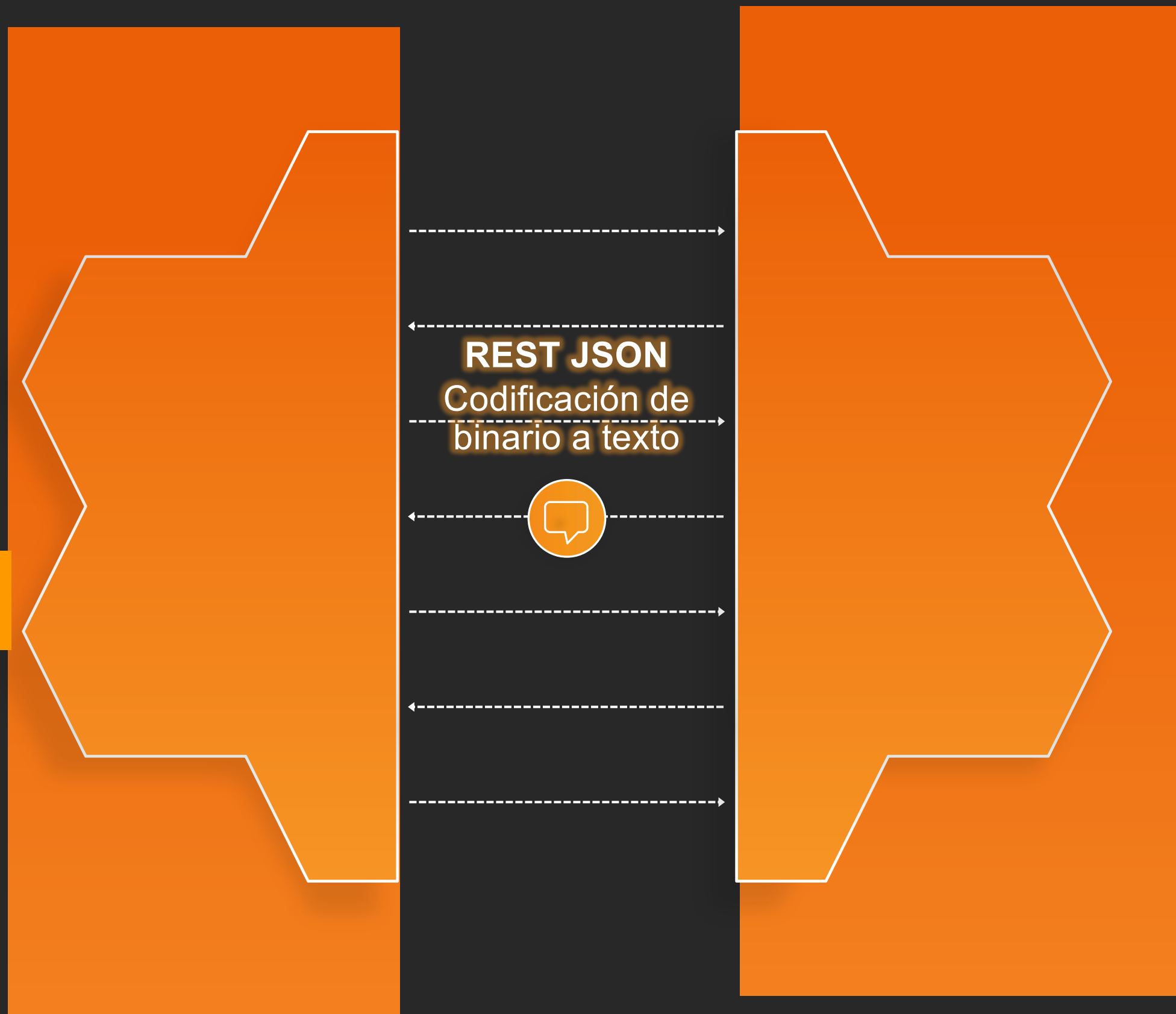
XML & SOAP



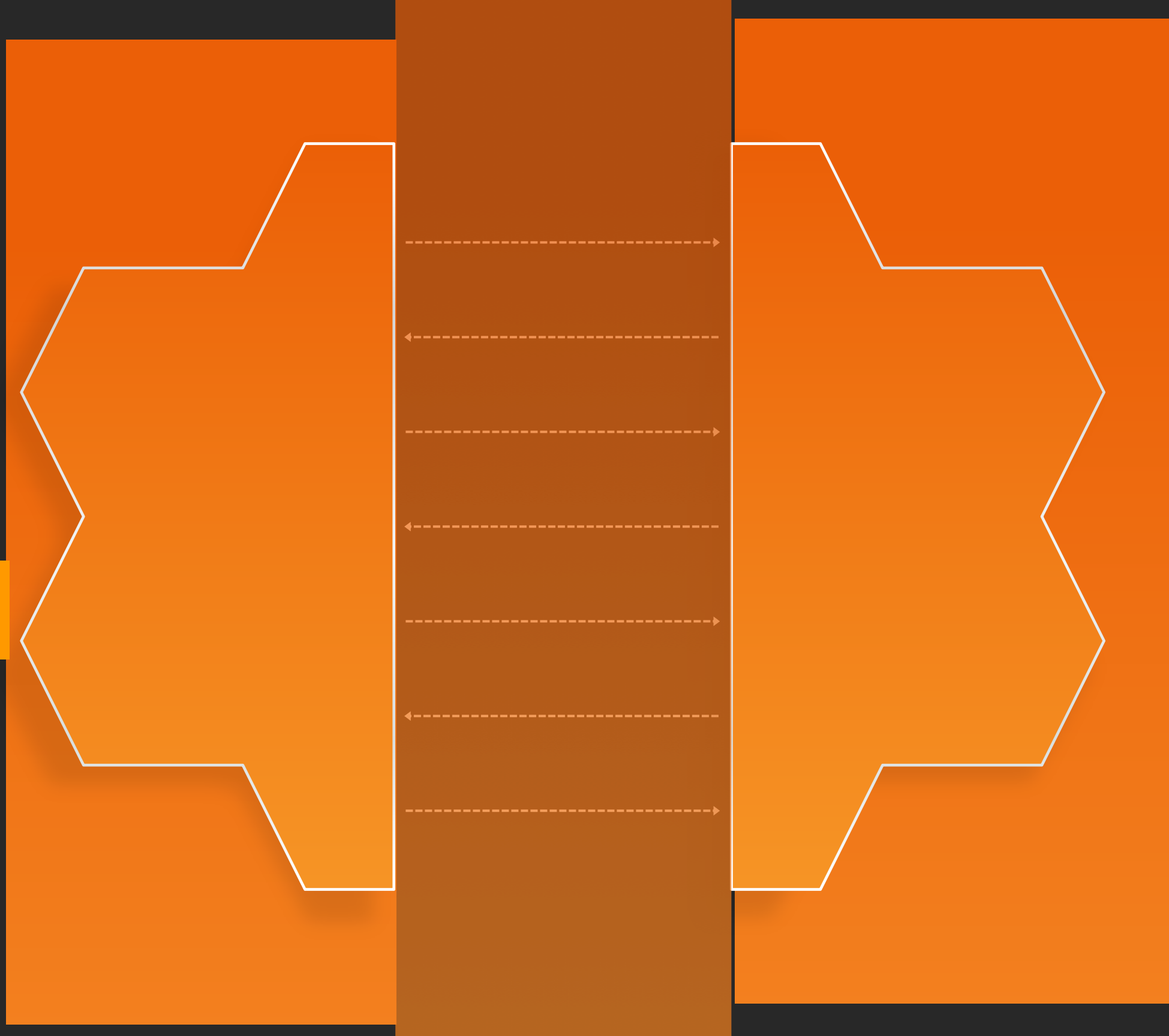
**Separación de
apps monolíticas
hace cinco años**

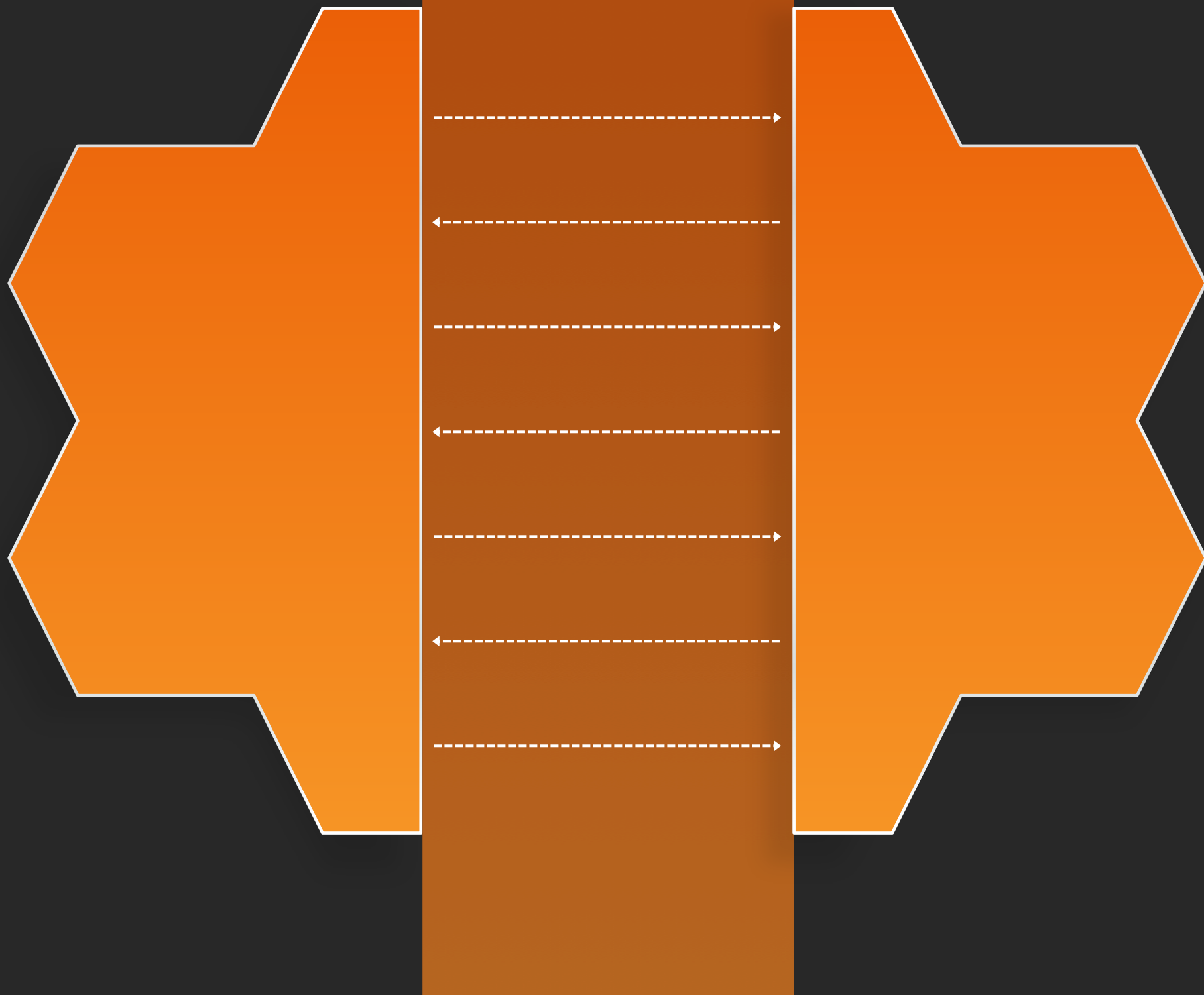


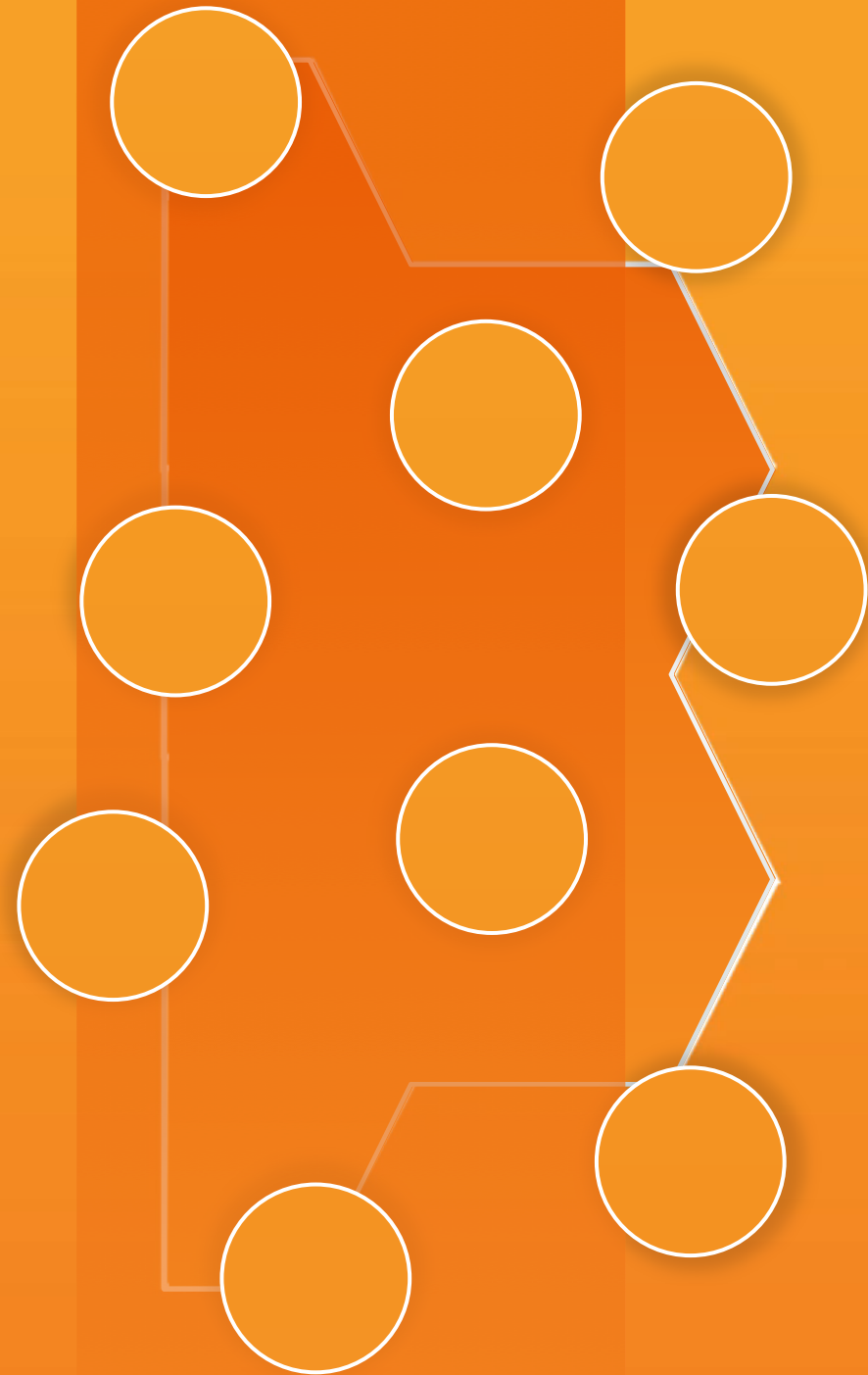
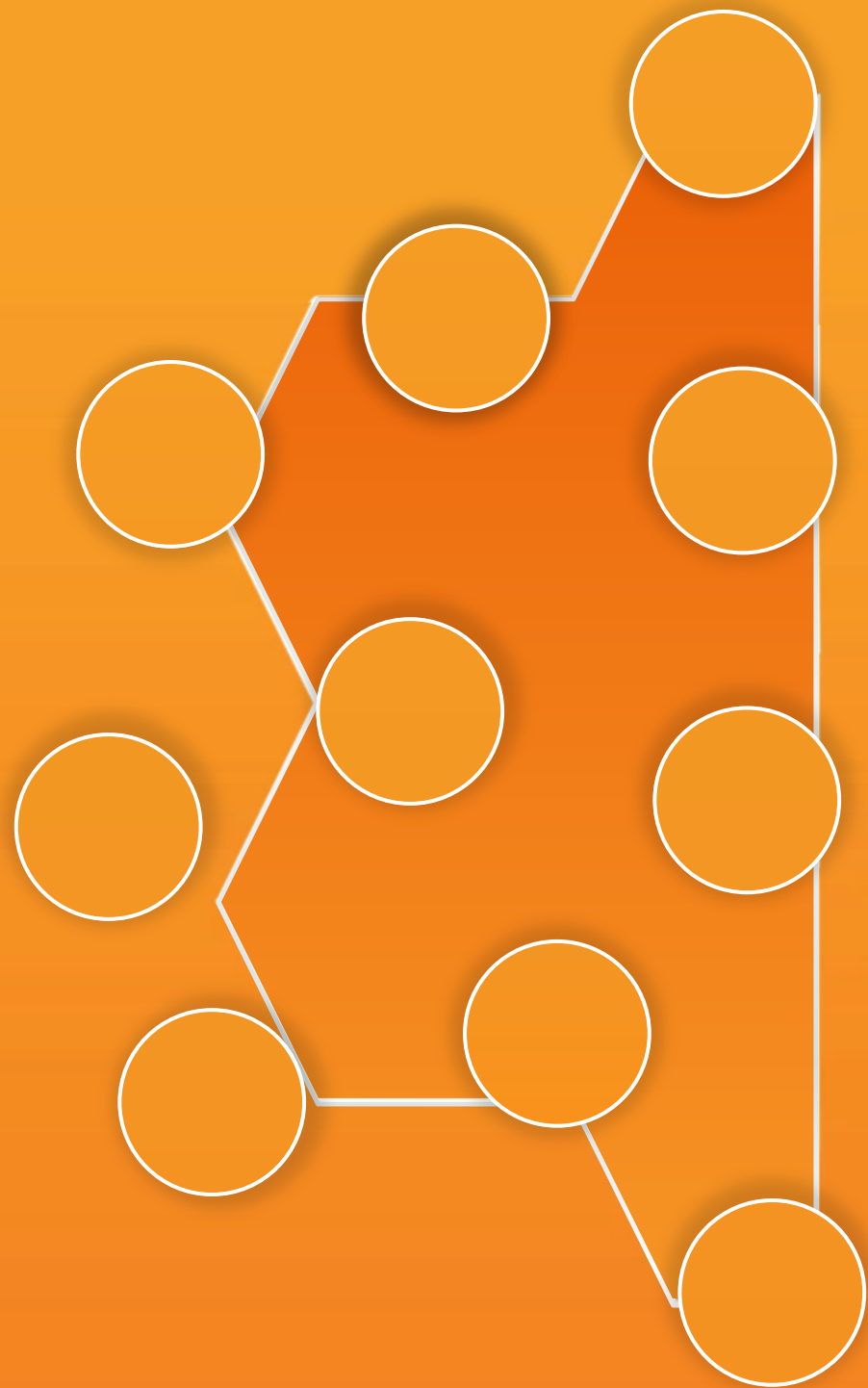
Separación de apps monolíticas cinco años atrás

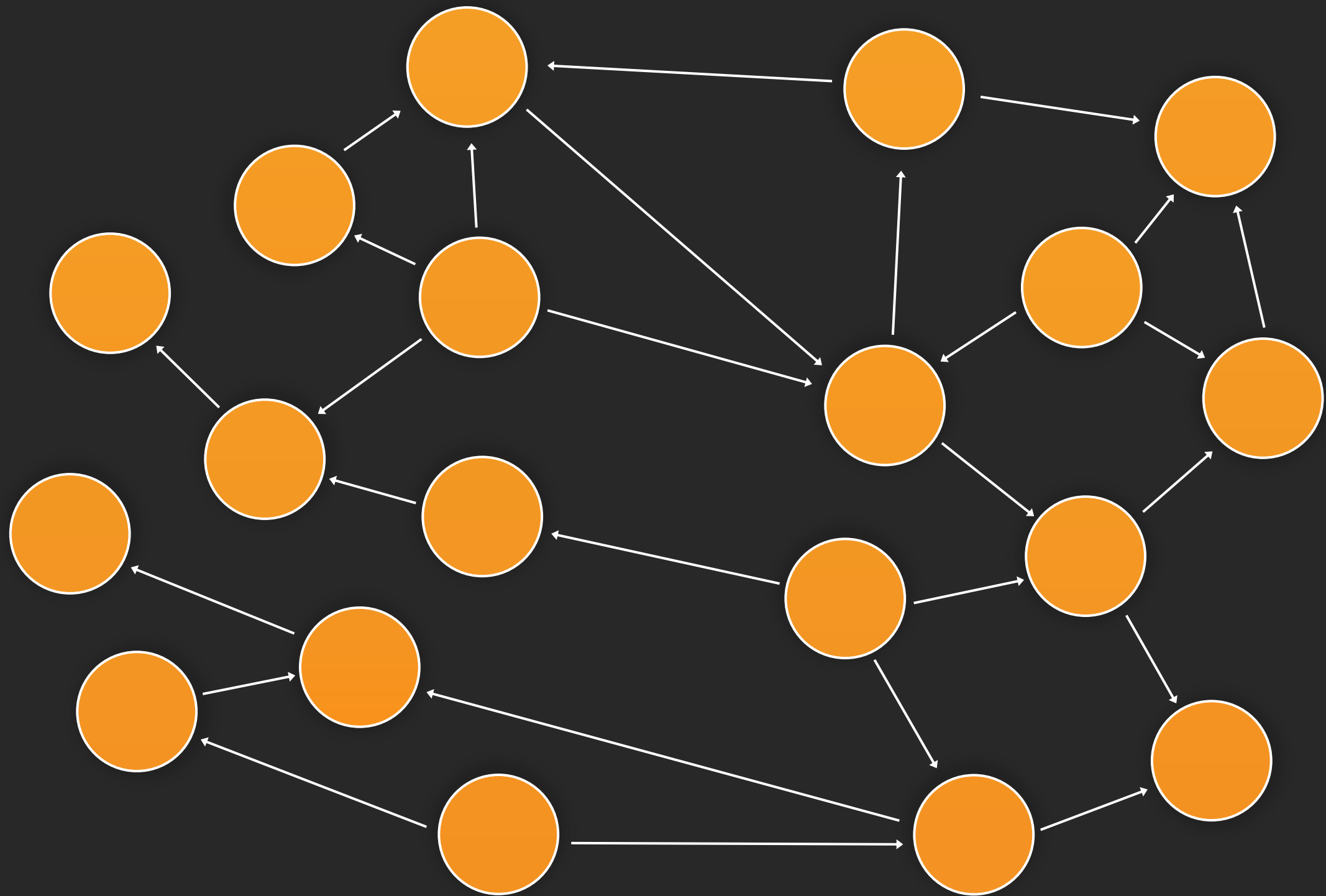


Separación de apps monolíticas cinco años atrás









Microservicios hace cinco años

De Microservicios a funciones hace cinco años

Los servicios estándar
proporcionan capacidades de
plataforma estandarizadas



API Gateway



Amazon
SQS



Amazon
DynamoDB



Amazon S3



Amazon SNS



Amazon
Kinesis

Microservicios a funciones

Los servicios estándar proporcionan capacidades de plataforma estandarizadas



De Microservicios a funciones



De Microservicios a funciones



API Gateway



Amazon S3



Amazon
SQS



Amazon
DynamoDB



Amazon
Kinesis



Amazon SNS

De Microservicios a funciones



API Gateway



Amazon
SQS



Amazon
DynamoDB



Amazon S3



Amazon SNS



Amazon
Kinesis

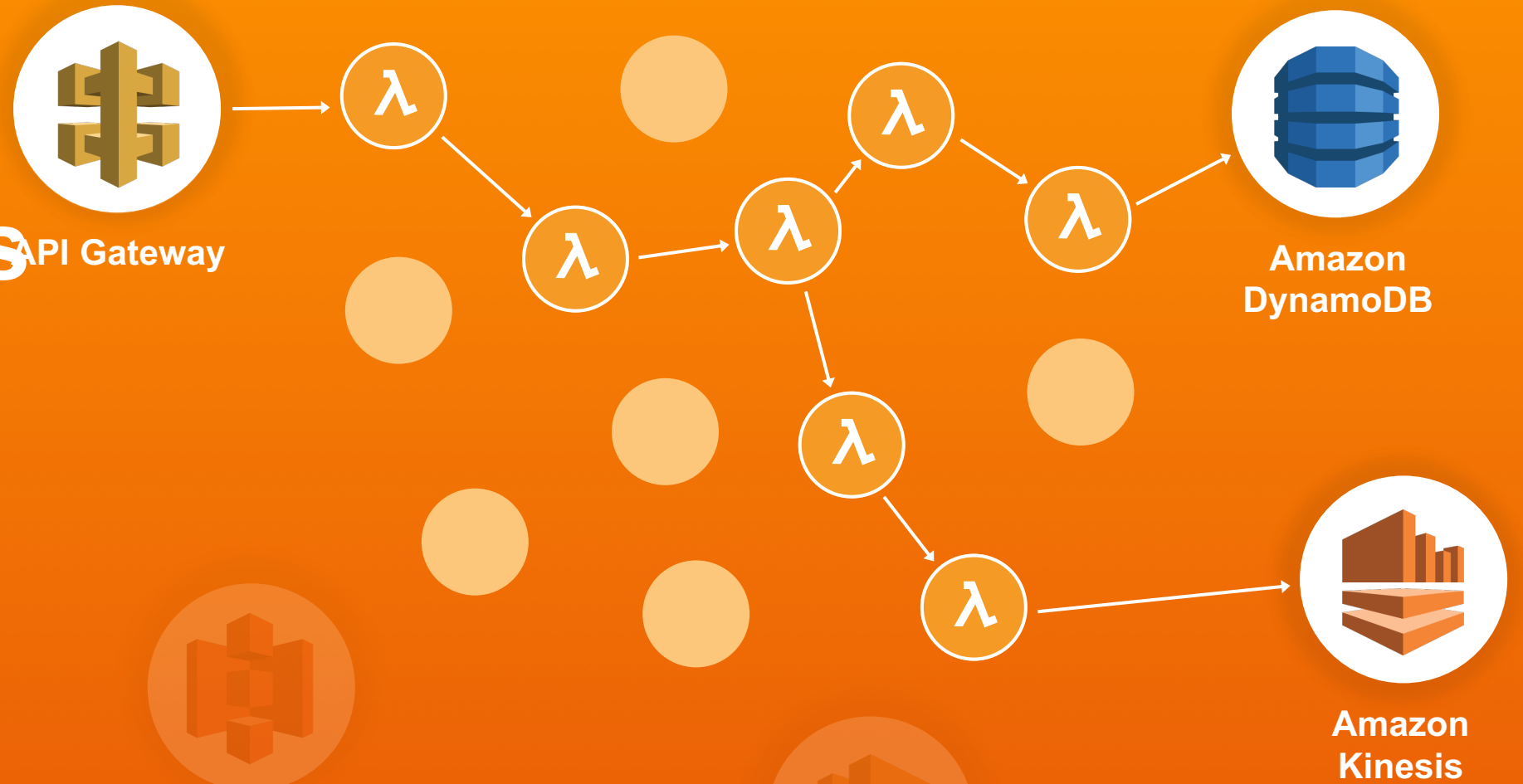
De Microservicios a funciones efímeras



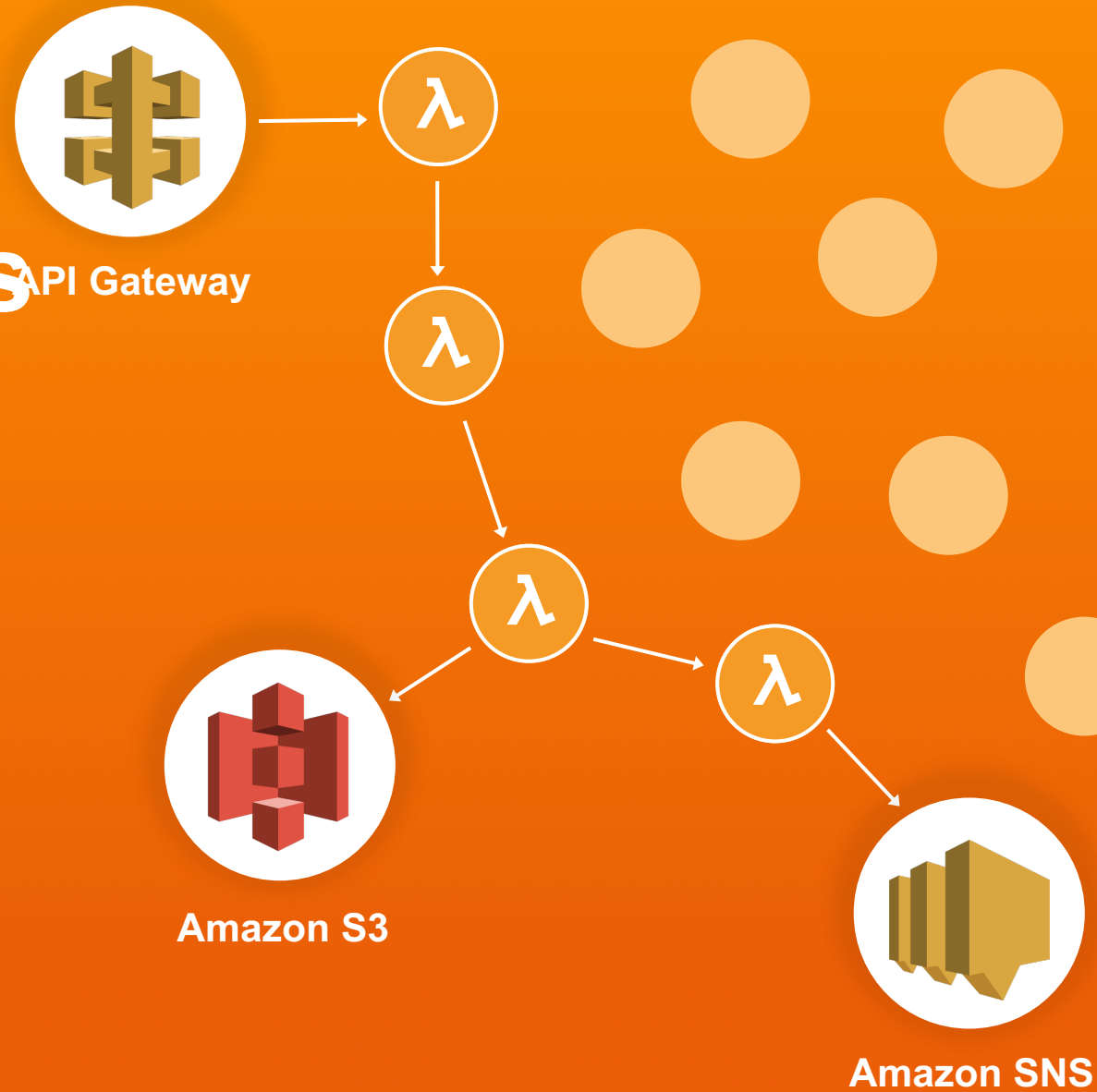
De Microservicios a funciones efímeras



De Microservicios a funciones efímeras



De Microservicios a funciones efímeras



De Microservicios a funciones efímeras

Cuando el sistema
está inactivo, se apaga
y no cuesta ejecutarlo



API Gateway



Amazon
SQS



Amazon
DynamoDB



Amazon S3



Amazon SNS



Amazon
Kinesis

Sin servidores significa...



Ningún servidor que aprovisionar
o gestionar



Nunca pagar por un sistema inactivo



Escala con el uso



Disponibilidad y
tolerancia integradas

Aplicaciones Sin Servidor y arquitecturas orientadas a eventos

Fuente del Evento

Función

Servicios (cualquiera)



Cambios en el estado de los datos



Solicitudes a punto final



Cambios en el estado de los recursos

Node.js

Python

Java

C#

Go

PowerShell

Ruby

El Modelo Sin Servidor es un modelo operativo que abarca muchas categorías de servicios diferentes

COMPUTO



Lambda



AWS
Fargate

ALMACENAMIENTO DE DATOS



Amazon S3



Aurora Serverless



DynamoDB

INTEGRACION



API Gateway



Amazon
SQS



Amazon
SNS



Step Functions



AWS AppSync

Por ahora, enfoquémonos en el cómputo



Lambda

**Ejecución de Código Sin Servidor,
Basado en Evento**

Corta vida

Tiempos de ejecución en todos los idiomas

Integración de fuente de datos

Infraestructura de gestión total



AWS Fargate

**Motor de cómputo Sin Servidor
para contenedores**

Ejecución Larga

Trae código existente

Orquestación de gestión total

Escalado de clústers de gestión total



¿Cómo tomamos la decisión?

Comparación de responsabilidad operativa

AWS gestiona

Cliente gestiona

Más persistente

Lambda
Funciones Sin Servidor

AWS Fargate
Contenedores Sin Servidor

**Amazon ECS/
Amazon EKS**
Contenedor como servicio

Amazon EC2
Infraestructura como Servicio

Menos persistente



<https://www.pexels.com/photo/action-adult-fast-fire-266403/>

Ideas sobre la selección... de contenedores

Si necesitas...

- Latencia más baja al iniciar
- Soporte para trabajos de cómputo de ejecución larga (> 15 minutos)
- Uso predecible y de mucho tráfico
- Persistencia de datos

Si quieres...

- Control total del ambiente de cómputo
 - Pero no solo para fines de control

Ideas sobre la selección... sin servidor

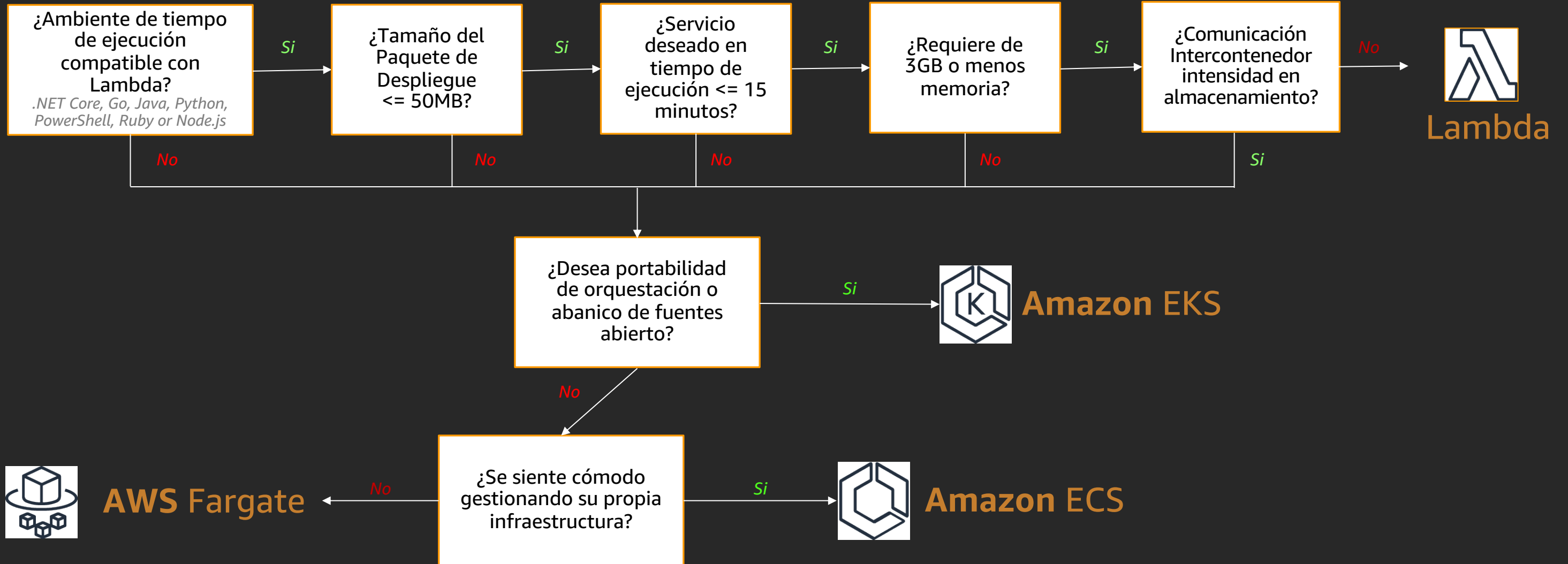
Si necesitas...

- Detonar acción para un evento
- Soporte para utilización variable
- Capacidad de manejar demanda desconocida

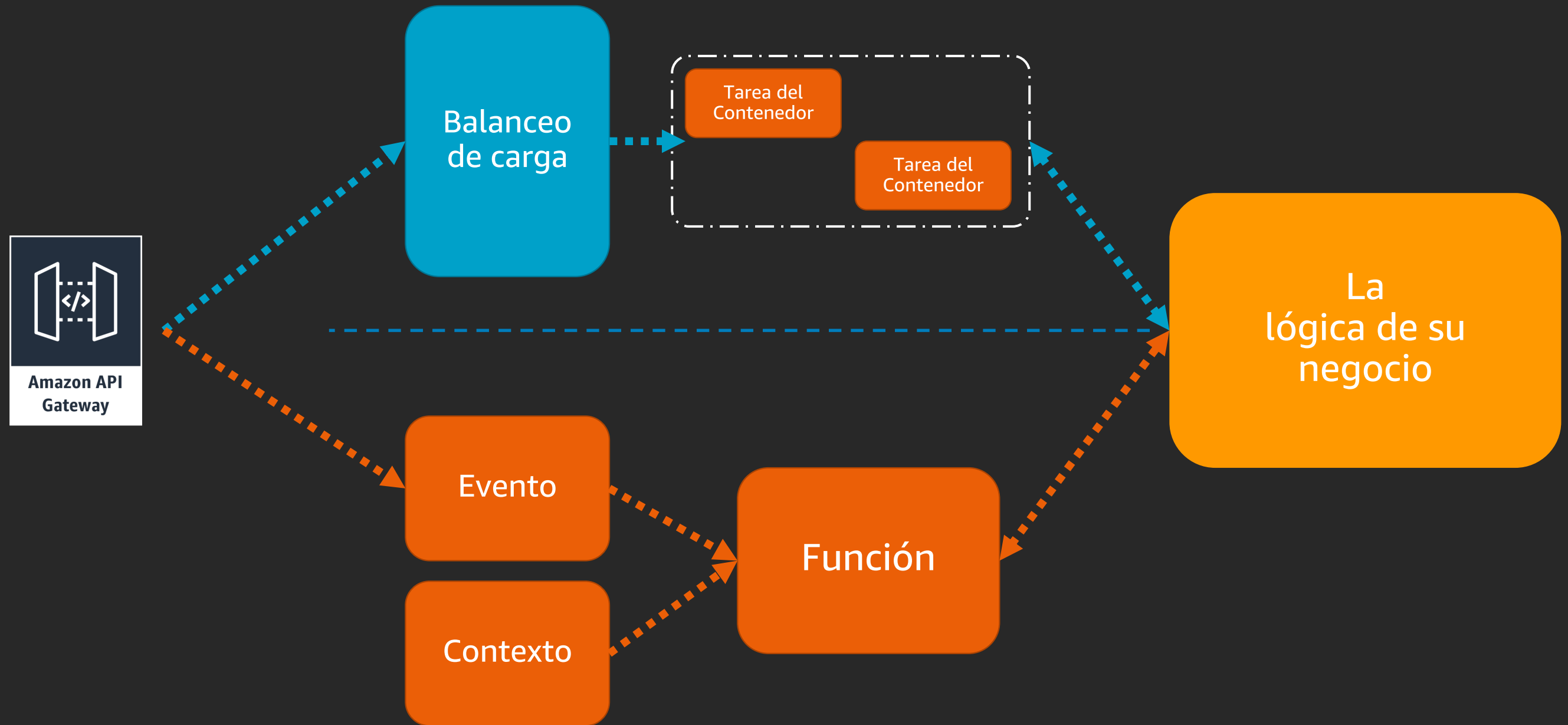
Si quieres...

- Comprobar rápidamente el valor del negocio
- Transferir complejidad operativa (por ejemplo, patching, escalado) a AWS
- Tomar menos decisiones

Árbol de Decisiones

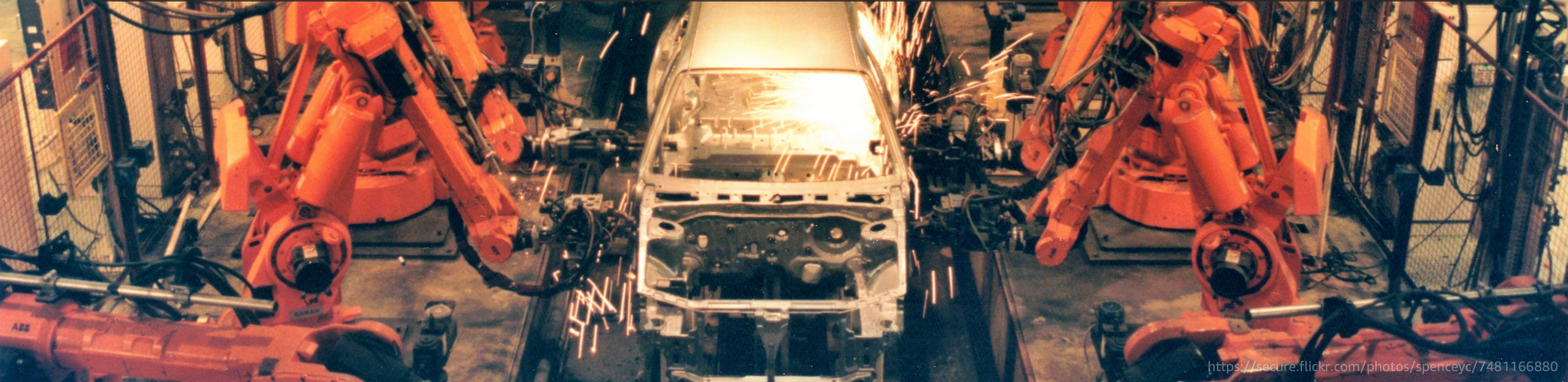


¿Cómo usarlas juntas?

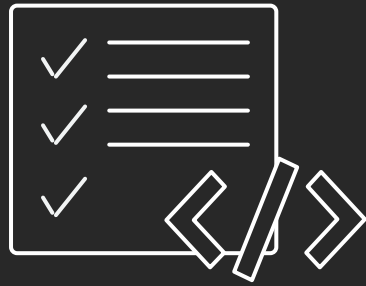




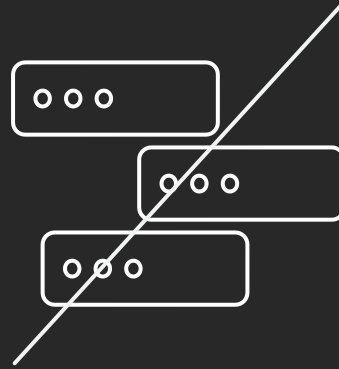
Construya y despliegue su aplicación



Establezca nuestro modelo de prueba/validación

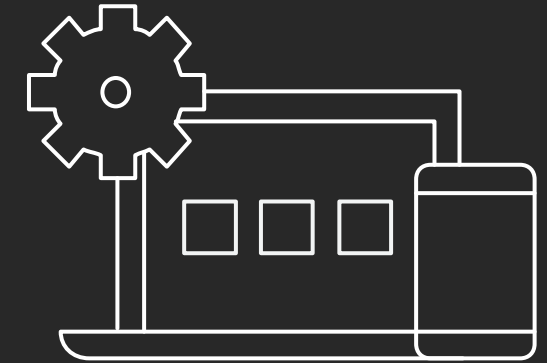


Asegurarnos que nuestro código cumple con estándares y compilados



Nuestras funciones sin servidor se relacionan con otros componentes.

Mecanismos para manejar fallas



Asegurarnos que toda nuestra aplicación/infraestructura sigue:

- Mejores prácticas de seguridad
- Demanda ascendente

Crear y depurar aplicaciones Lambda en AWS usando su IDE favorito



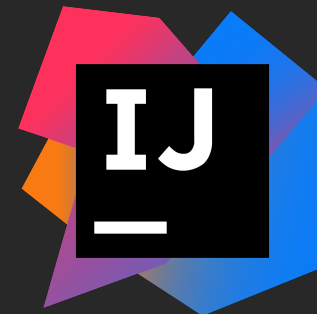
AWS
Cloud9

Python, Node



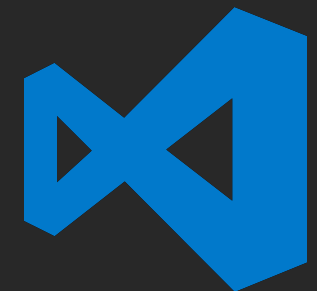
AWS Toolkit
para PyCharm

Python



AWS Toolkit
para IntelliJ

Java, Python



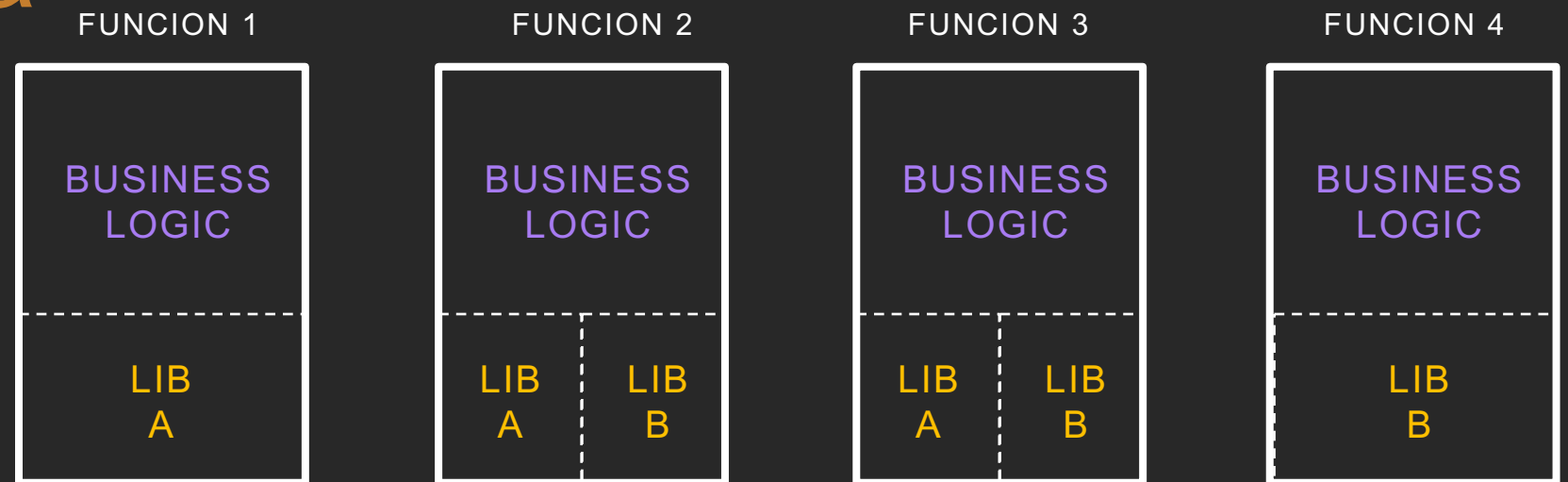
AWS Toolkit
para Visual Studio
Code

.NET, Node

Capas de Lambda

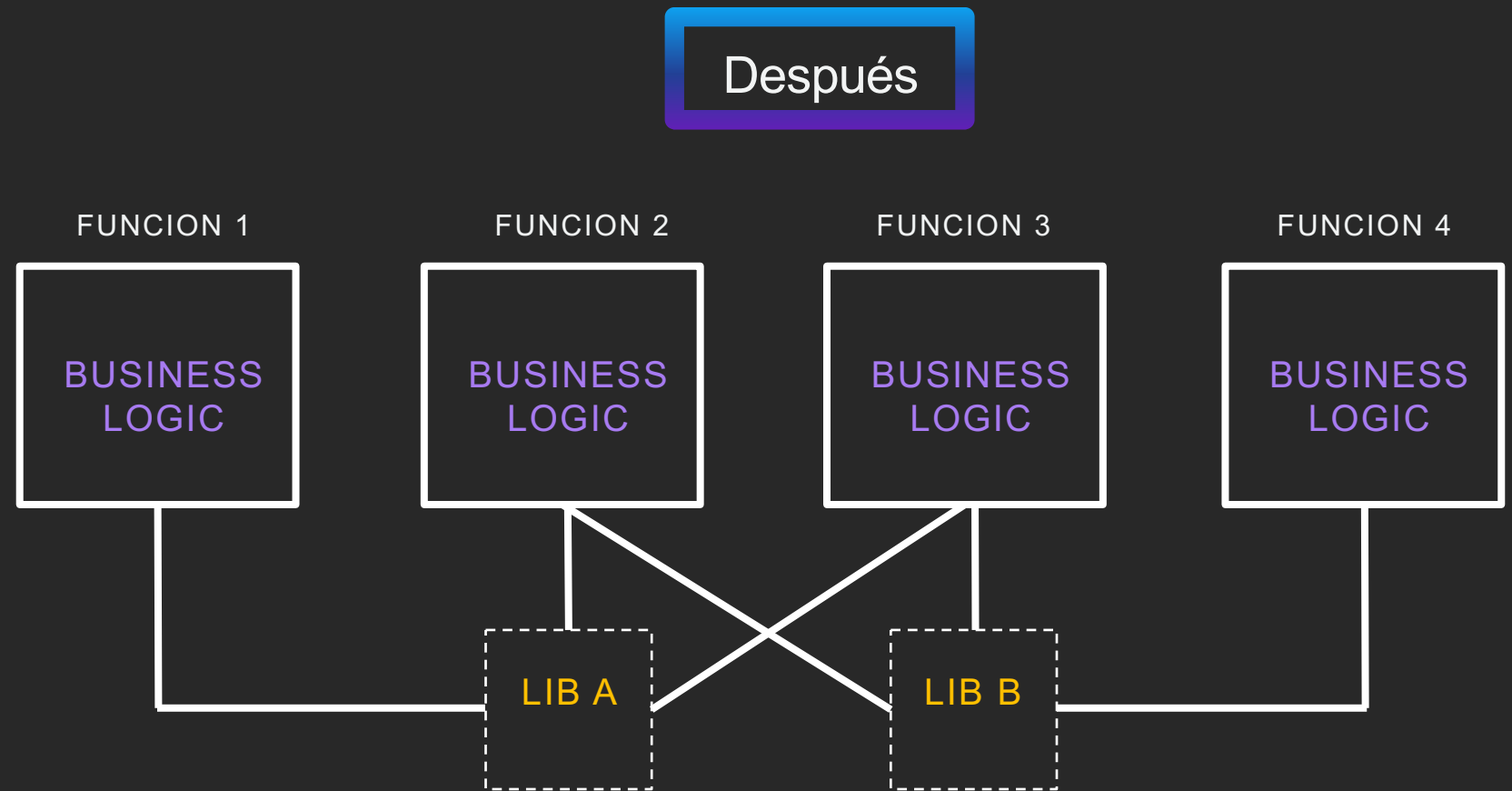
Extienda el ambiente de ejecución de Lambda con cualquier binario, dependencia o tiempo de ejecución

Antes



Capas de Lambda

Extienda el ambiente de ejecución de Lambda con cualquier binario, dependencia o tiempo de ejecución



Herramientas de prueba



Inspección de Código / cobertura de prueba:

Landscape

<https://landscape.io/>
(solo para Python)

CodeClimate

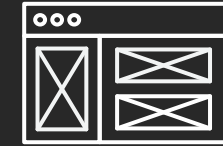
<https://codeclimate.com/>

Coveralls.io

<https://coveralls.io/>

LocalStack spins up the following core Cloud APIs on your local machine:

- API Gateway at <http://localhost:4567>
- Kinesis at <http://localhost:4568>
- DynamoDB at <http://localhost:4569>
- DynamoDB Streams at <http://localhost:4570>
- Amazon ES at <http://localhost:4571>
- Amazon S3 at <http://localhost:4572>
- Kinesis data firehose at <http://localhost:45713>
- Lambda at <http://localhost:4574>
- Amazon SNS at <http://localhost:4575>
- Amazon SQS at <http://localhost:4576>
- Amazon Redshift at <http://localhost:4577>
- Amazon SES at <http://localhost:4579>
- Route 53 at <http://localhost:4580>
- AWS CloudFormation at <http://localhost:4581>
- CloudWatch at <http://localhost:4582>
- SSM at <http://localhost:4583>
- Secrets Manager at <http://localhost:4584>
- Step Functions at <http://localhost:4585>
- CloudWatch Logs at <http://localhost:4586>
- AWS STS at <http://localhost:4592>
- IAM at <http://localhost:4593>



Interfaz API / UI prueba:

Runscope

<https://www.runscope.com/>
API Monitoreo/Prueba

Ghost Inspector

<https://ghostinspector.com/>
Prueba de interfaz web



No se puede avanzar si no se puede medir
lo qué sucede

Medir y loggear son un derecho universal

Métricas CloudWatch



7 métricas integradas para Lambda

Conteo de invocaciones
Duración de la invocación
Errores de invocación
Intentos limitados
Iterator Age
Errores DLQ
Concurrencia

Puede elegir “[put-metric-data](#)” desde su código de función para personalizar las métricas

7 métricas integradas para API Gateway

Conteo de llamadas a API
Latencia
4XXs
5XXs
Latencia de Integración
Conteo aciertos en Cache
Conteo errores en Cache

Las métricas de error y caché soportan [promedios](#) y [percentiles](#)

Accesso a loggeo API Gateway

Settings

Logs

Stage Variables

SDK Generation

Export

Deployment History

Documentation History

Configure the metering and caching settings for the stage.

CloudWatch Settings

Enable CloudWatch Logs

Enable Detailed CloudWatch Metrics

Custom Access Logging

Enable Access Logging

CloudWatch Group

Log Format

Insert Example:

CLFJSONXMLCSV

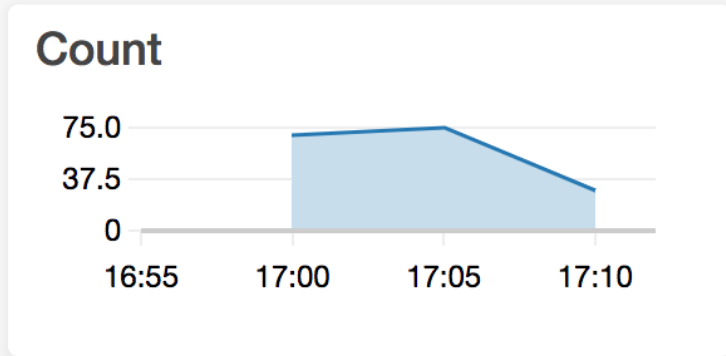
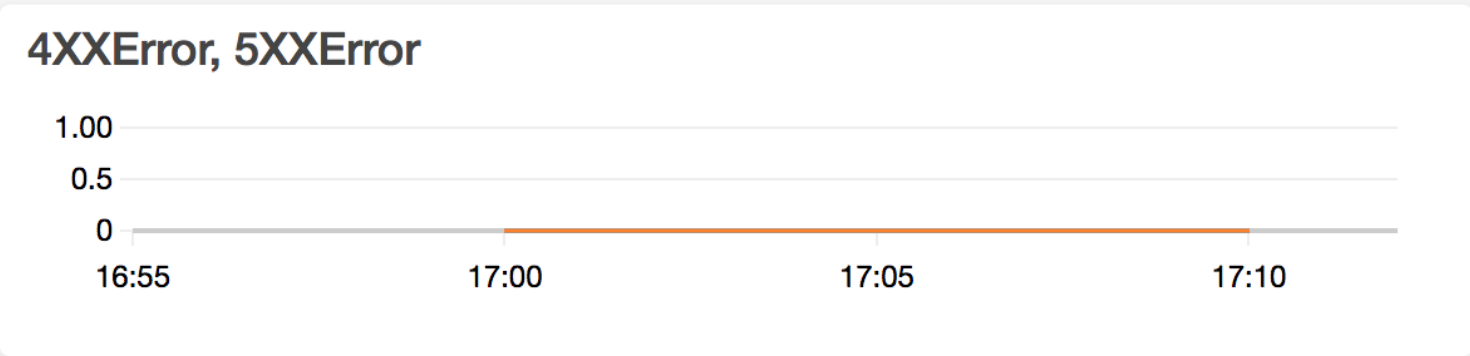
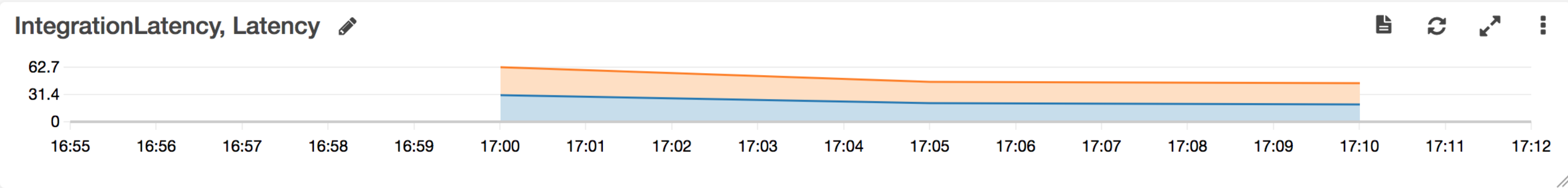
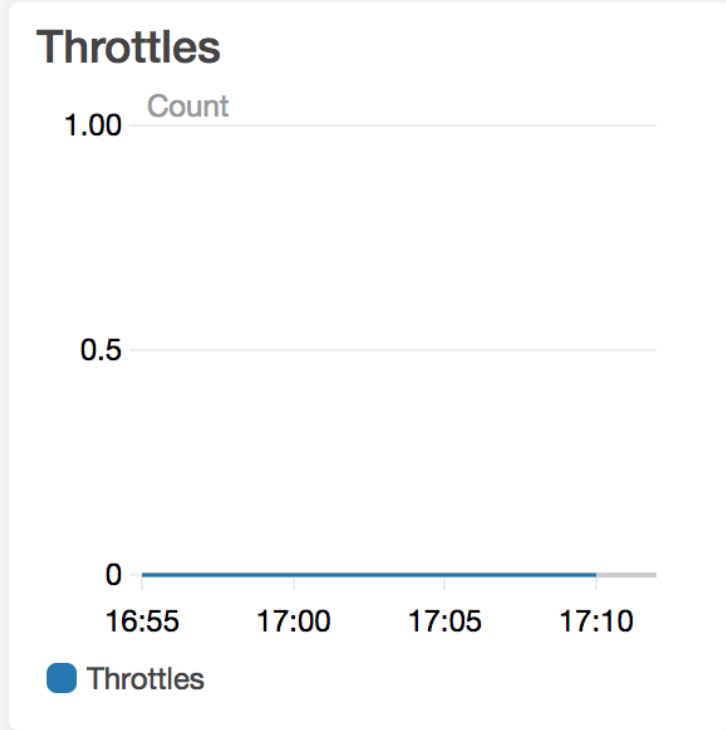
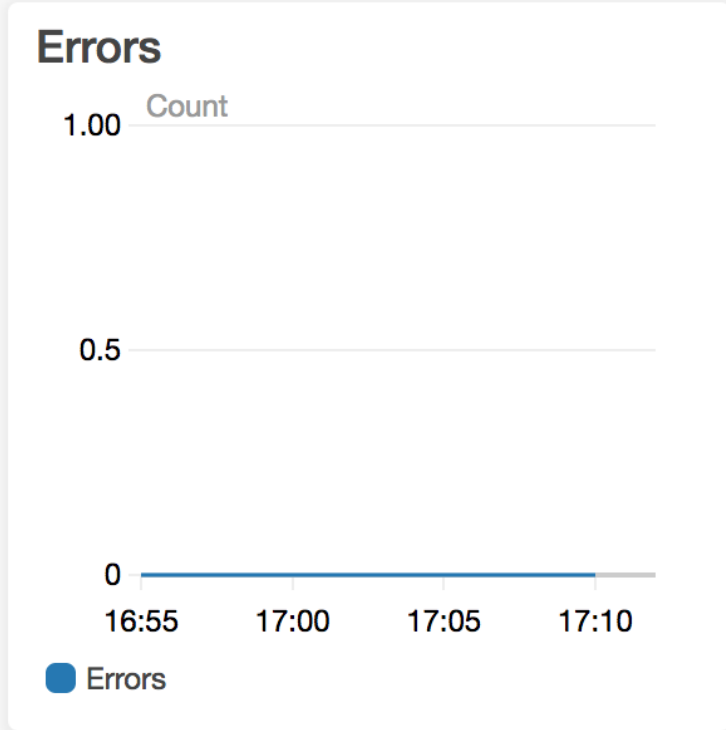
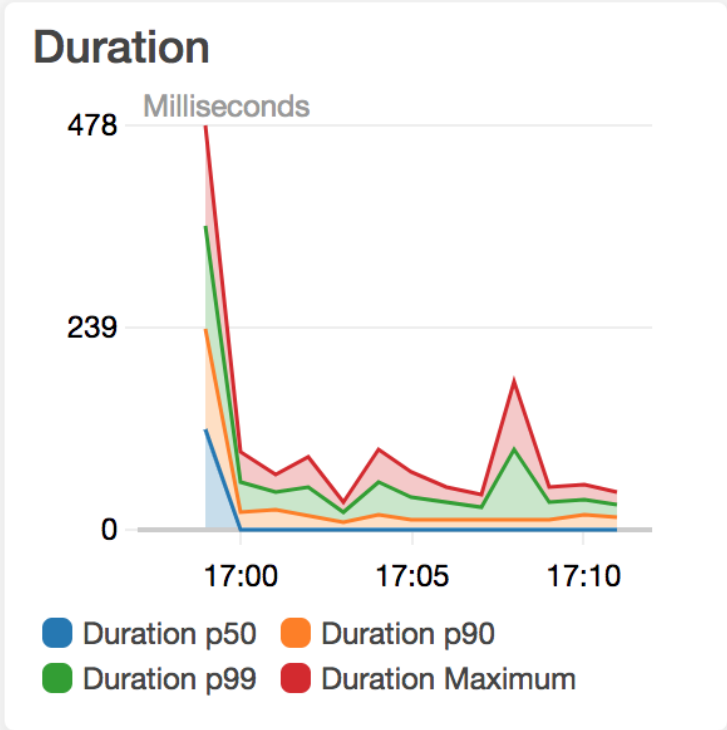
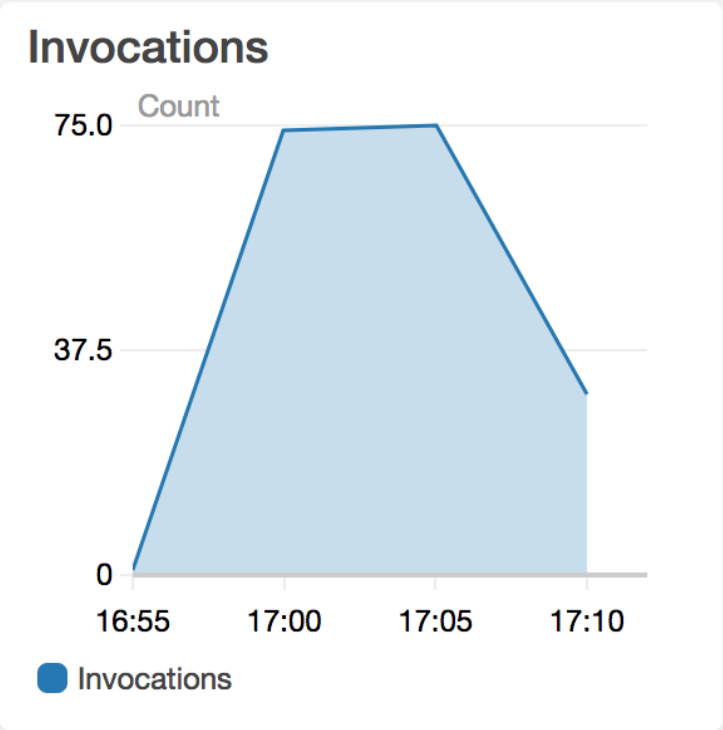
[List of Log Variables](#)

23:59:42

{ "requestId":"da82692e-ce4e-11e7-ada2-6d3cd4a95886",
{
 "requestId": "da82692e-ce4e-11e7-ada2-6d3cd4a95886",
 "ip": "54.240.196.186",
 "caller": "-",
 "user": "-",
 "requestTime": "20/Nov/2017:23:59:32 +0000",
 "httpMethod": "GET",
 "resourcePath": "/pets",
 "status": "200",
 "protocol": "HTTP/1.1",
 "responseLength": "3"
}

Log Example

Save Changes



X-Ray

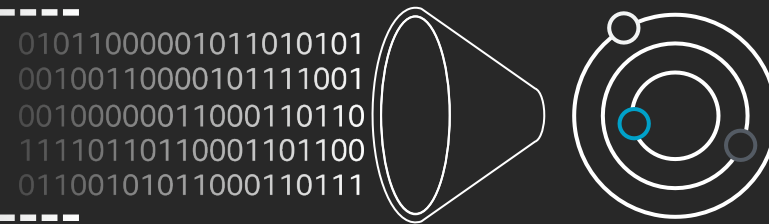
Solicitud de rastreo



X-Ray rastrea las solicitudes hechas a su aplicación

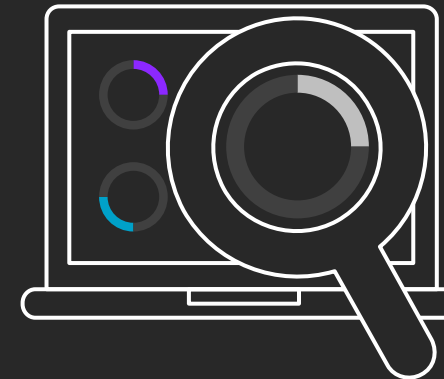
X-Ray recaba datos acerca de la solicitud de cada uno de los servicios de aplicaciones subyacentes que atraviesa

Registro de rastreos



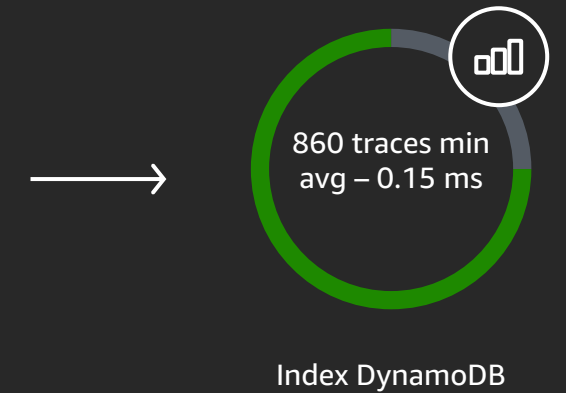
X-Ray combina los datos recabados de cada servicio en unidades singulares llamadas rastreos

Vista de mapa de servicios



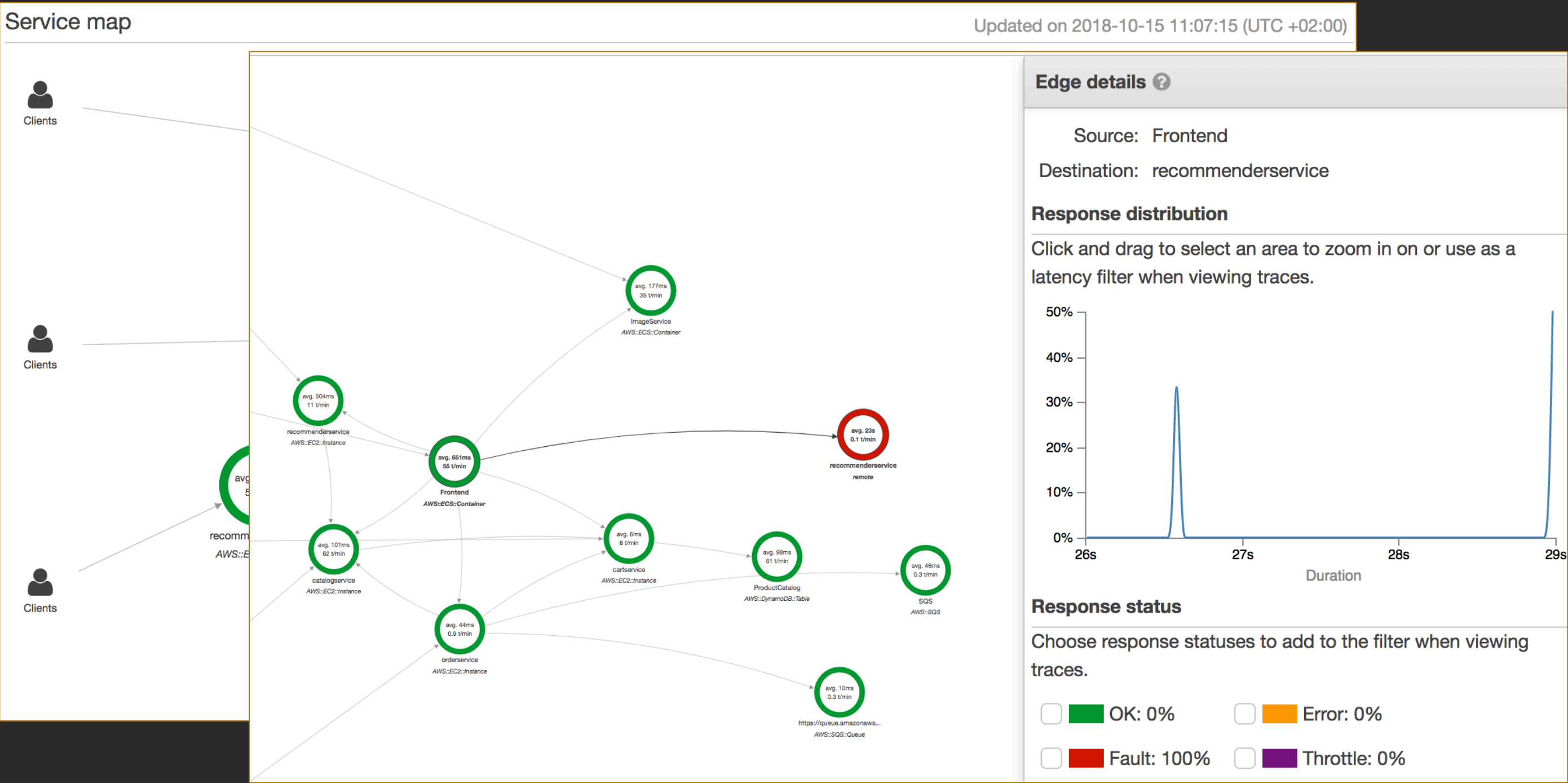
Visualiza el mapa de servicios para ver rastreo de datos como latencias, status de HTTP y metadata para cada servicio

Análisis de temas



Profundice en el servicio mostrando un comportamiento inusual para identificar el problema raíz

Así es como debe verse



A black and white historical photograph showing three workers in a trench installing large, corrugated metal pipes. One worker is on the left, another is in the center foreground, and a third is standing on top of a pipe in the background. The pipes are being laid in a trench, and the workers are using tools to secure them. The image has a grainy, historical quality.

Despliegue de su aplicación sin servidor

60X
Wise
Putting Iron Hoops
around Stave Pipe at Sta

CodeDeploy + Lambda



Usa **AWS SAM** para desplegar aplicaciones sin servidor

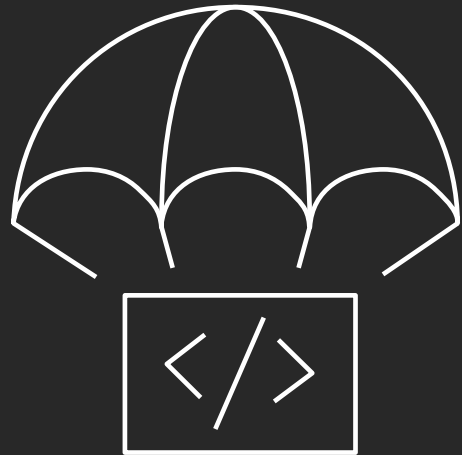
Soporta cambios de tráfico de alias **Lambda** y permite desplegar canaries y despliegues en azul/verde

Puede retroceder con base en métricas / alarmas CloudWatch

Disparadores pre/post-tráfico pueden integrarse con otros servicios (o incluso pedir funciones Lambda)



CodeDeploy + Lambda



CodeDeploy viene con múltiples capacidades añadidas:

Configuraciones de despliegue a la medida como:

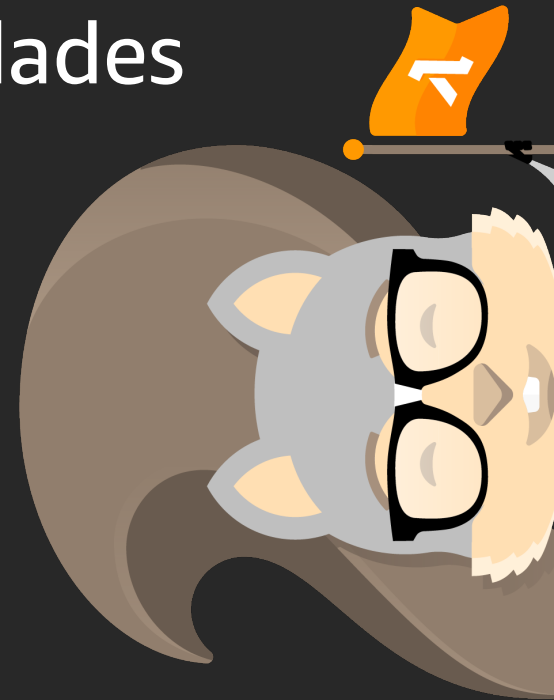
"Canary 5% por 1 hora"

"20% lineal cada 1 hora"

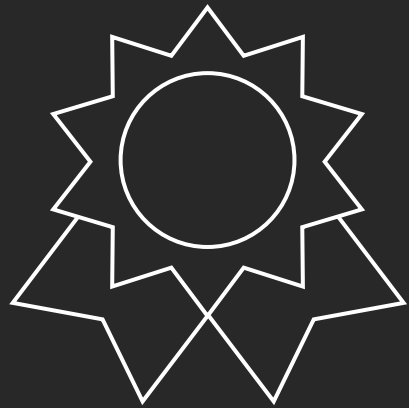
Notificaciones vía Amazon SNS de éxito/falla/retroceso

Consola con visibilidad de status de despliegue, historia y retrocesos

Despliegue Gradual de código (AWS SAM)



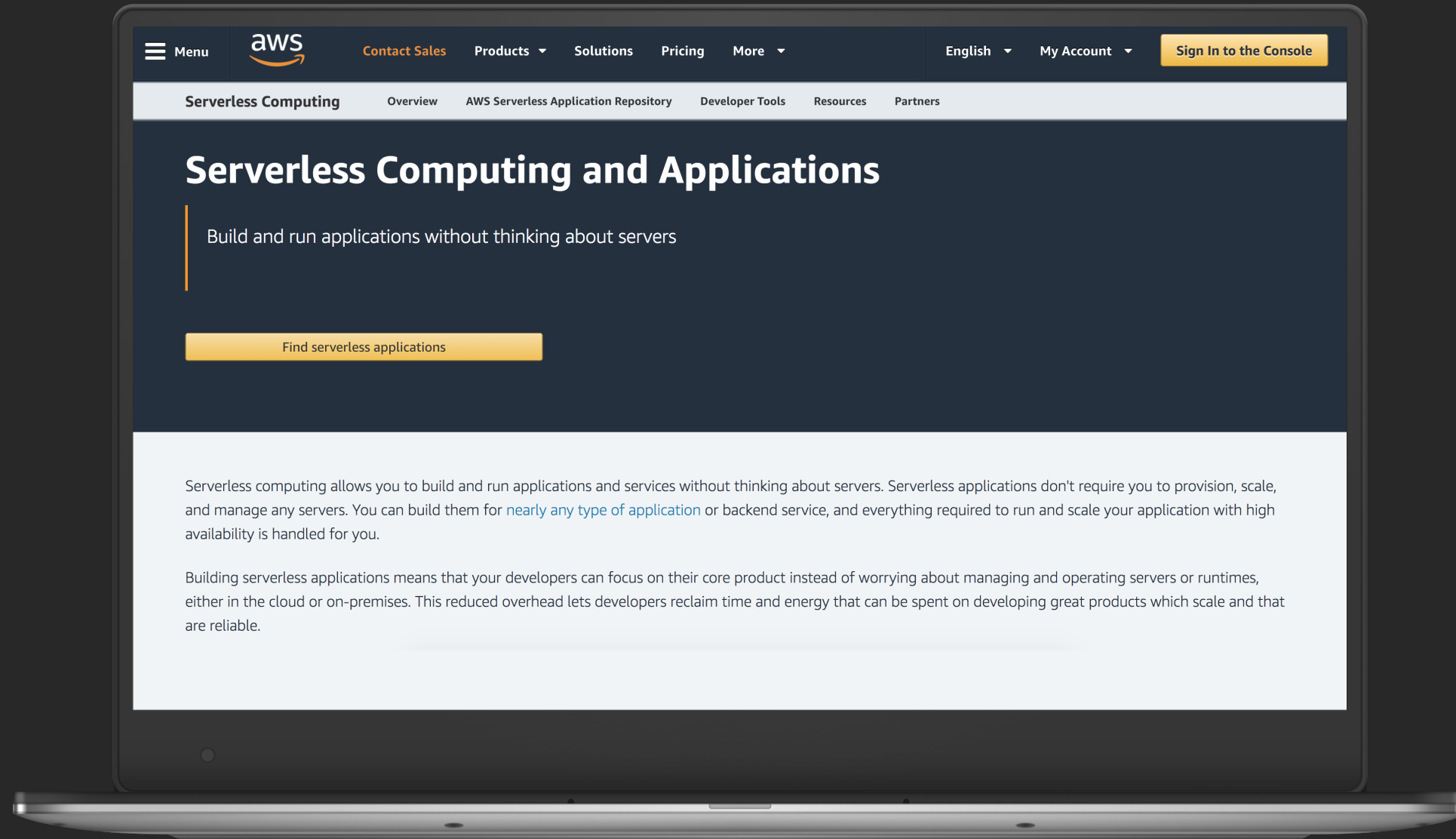
¿Ambientes, etapas, versiones y canaries?



Algunas mejores prácticas:

1. Use **azul**, **verde** o **canaries** para despliegue de producción con retroceso ya que permite la automatización.
2. En **versiones** de Lambda es útil si necesita soportar múltiples versiones para múltiples clientes/puntos de invocación.
3. En API Gateway, **las etapas** funcionan de modo similar y son útiles si requiere soportar **múltiples versiones API**.
4. Intente tener **siempre “stacks” separados** para ambientes de desarrollo, prueba, etapas y producción.
 - **No use etapas para** esto.

aws.amazon.com/serverless



Instituto Nacional Electoral

Caso de éxito AWS

Yuri Gonzalez

Director de Seguridad, Unidad de Servicios de Informática del
Instituto Federal Electoral

Agenda

- Acerca del INE
- Elecciones de México 2018 en AWS
- Servicios de AWS utilizados
- Resultados obtenidos
- Planes a futuro
- Conclusión

Acerca del INE

El Instituto Nacional Electoral es una entidad **pública, autónoma y permanente.**

Es independiente en cuanto a sus decisiones y funciones y cuenta con una personalidad jurídica y un presupuesto propio.

El presupuesto se envía al Congreso para que – en su caso - sea aprobado como parte del presupuesto general de egresos de la nación.



Acerca del INE

El Órgano de Dirección máximo del Instituto es el Consejo General.

Está integrado por **once integrantes con el derecho a voz y voto:**

- Un Consejero Presidente y
- Diez Consejeras y Consejeros Electorales,

Electos mediante un proceso publico por dos tercios de la cámara de Senadores.



Atribuciones del INE

En Elecciones Federales y Locales



Capacitación
Electoral



Geografía Electoral



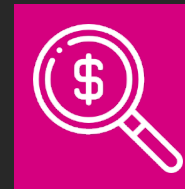
Registro Federal de
Electores



Ubicación de Casillas y
designación de funcionarios de
casilla



Definición de las reglas de
resultados preliminares, encuestas
de salida, conteos rápidos.



Control y supervisión del
financiamiento de Partidos
Políticos y campañas electorales

Elecciones de México en 2018

- 90 millones de potenciales votantes .
- Aproximadamente 157,000 casillas electorales.
- 1.5 millones de funcionarios de casilla reclutados de la ciudadanía.
- Más de 34,000 puestos de elección popular fueron concursados.



Sistemas Informáticos utilizados

35 Sistemas Informáticos soportan los diversos procesos del Instituto, entre los que se encuentran:

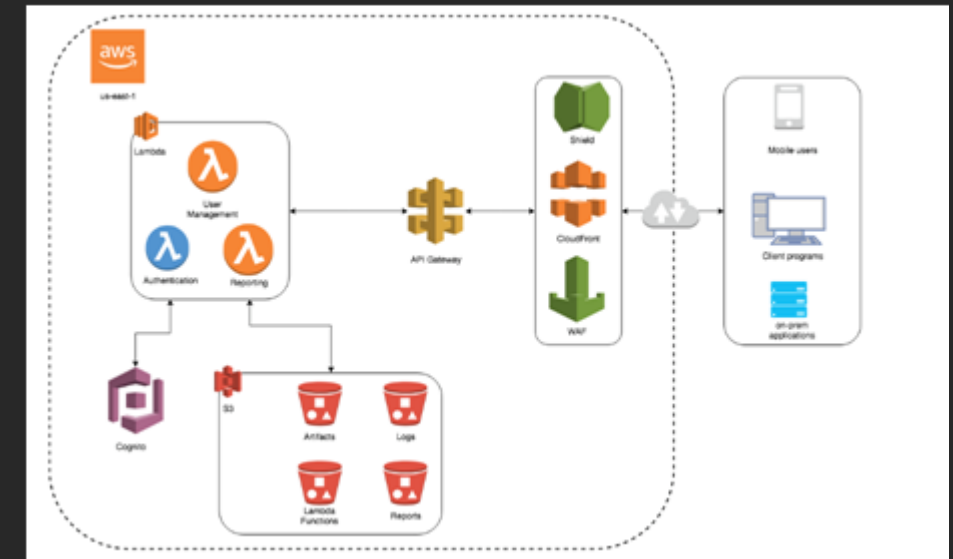
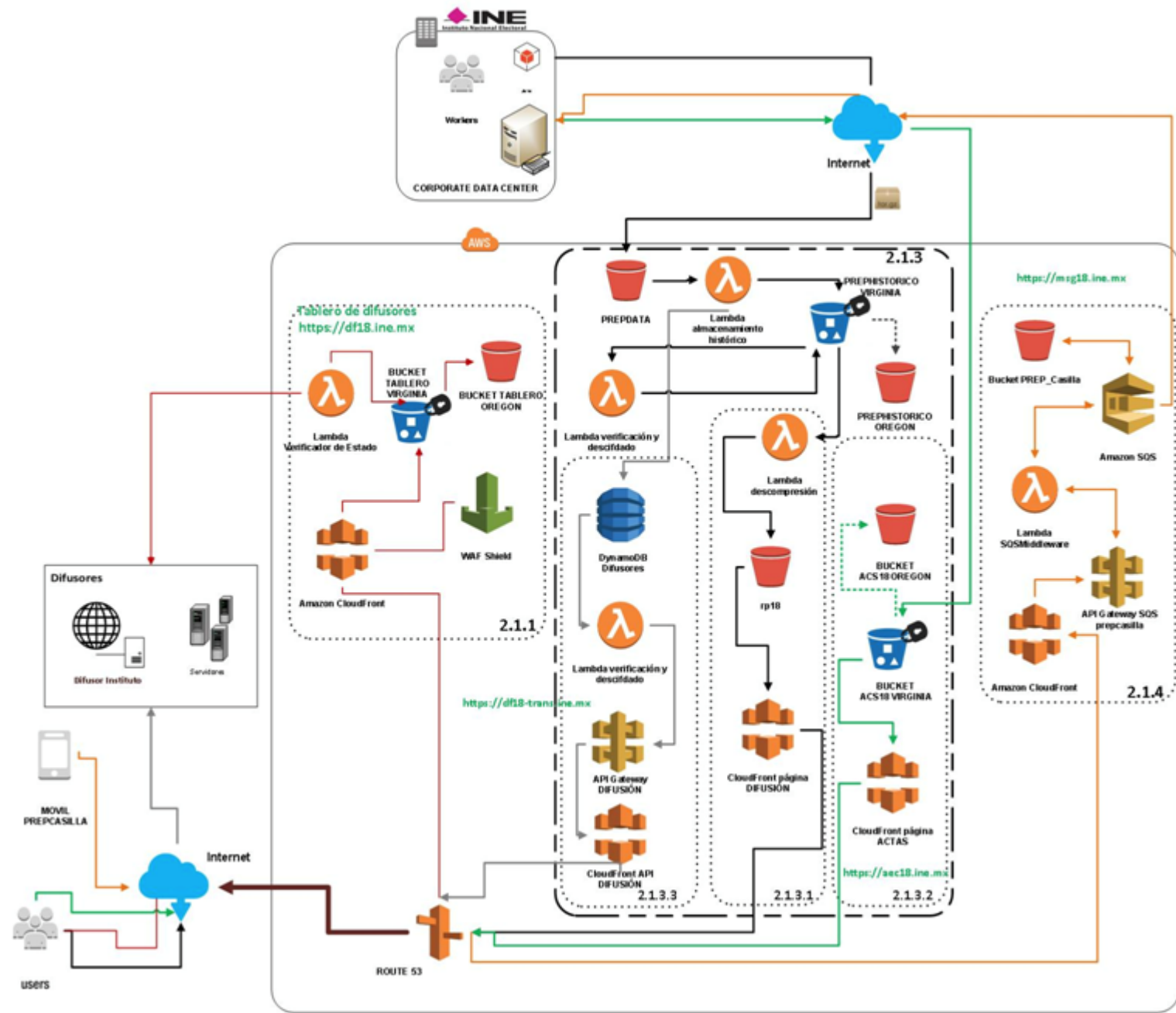
- Observadores Electorales
- Registro Nacional de Candidatos y Precandidatos
- Ubicación de Casillas
- **Sistema de Información de la Jornada Electoral**
- **Programa de Resultados Preliminares**
- Conteo Rápido

Servicios de AWS utilizados

Cognito	API Gateway
Guard Duty	Lambda
S3	Cloud Front
Shield	WAF
EC2	Dynamo DB
Aurora	IEM

- Simulación de ataque DDOS: Simulaciones (estilo Gameday) con equipo de servicios. La simulación logró 12K RPS y 9Gbps con ataques L3 y L4
- Cognito: 5 simulaciones que autentican a 40K usuarios. Este componente fue clave para la consolidación de votos.
- Simulación SQS con 450K mensajes en una hora
- Sistema de transmisión electoral: 100K RPS y 40Gbps

Arquitectura



Resultados obtenidos

- El día de las elecciones, todo el evento se desarrolló sin problemas ni interrupciones.
- Cuando se publicaron los resultados de las elecciones, en el pico de 1,5 millones de solicitudes por minuto estaban haciendo ping al sitio web principal y obteniendo un usuario único cargando el sitio web en 300 ms
- Los datos de los distintos puestos de votación se consolidaron en AWS y se compartieron con una sala llena de prensa cuando el presidente del INE anunció los resultados a las 20h.

Planes con AWS a futuro

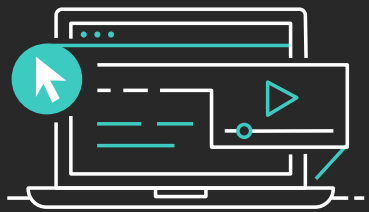
Migración de los Sistemas Informáticos soportan los diversos procesos del Instituto, entre los que se encuentran:

- Observadores Electorales
- Registro Nacional de Candidatos y Precandidatos
- Ubicación de Casillas
- Conteo Rápido

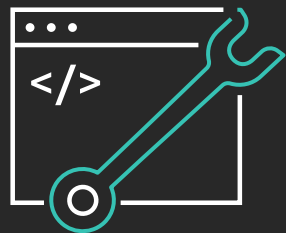
AWS formación técnica y certificación

Aprenda Serverless con AWS formación técnica y certificación

Recursos creados por los expertos de AWS para ayudarlo a aprender sobre aplicaciones sin servidor



Cursos gratuitos y bajo demanda de aplicaciones sin servidor, incluidos Fundamentos de AWS Lambda y profundos en AWS Fargate



Aprenda a "pensar sin servidores" con un nuevo curso de nivel intermedio de tres horas de Soluciones de arquitectura sin servidores

Visit the Learning Library at <https://aws.training>

Usando AWS para ejecutar cargas críticas en sector público: Consulta de resultados a elecciones nacionales o estatales



<https://go.aws/2VHr1g1>

iGracias!



Por favor completa la encuesta de la sesión en la aplicación móvil