



Habilitando negocios con arquitectura basada en eventos

Peterson Larentis
Sr. Serverless SA, LATAM

Conectando Sistemas

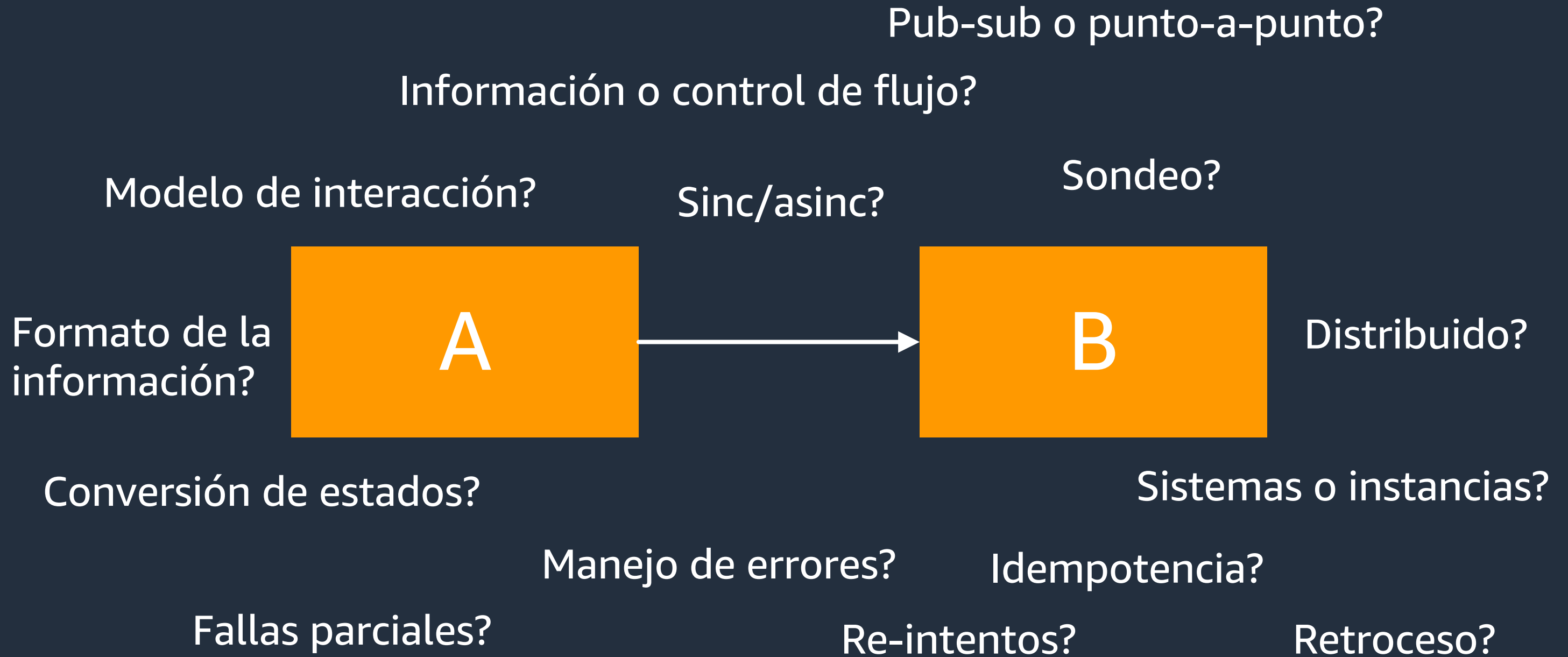
Al inicio es fácil utilizar arquitecturas acopladas ...



...pero la complejidad se multiplica al escalar

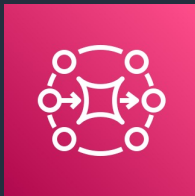
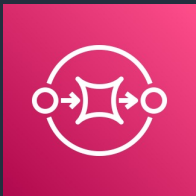
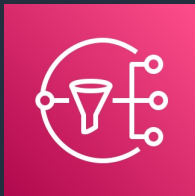
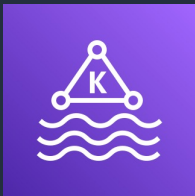


Conectar dos sistemas. Que tan difícil puede ser?



Separe su arquitectura de su elección de producto

La mayoría de productos combinan varios aspectos



- Orientado a mensajes
- Asíncrono
- Publicar-subscribir
- Eventos
- Basado en eventos
- Distribuida

X	X	X	X
X	X	X	X
X	X		
X	X		
?	?		
X	X	X	X

Nota: Únicamente para discusión. No es una tabla de propiedades.

Por que construir arquitecturas basadas en eventos?

Por que construir arquitecturas basada en eventos?



Tolerancia de fallos



Escalabilidad

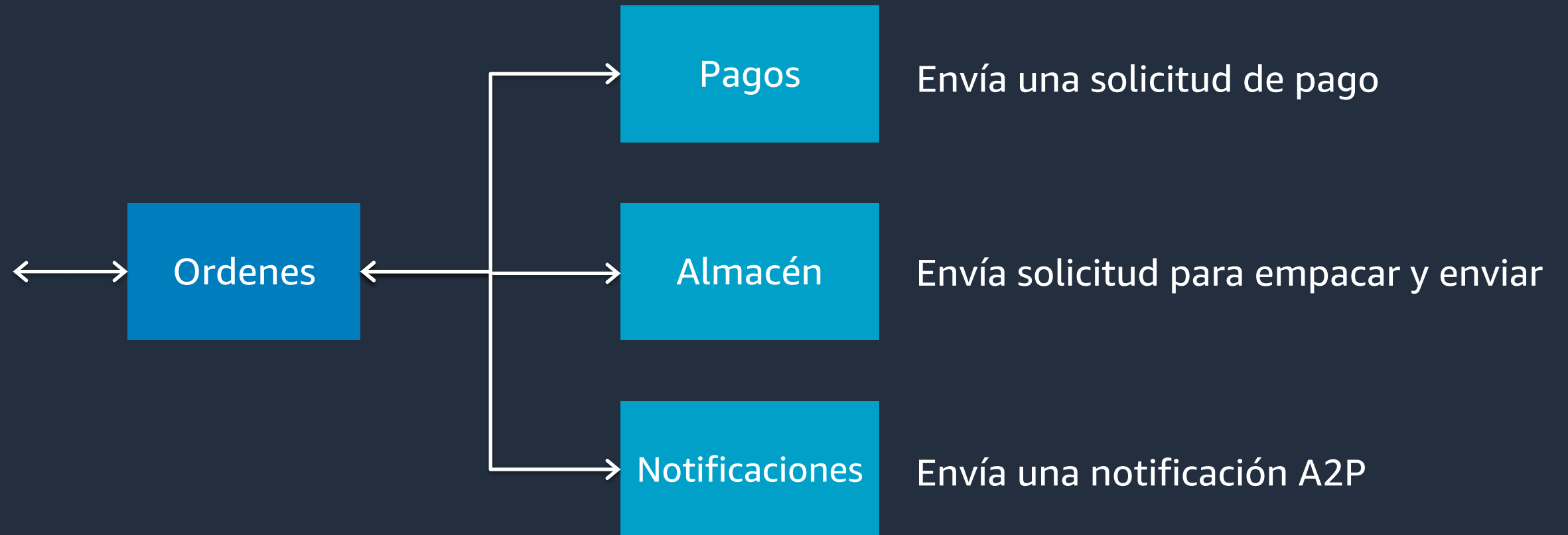


Extensible

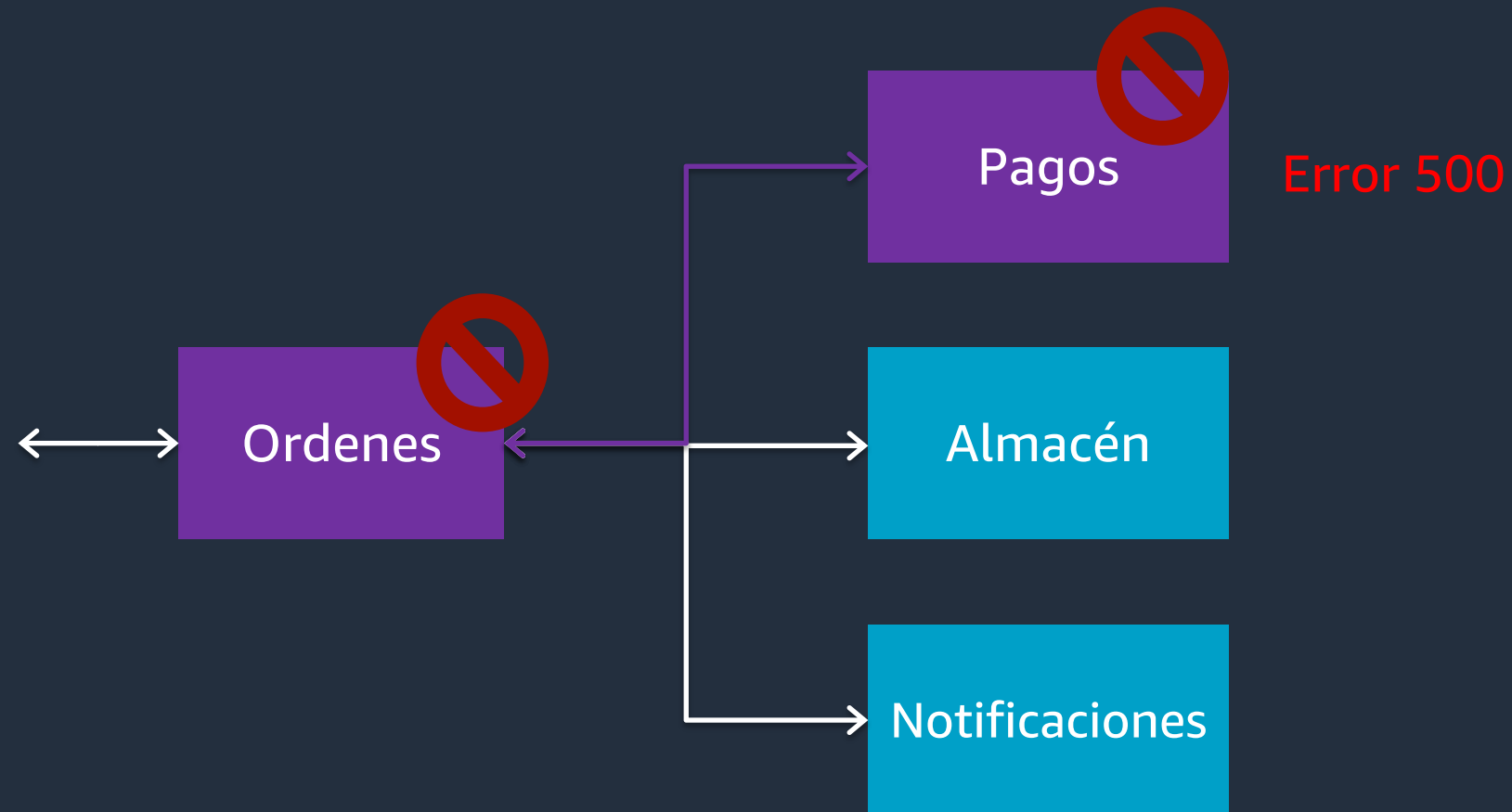


Agilidad

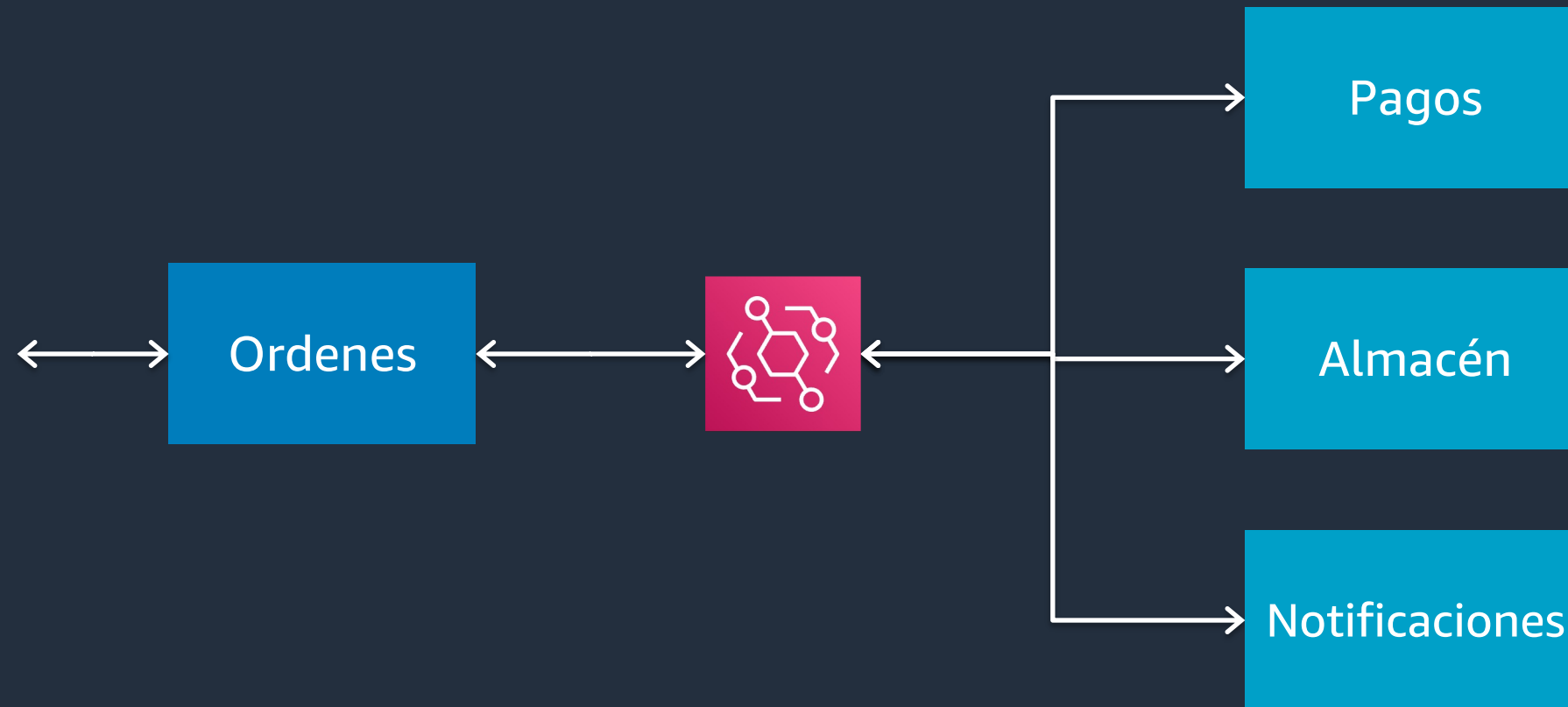
Ejemplo de arquitectura acoplada



Ejemplo de arquitectura acoplada



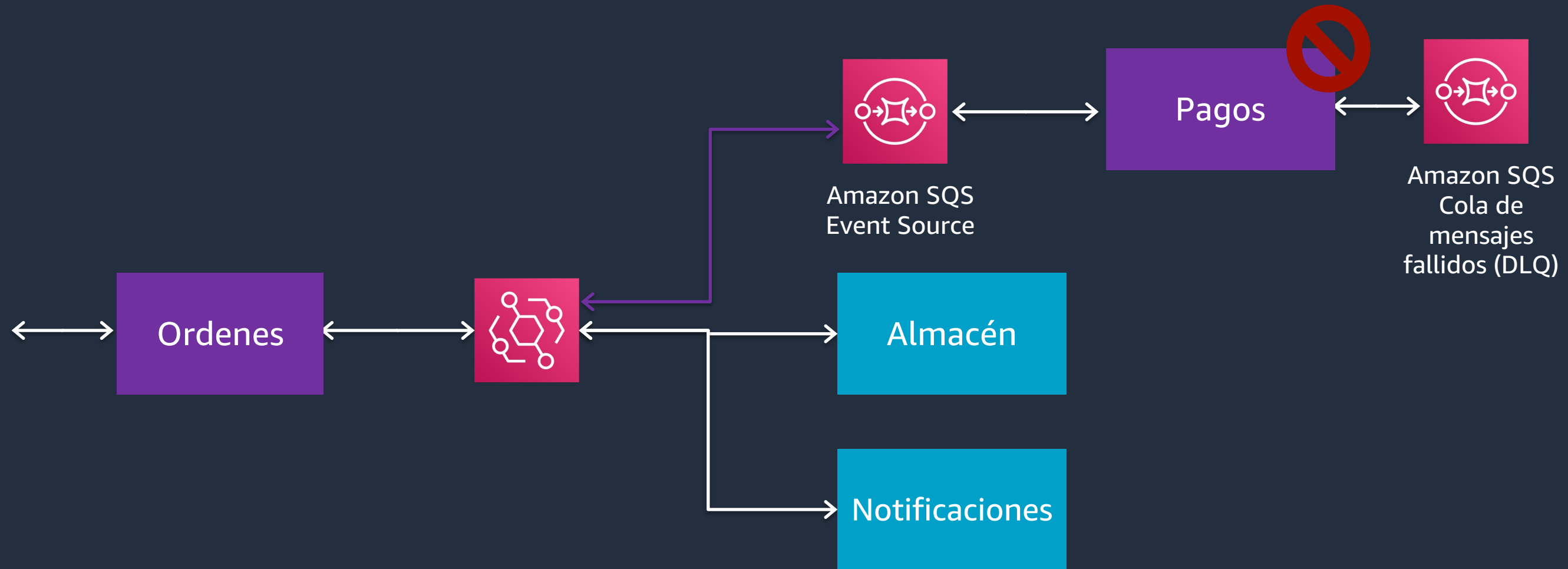
Ejemplo de arquitectura desacoplada



El servicio de ordenes ya no tiene que esperar por la respuesta de los otros servicios.

- Alta disponibilidad del servicio de ordenes
- Baja latencia para los usuarios finales.
- No mas fallas en cascada.

Ejemplo de arquitectura desacoplada



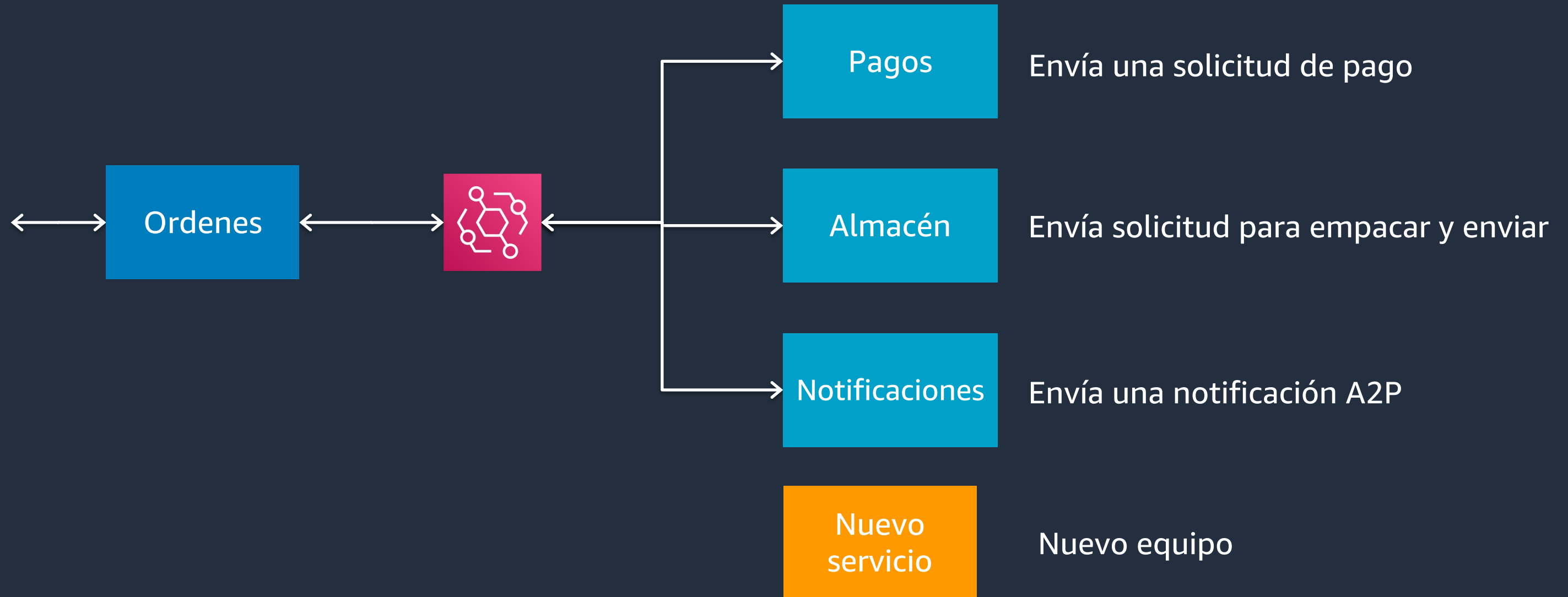
Ejemplo de arquitectura desacoplada (nuevo servicio)



Agilidad



Extensible



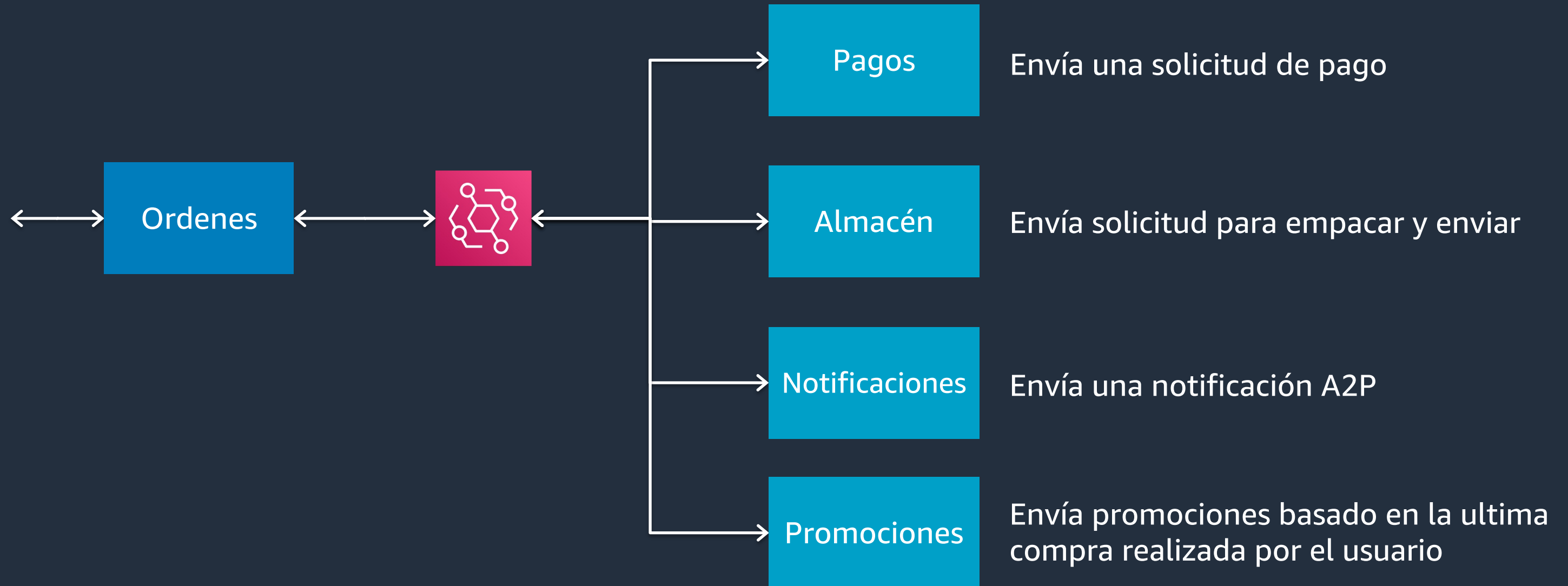
Ejemplo de arquitectura desacoplada (nuevo servicio)



Agilidad



Extensible



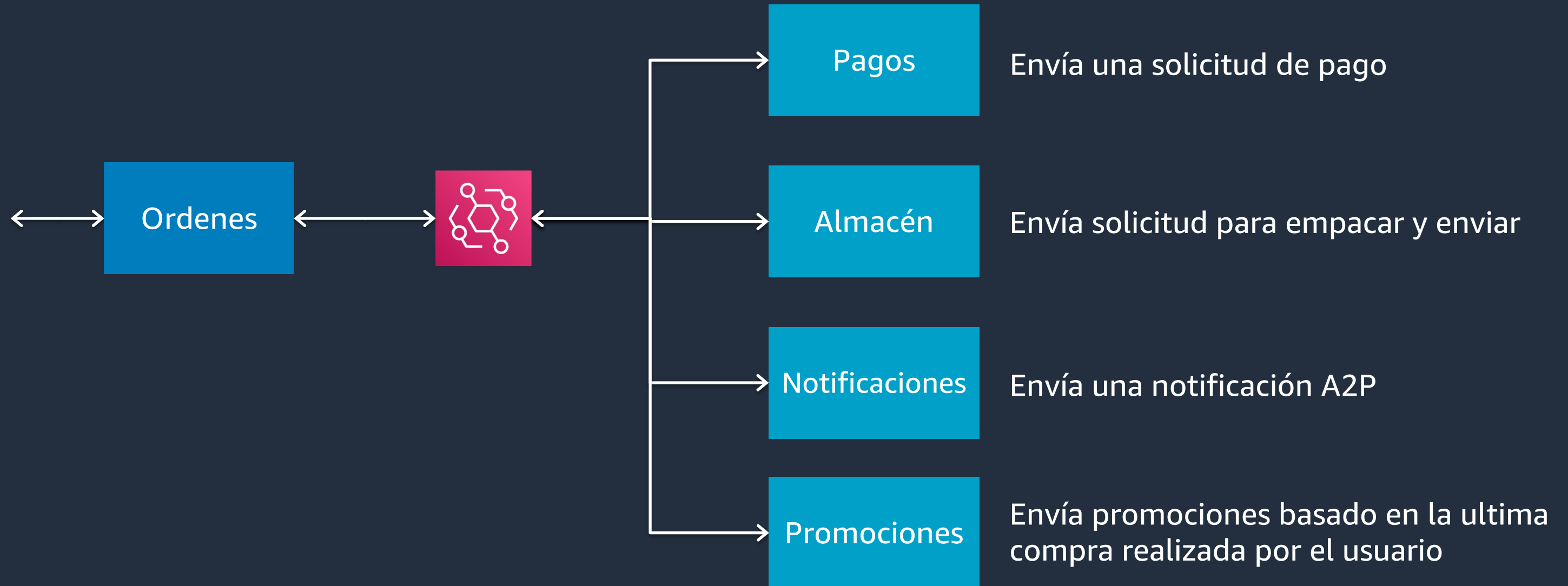
Ejemplo de arquitectura desacoplada (nuevo servicio)



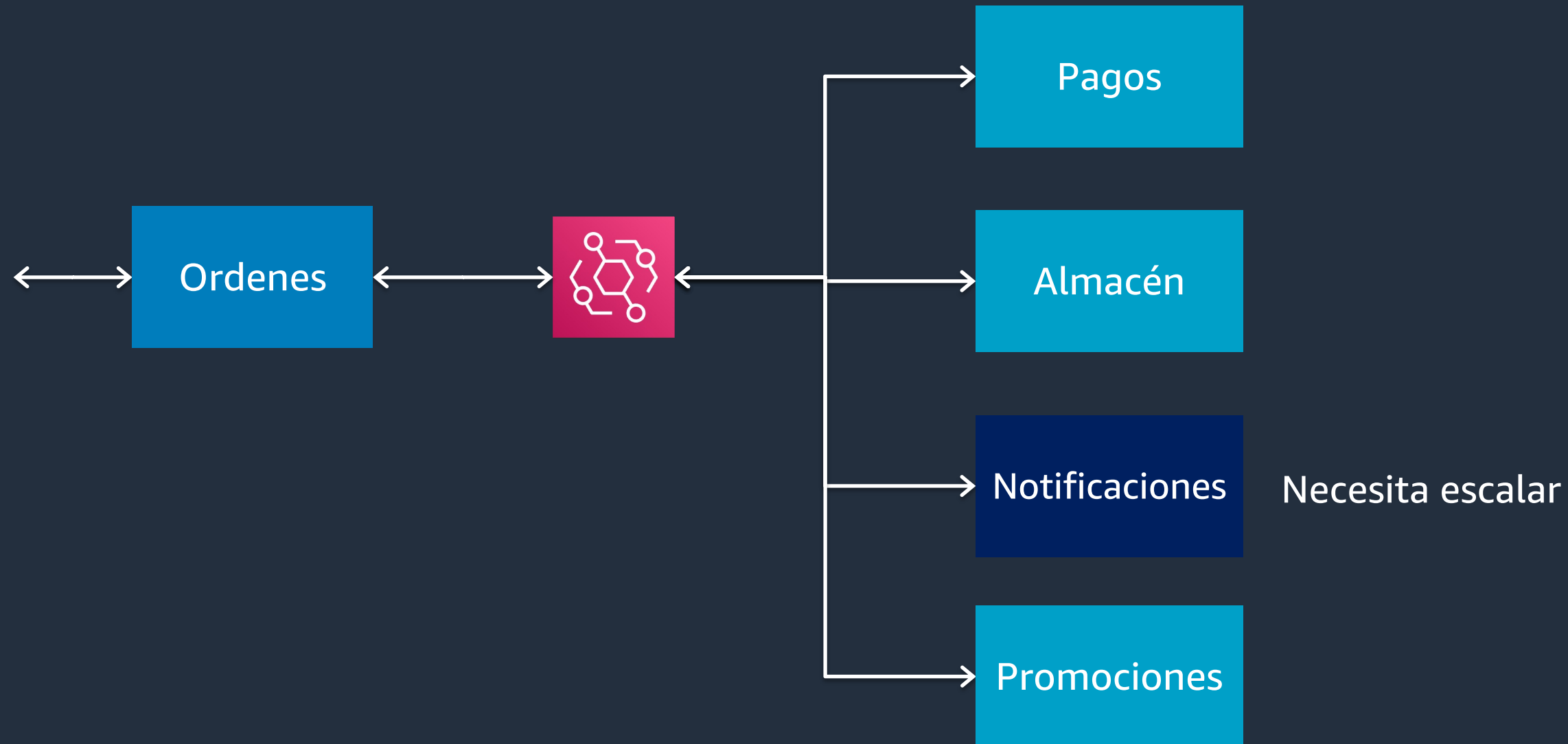
Agilidad



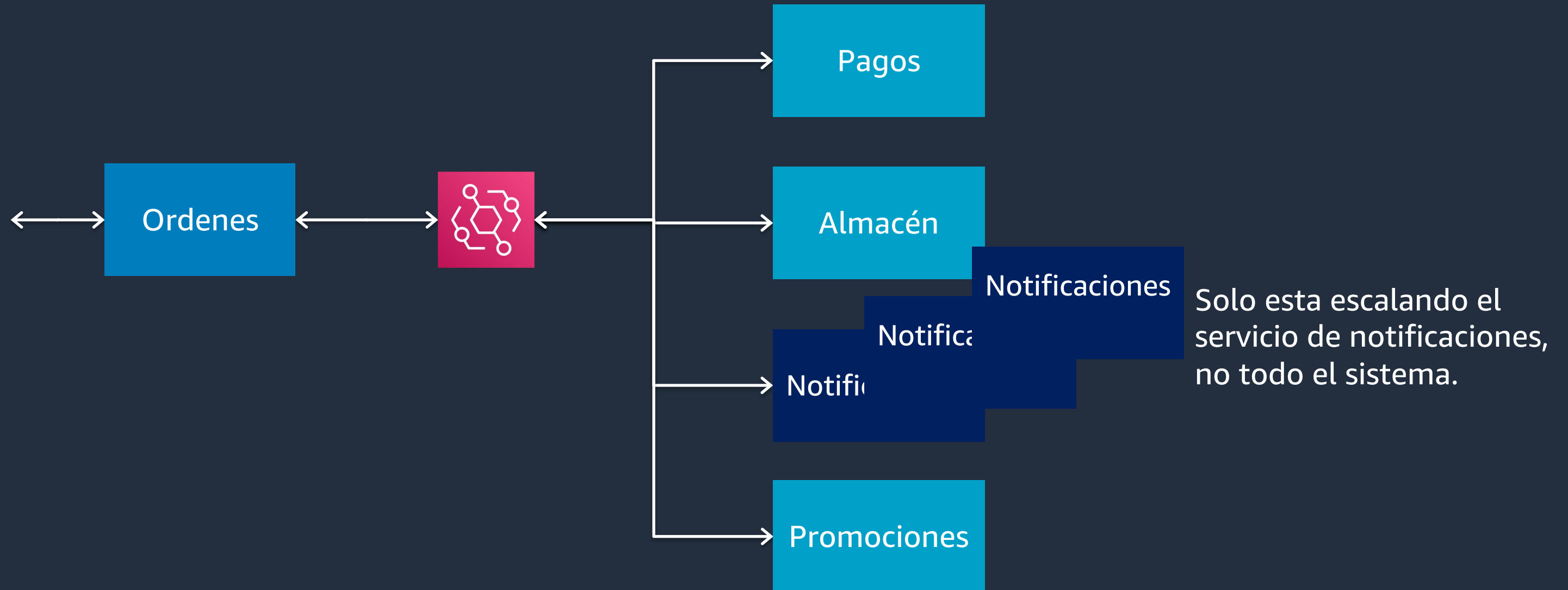
Extensible



Ejemplo de arquitectura desacoplada



Ejemplo de arquitectura desacoplada



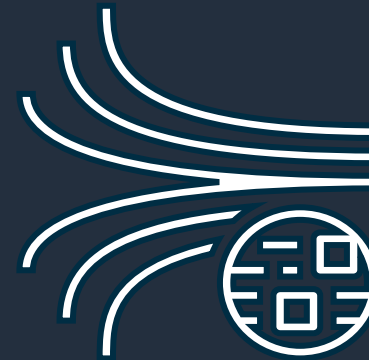
Que servicios de AWS se pueden utilizar en una arquitectura desacoplada?

Manejo de eventos



Enrutador de eventos

Filtra y enruta los eventos para enviarlos a los consumidores que están suscritos.



Transmisión de eventos

Transmisión continua de eventos donde los consumidores pueden extraer eventos

La arquitectura basada en eventos se divide en tres partes:



Productor de eventos

Publica eventos
(web o aplicaciones
móviles,
microservicios,
dispositivos IoT)



Ingestión de eventos



Consumidor de eventos

Procesa los eventos
(Actualiza la base de
datos, envía una
notificación, ejecuta
análisis de ML)

La arquitectura basada en eventos se divide en tres partes:



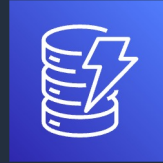
Productores de eventos



Amazon Simple Storage Service (S3)



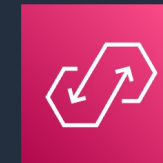
Amazon API Gateway



Amazon DynamoDB



AWS Lambda



AWS AppFlow



Amazon Redshift



Amazon Translate



AWS Step Functions

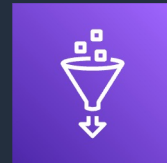
Cualquiera de los mas de 200+ Servicios de AWS produce eventos,
O tus eventos personalizados.



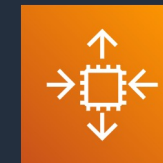
AWS Batch



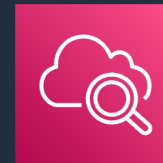
Amazon Elastic Container Service



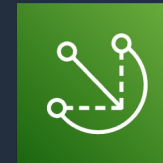
AWS Glue



Amazon EC2 Auto Scaling



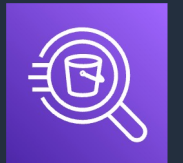
Amazon CloudWatch



AWS IoT Greengrass



Amazon Managed Blockchain



Amazon Athena

La arquitectura basada en eventos se divide en tres partes:



Productor de eventos

Publica eventos
(web o aplicaciones
móviles,
microservicios,
dispositivos IoT)



Ingestión de eventos



Consumidor de eventos

Procesa los eventos
(Actualiza la base de
datos, envía una
notificación, ejecuta
análisis de ML)

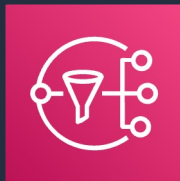
Manejo de eventos

	Almacenador de eventos		Enrutador de eventos	
	Colas	Transmisión	Temas	Bus de eventos
Nativos de AWS	 Amazon SQS	 Amazon Kinesis	 Amazon SNS	 Amazon EventBridge
Manejado por código abierto	 Amazon MQ	 Amazon MSK	 Amazon MQ	

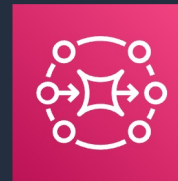
Comparando servicios para manejo de eventos: Operaciones



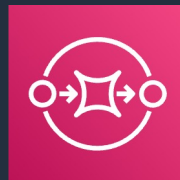
Amazon EventBridge



Amazon SNS



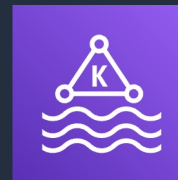
Amazon MQ



Amazon SQS



Amazon Kinesis
Data Streams



Amazon MSK



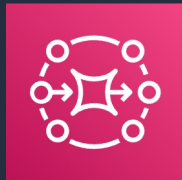
Menos

Responsabilidad operacional

Mas

Comparando servicios para el manejo de eventos: Ordenamiento

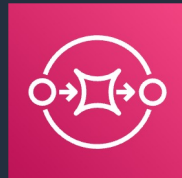
Garantiza orden de eventos



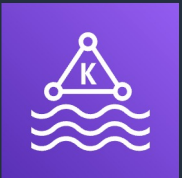
Amazon MQ



Amazon Kinesis
Data Streams



Amazon SQS
FIFO

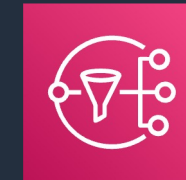


Amazon MSK

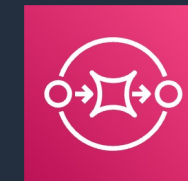
No garantiza orden de eventos



Amazon EventBridge



Amazon SNS



Amazon SQS

La arquitectura basada en eventos se divide en tres partes:



Productor de eventos

Publica eventos
(web o aplicaciones
móviles,
microservicios,
dispositivos IoT)



Ingestión de eventos



Consumidor
de eventos

Procesa los eventos
(Actualiza la base de
datos, envía una
notificación, ejecuta
análisis de ML)

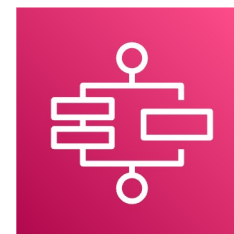
Event consumers

Los consumidores de eventos pueden ser independientes del tiempo de ejecución, el modelo operativo y la tecnología, ya que siempre y cuando puedan recibir eventos.

Los servicios sin servidor son especialmente adecuados para las basadas en eventos: solo se ejecutan cuando hay eventos que procesar y escalar hacia arriba y hacia abajo automáticamente con el volumen de eventos.



Invoke a Lambda function

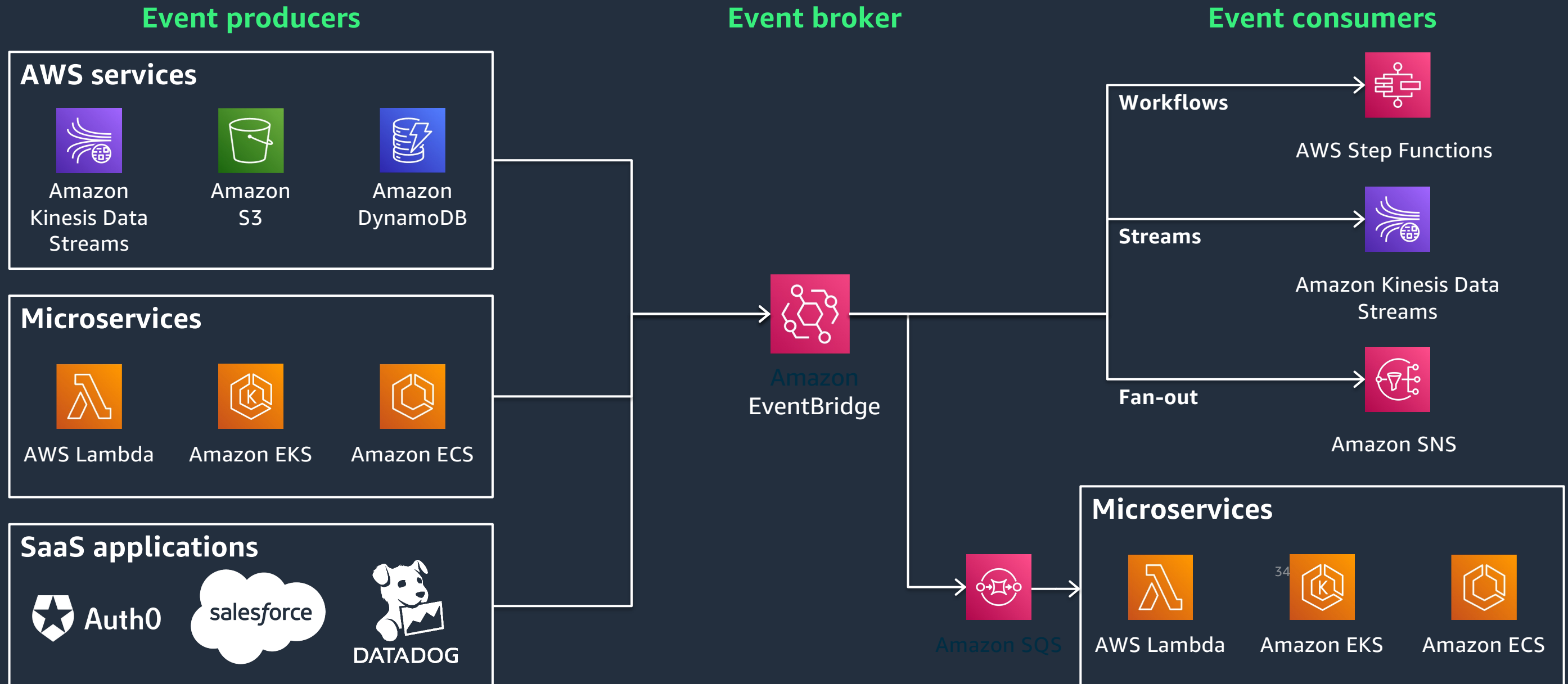


Invoke a Step Functions workflow



Send events to containers microservices

AWS portfolio of purpose-built services for EDAs



**Que están construyendo los clientes
con arquitecturas basadas en
eventos ?**

Customers building event-driven architectures on AWS



El middleware de pedidos de Taco Bell se lanza a producción en 2 meses y puede Escalar a millones de pedidos por hora



Nationwide Children's Hospital organiza la computación intensiva Análisis del cáncer con eventos y flujos de trabajo



Broadridge automates thousands of hours of manual data extraction and validation



MatHem aceleró el tiempo de lanzamiento de las funciones por 5-10x, construyendo una ventana de entrega característica MVP en 3 días

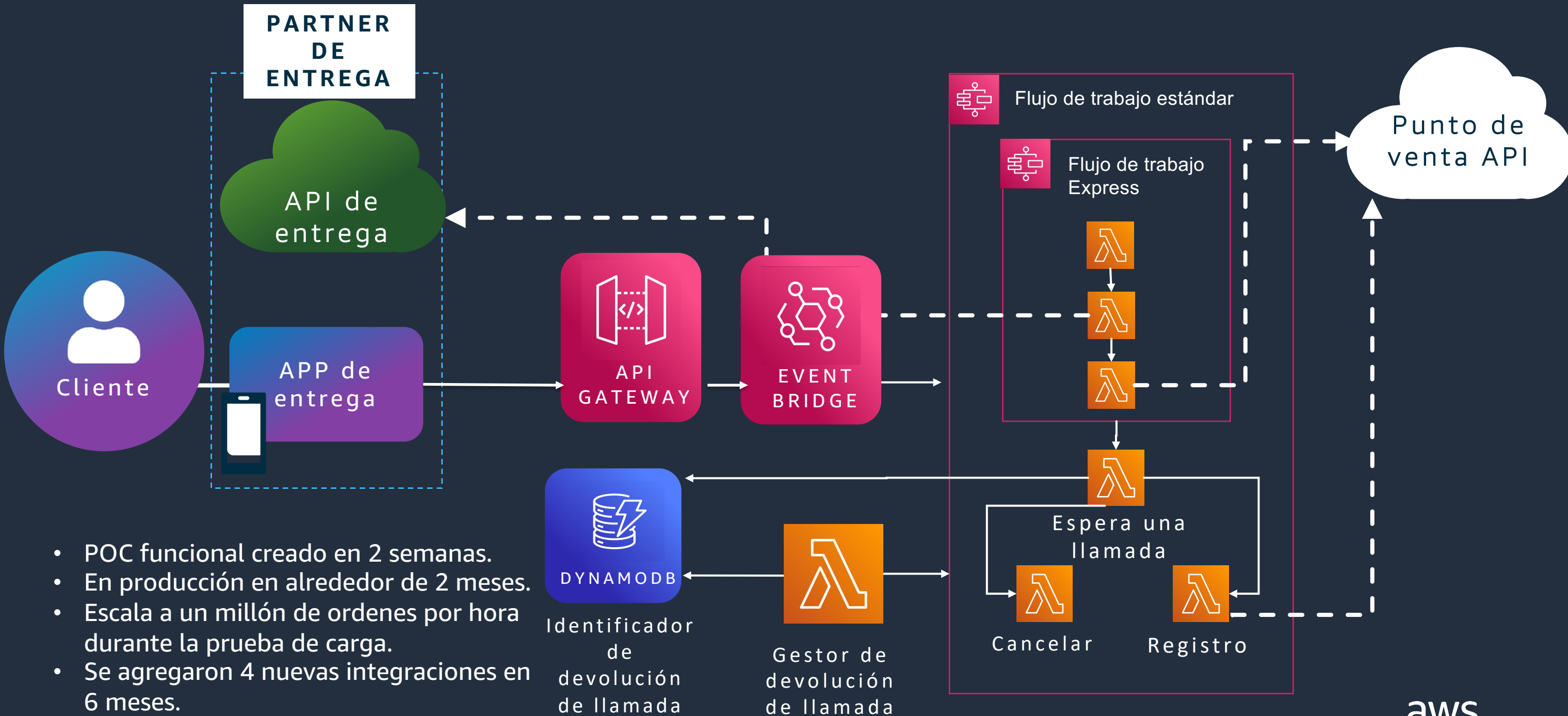
38

Lego utiliza una arquitectura basada en eventos diseñada para ser escalable.



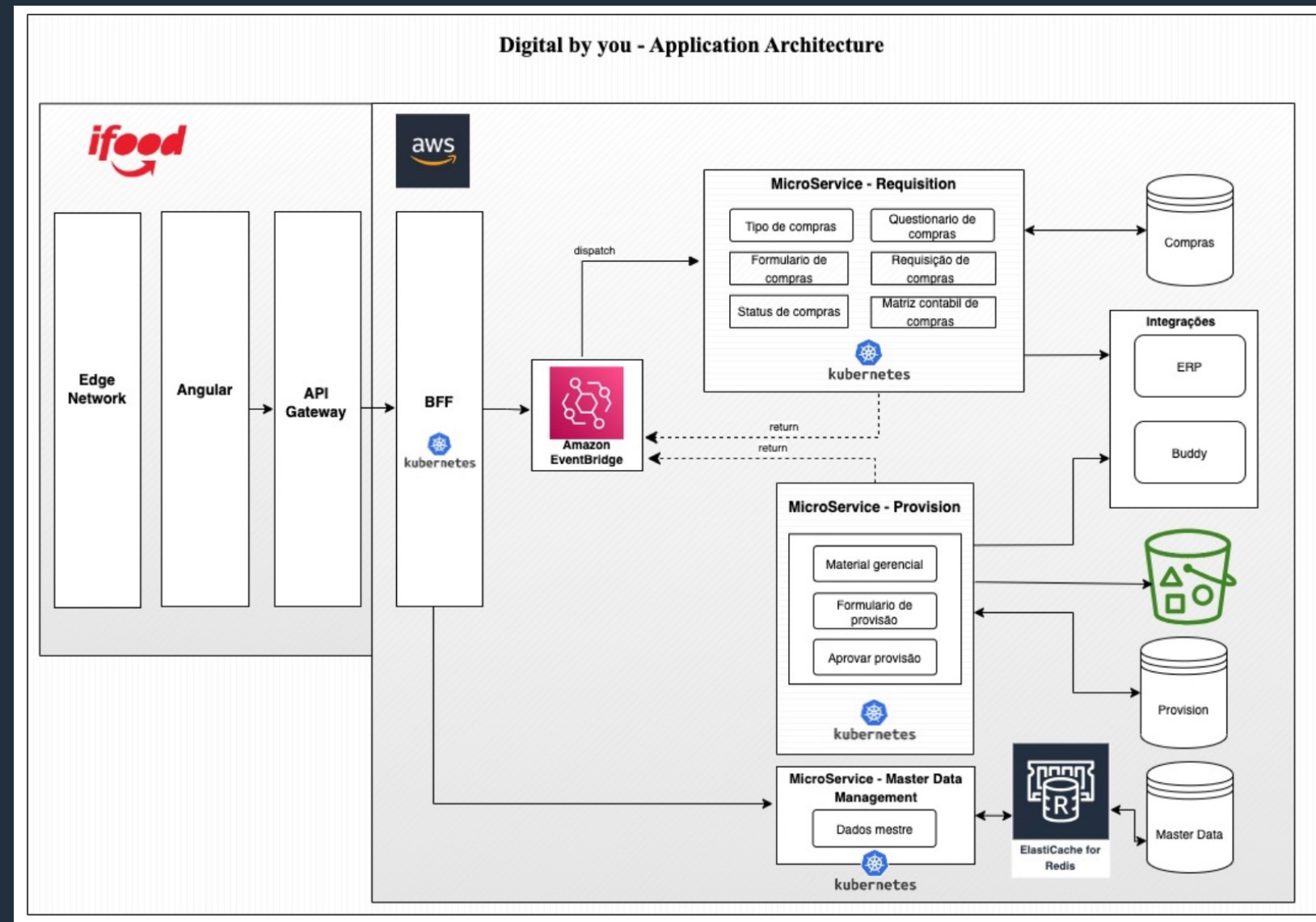
- El tráfico de Lego aumento durante días de alta demanda – esto llevo al fallo del “Black Friday” en el 2017.
- Una arquitectura basada en eventos escala automáticamente, eliminando los problemas de fallo durante los picos de carga.

Taco Bell – Solución partner middleware para órdenes



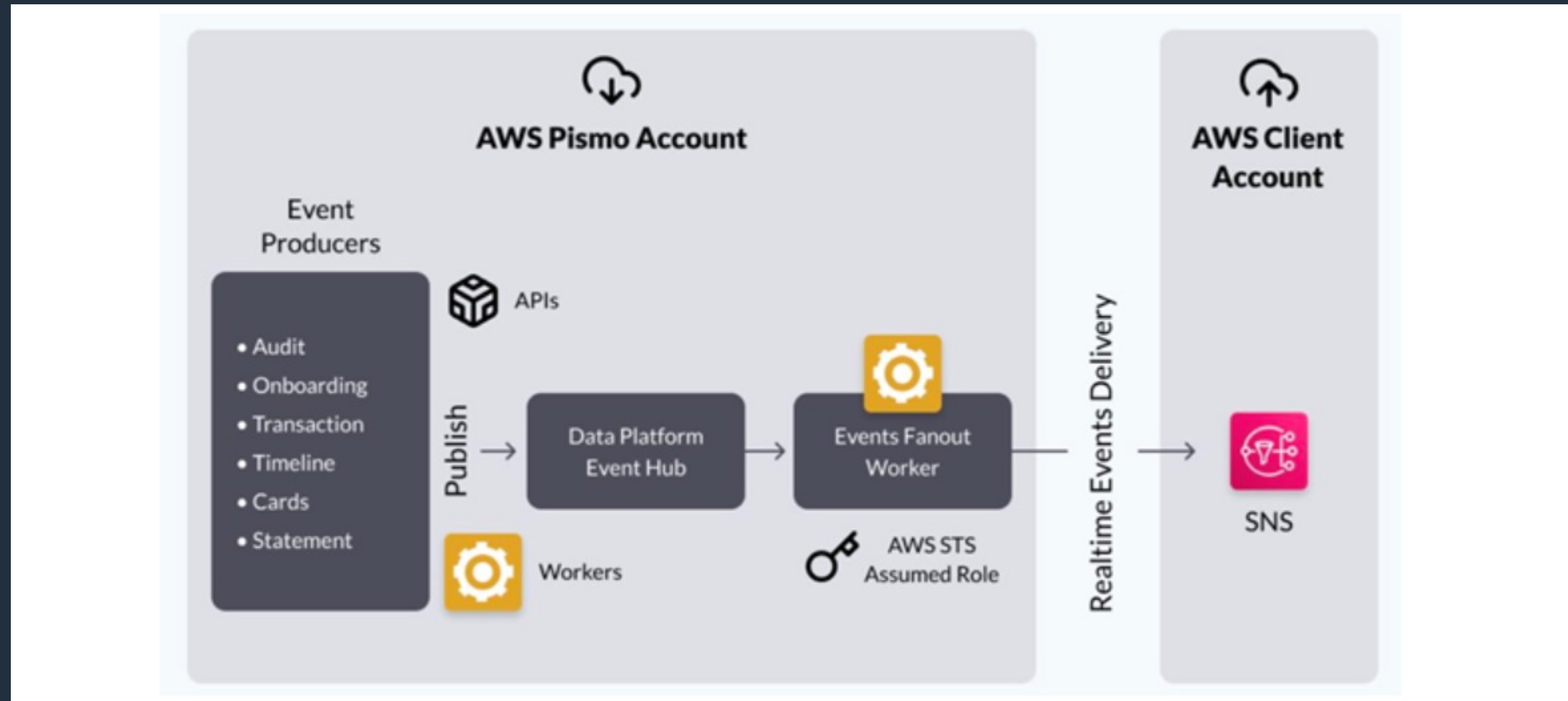
- POC funcional creado en 2 semanas.
- En producción en alrededor de 2 meses.
- Escala a un millón de órdenes por hora durante la prueba de carga.
- Se agregaron 4 nuevas integraciones en 6 meses.

Cómo iFood se benefició de la arquitectura impulsada por eventos para modernizar su middleware financiero



<https://aws.amazon.com/pt/blogs/aws-brasil/como-ifood-se-beneficiou-da-arquitetura-orientada-a-eventos-para-modernizar-seu-middleware-financeiro/>

Cómo Pismo procesa 200 millones de eventos al día con los servicios de AWS



<https://aws.amazon.com/pt/blogs/aws-brasil/como-a-pismo-processa-200-milhoes-de-eventos-por-dia-utilizando-servicos-aws/>

Takeaways

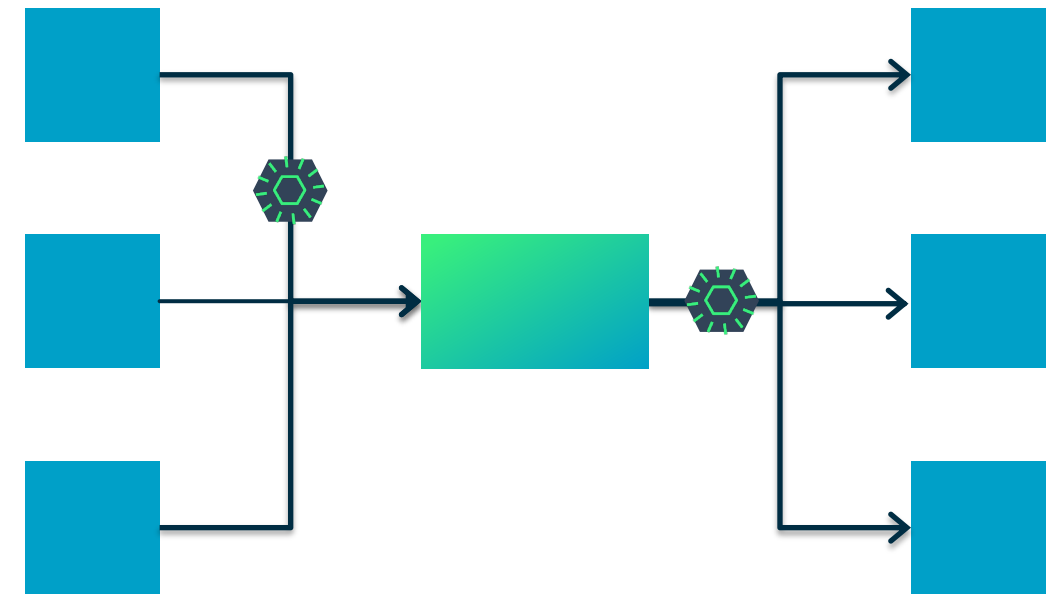
In an event-driven architecture, systems and development teams **communicate** via events

EDAs can give you:

- Greater developer agility and extensibility
- Increased scalability and fault tolerance
- Lower total cost of ownership

Serverless services are uniquely compatible with EDA

Combine multiple **purpose-built** services into EDA patterns

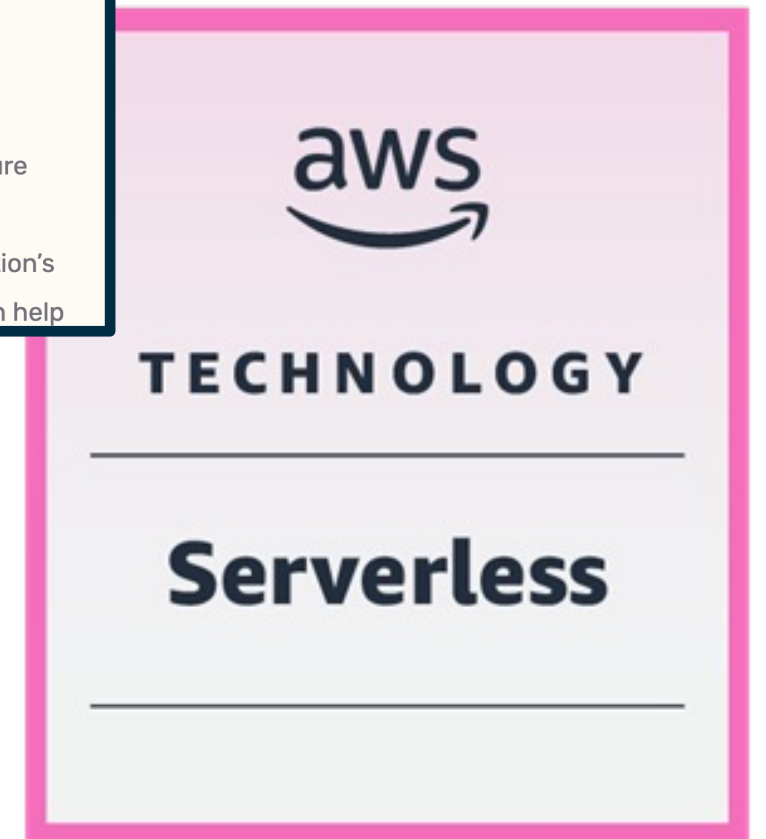
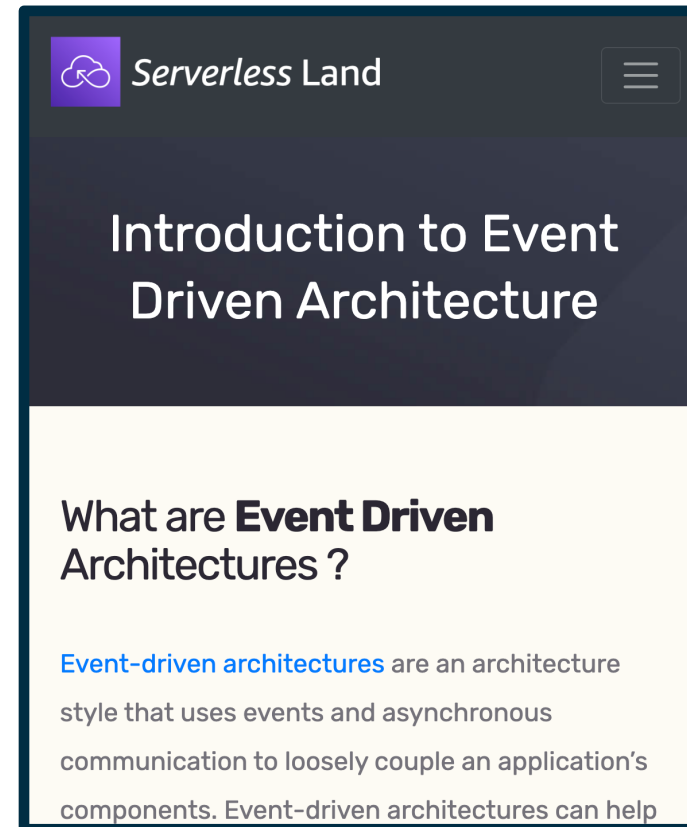


Learn more

Event-Driven Architectures on Serverless Land

- Workshops
- Guides
- Architecture patterns

Serverless Learning Badge on AWS Skill Builder



Muchas gracias!

plarenti@

