



Arquitecturas de nube con AWS

Ing. Fernando Lichtschein

Ing. Mora Villa Abrille

13. Arquitecturas desacopladas

Objetivos

- Distinguir arquitecturas con alto y bajo acoplamiento.
- Comprender el funcionamiento y los casos de uso de Amazon Simple Queue Service (Amazon SQS).
- Comprender el funcionamiento e identificar cuándo usar Amazon Notification Service (Amazon SNS).
- Describir Amazon MQ.
- Desacoplar cargas de trabajo con Amazon SQS.

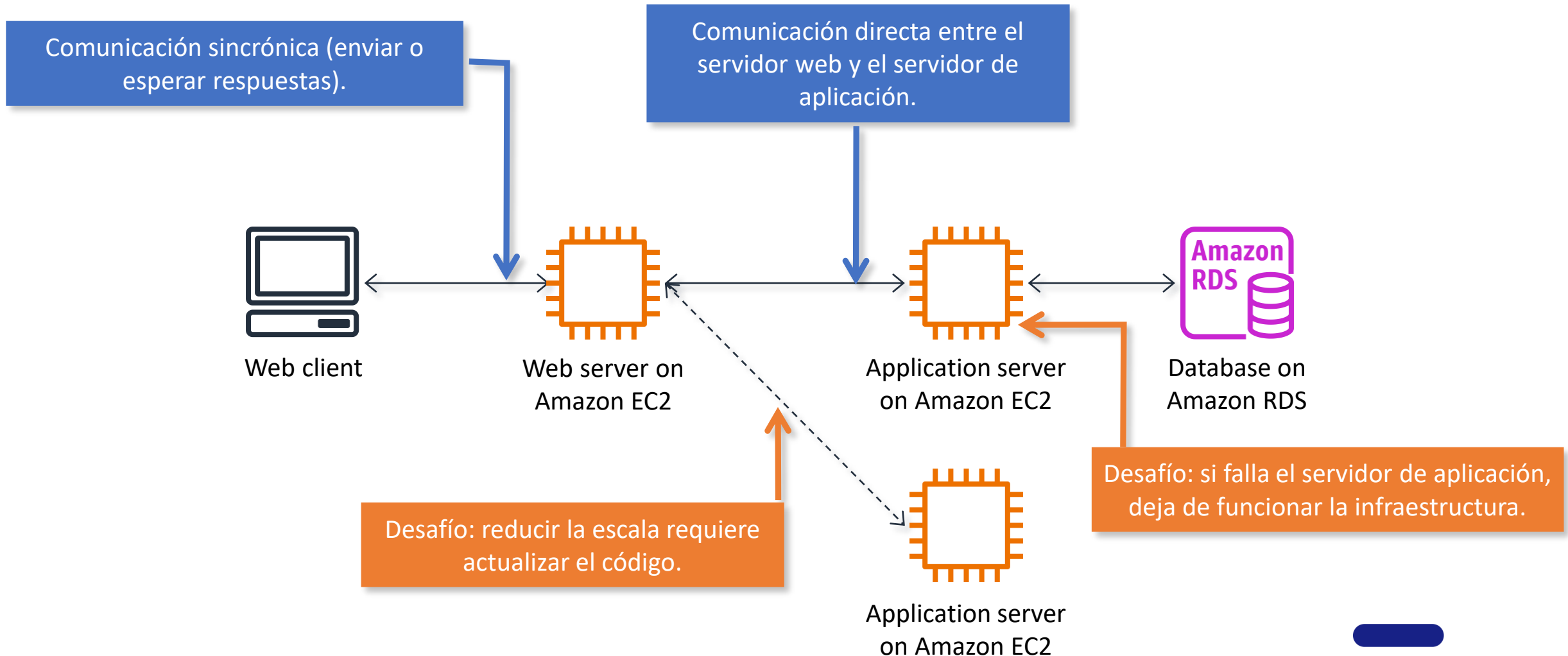
Objetivos

De un arquitecto de nube

- Identificar cuellos de botella potenciales para que la arquitectura escale en la medida que aumenta el tráfico.
- Limitar el impacto de las fallas para que las fallas de los componentes no produzcan la caída de la aplicación.
- Implementar arquitecturas que permitan aplicar cambios a un componente sin afectar a los demás, para minimizar las bajas por mantenimiento.

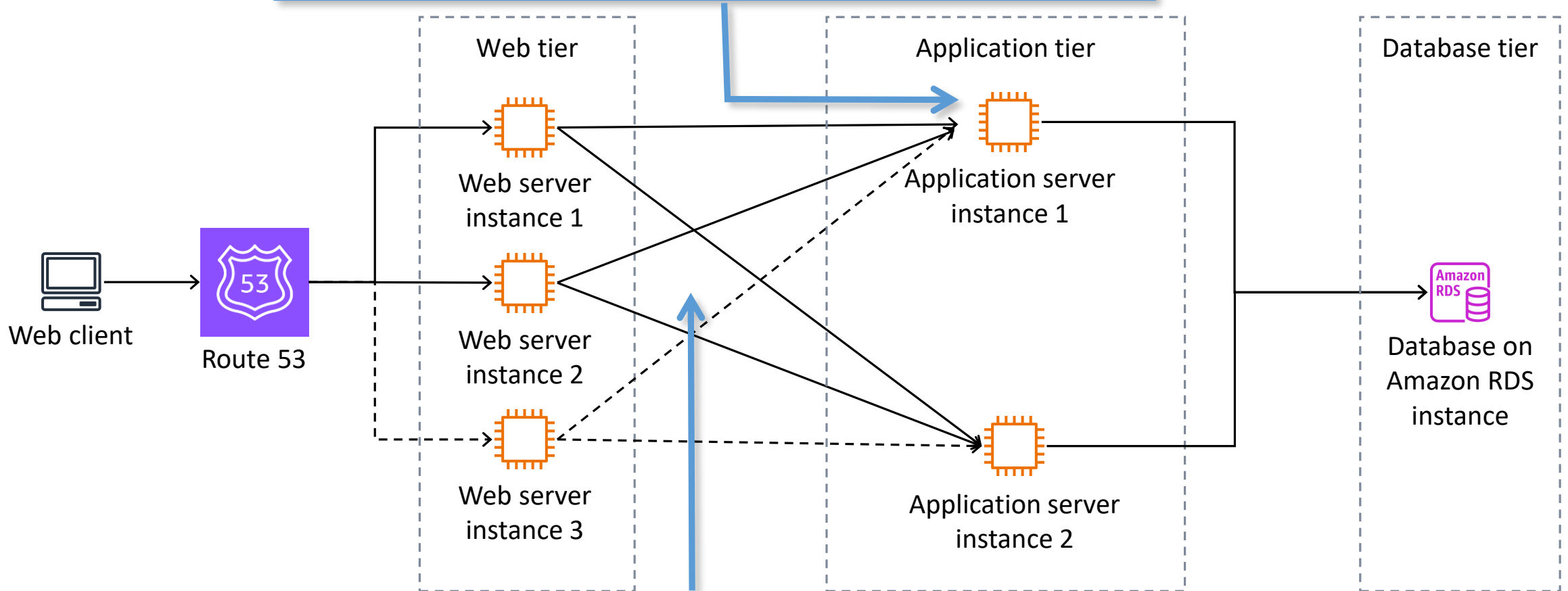
Acoplamiento

Acoplamiento fuerte en arquitecturas de tres capas



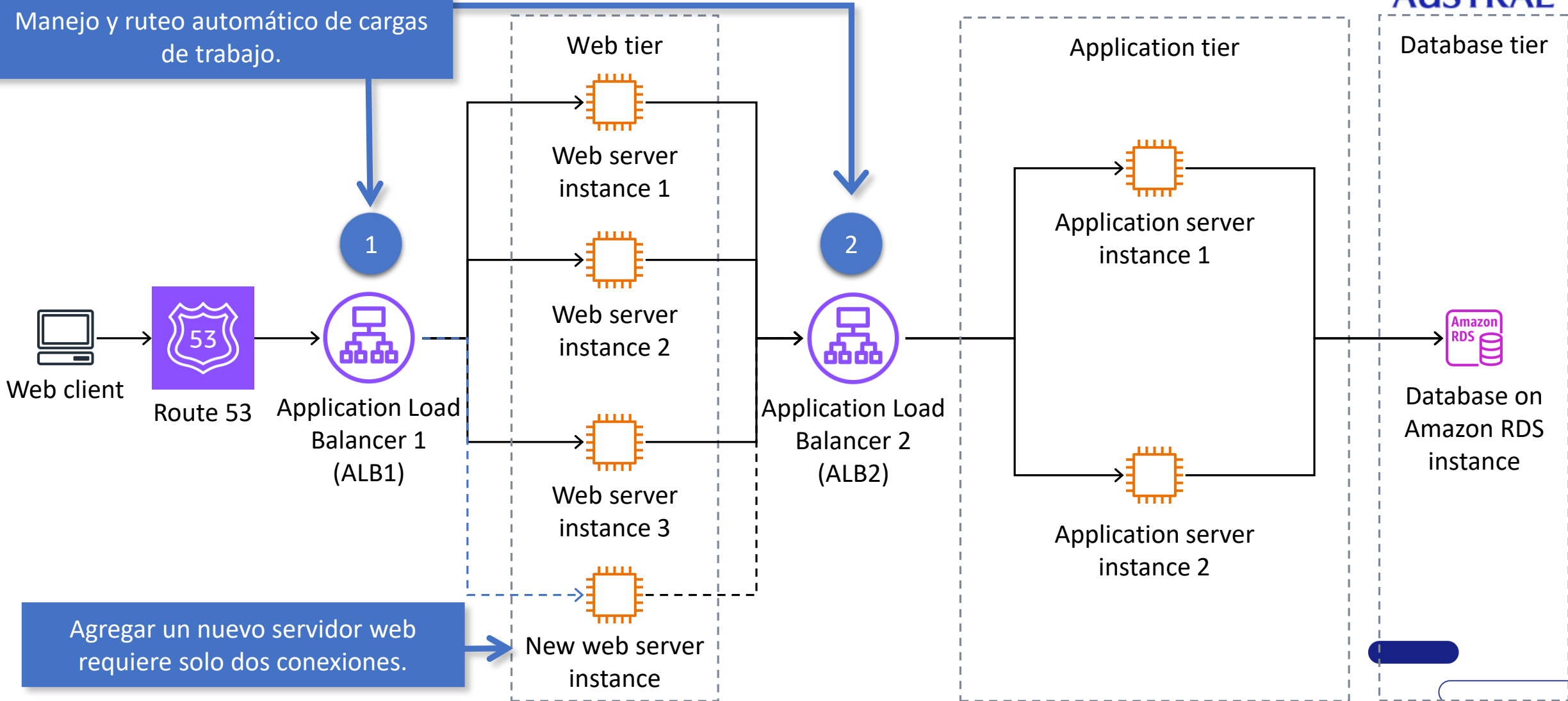
El acoplamiento dificulta el escalamiento

Una caída de la aplicación tiene impacto en todos los web servers.

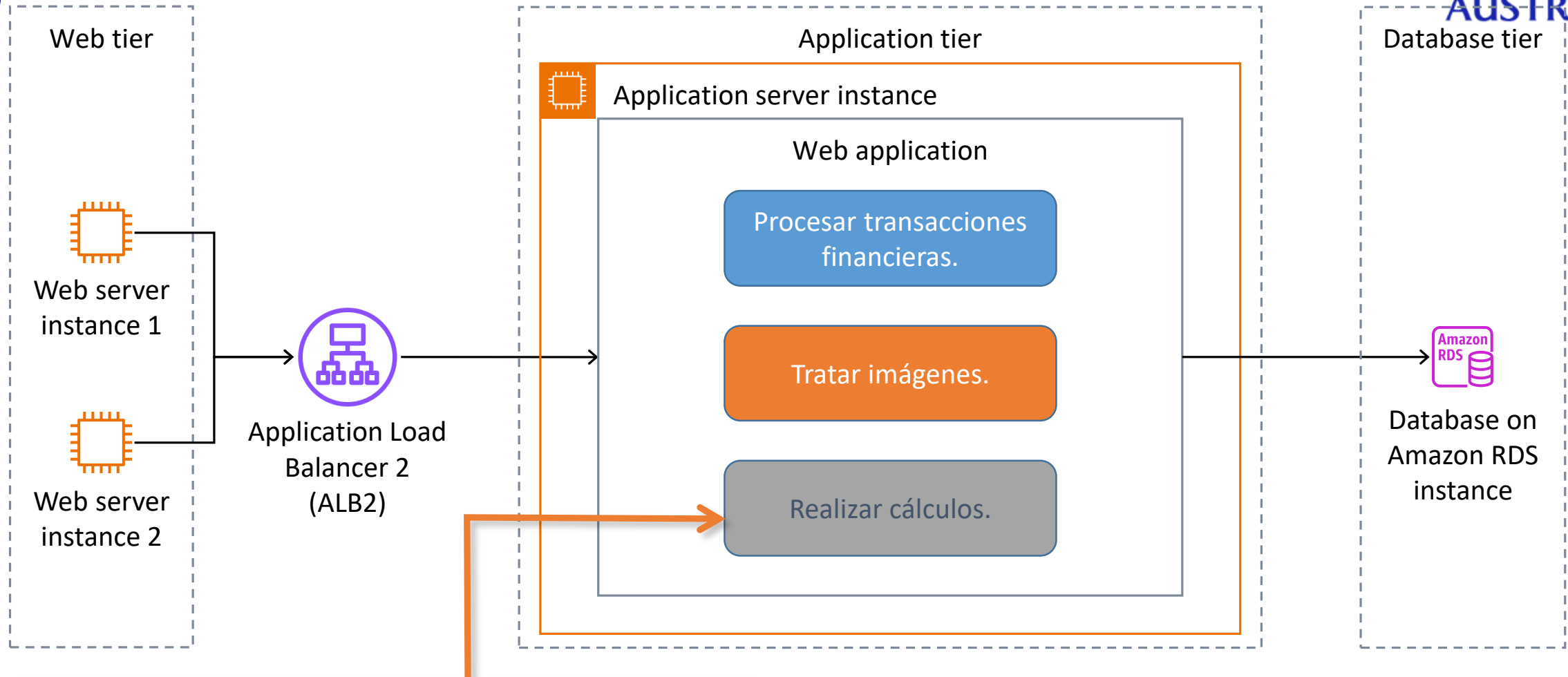


Cada nuevo servidor requiere múltiples conexiones que deben ser actualizadas en el código.

Acoplamiento débil



Acoplamiento fuerte dentro de una aplicación

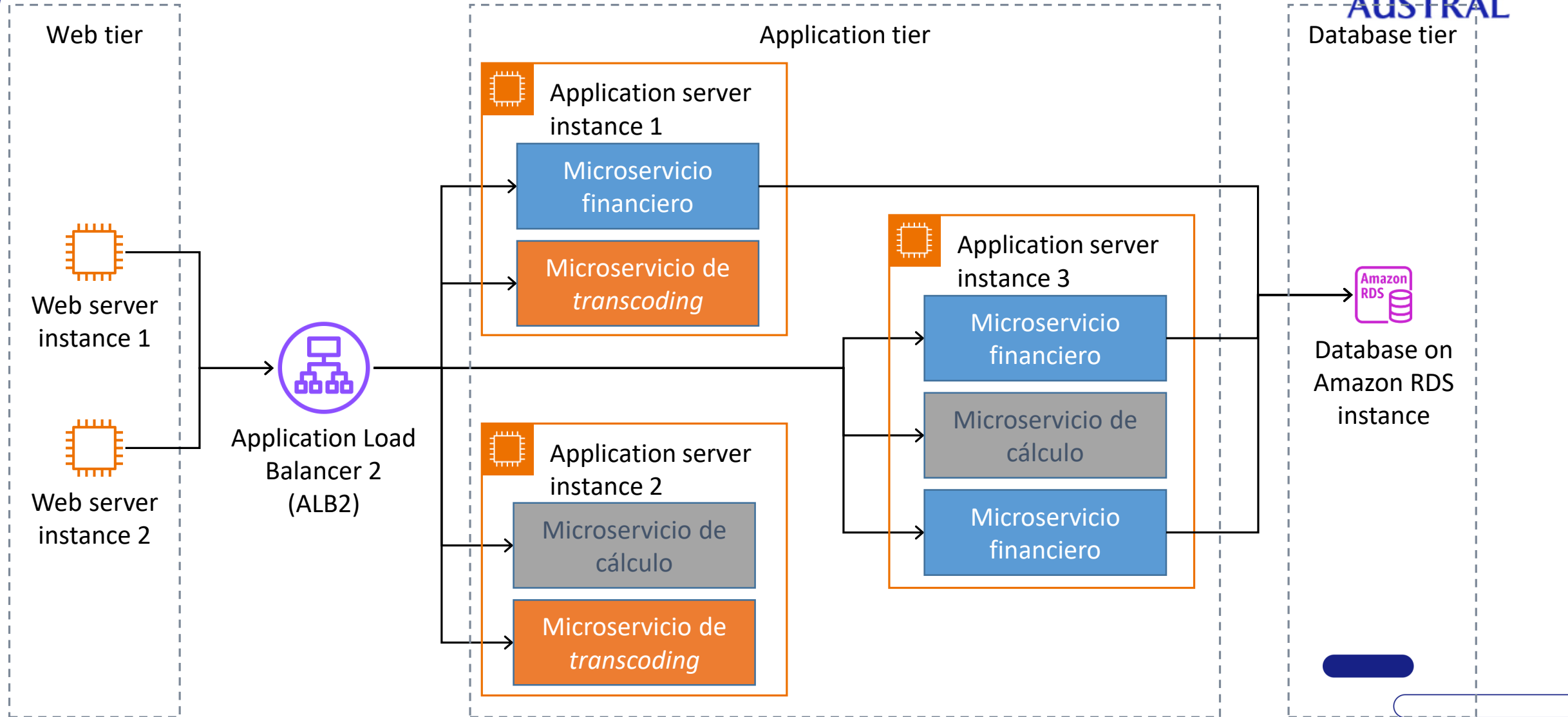


Desafío: la falla de una función afecta a toda la aplicación.

Acoplamiento débil: microservicios



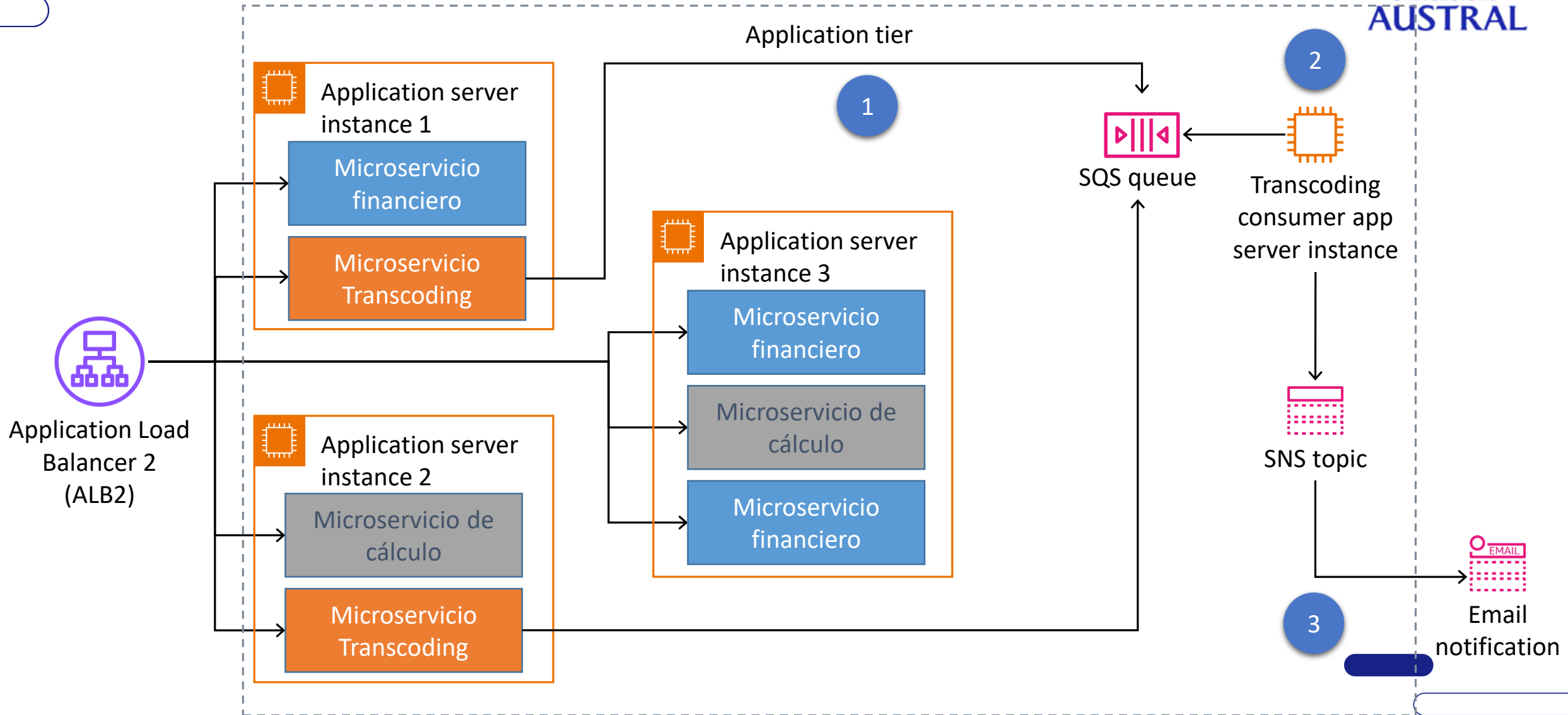
UNIVERSIDAD
AUSTRAL



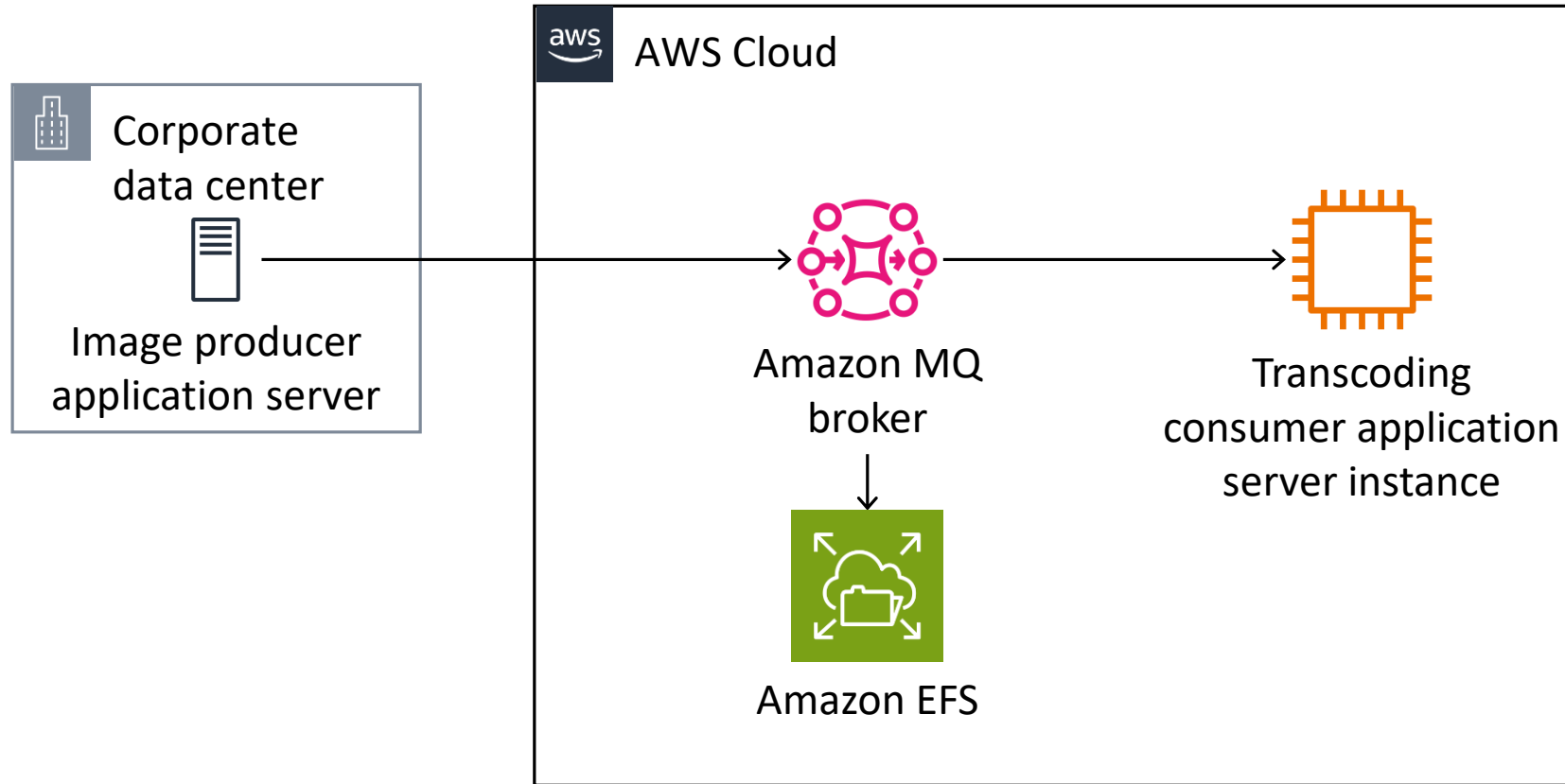
SQS y SNS



UNIVERSIDAD
AUSTRAL

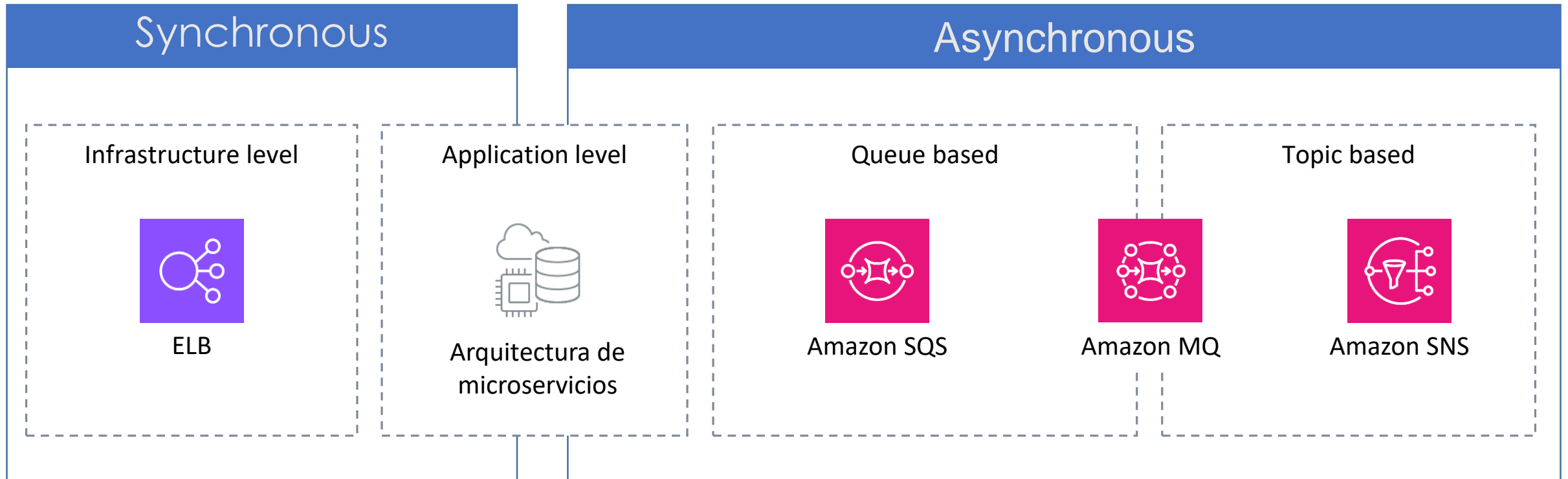


Amazon MQ



Amazon MQ

El acoplamiento débil resuelve problemas de integración y se puede lograr mediante soluciones sincrónicas y asincrónicas.



Resumen

Los sistemas fuertemente acoplados son difíciles de escalar, e introducen puntos únicos de falla y cuellos de botella.

Una arquitectura débilmente acoplada elimina las dependencias directas entre componentes relacionados y brinda escalabilidad y resiliencia.

Las soluciones débilmente acopladas dividen las capas de la infraestructura o las funciones de la aplicación. Típicamente, introducen un componente intermedio entre ellas.

Las soluciones débilmente acopladas pueden ser sincrónicas o asincrónicas:

- ELB es un ejemplo de solución sincrónica

- Amazon SQS, Amazon SNS, y Amazon MQ son ejemplos de soluciones asincrónicas.

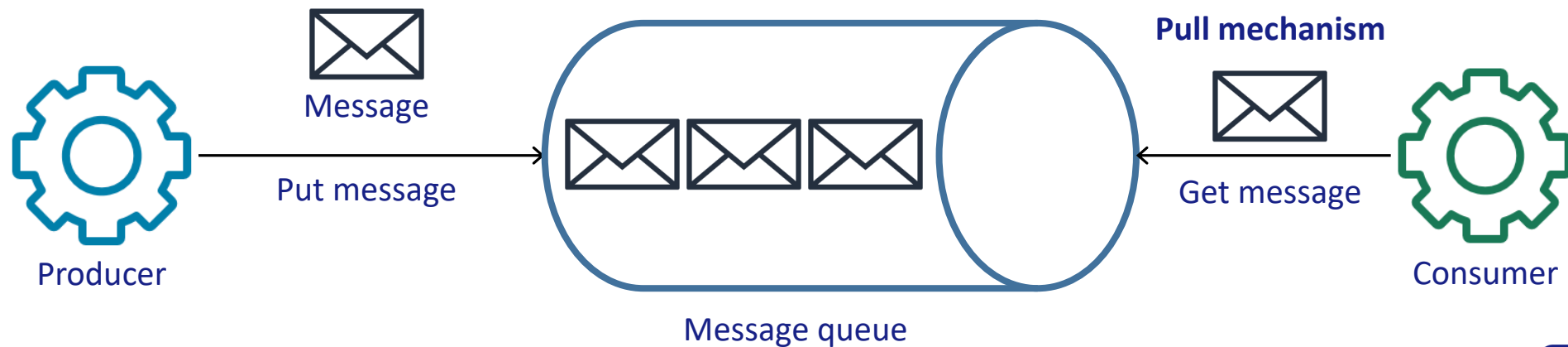
Amazon SQS

Amazon SQS

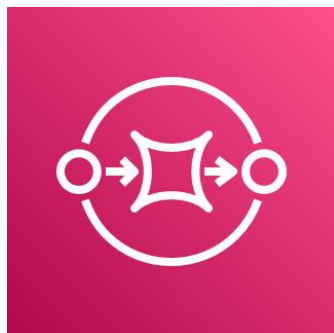
Se pueden desacoplar las aplicaciones de manera asincrónica, mediante mensajería punto a punto.

Una aplicación (“productora”) envía un mensaje a una única aplicación receptora (“consumidora”).

Se utiliza una cola de mensajes para desacoplar las aplicaciones.



Amazon SQS



Amazon
SQS

Servicio de colas de mensajería administrado por AWS

Ayuda a integrar y desacoplar sistemas de software distribuidos y componentes de aplicación.

Brinda funciones de colas de mensajería de alta disponibilidad, seguras y durables.

Tiene una consola de administración (AWS Management Console) y una API de web services

Amazon SQS

Características y beneficios

Administrado

No necesitamos administrar la infraestructura ni el software de mensajería.

Confiabilidad

Envía grandes cantidades de datos sin pérdida de mensajes.

Seguridad

Envío seguro de datos sensibles entre aplicaciones.

Escalabilidad

Escalamiento elástico basado en el uso.

Amazon SQS

Componentes



Mensaje

Longitud máxima: 256 KB.

Se mantiene en la cola hasta que es borrado explícitamente, o supera el tiempo de retención de la cola.



Cola

Amazon SQS ofrece dos tipos de cola: estándar y *first-in-first-out* (FIFO)

Los parámetros de la cola se pueden configurar. Por ejemplo:

- Período de retención de mensajes
- Timeout de visibilidad
- Tiempo de espera de recepción de mensajes (short polling vs. long polling)



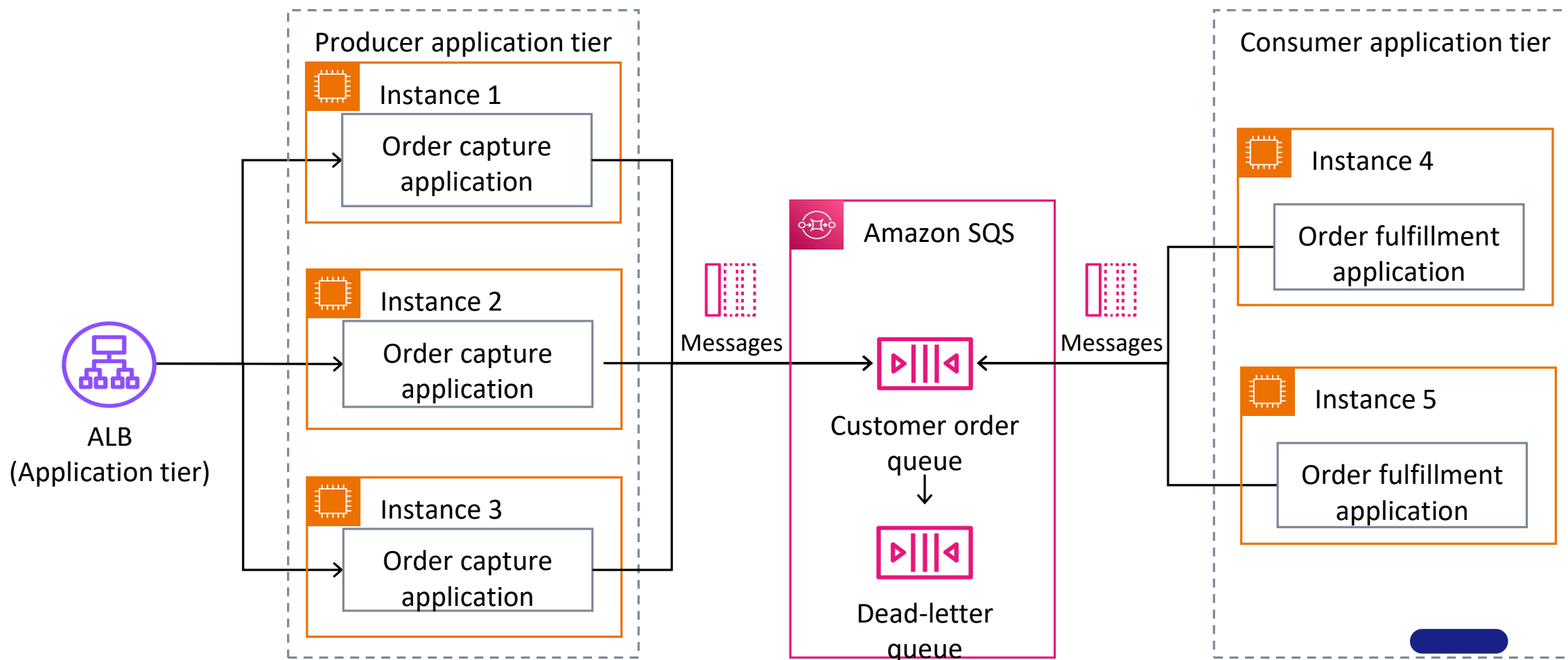
DLQ

Se puede asociar una DLQ (dead-letter queue) a cualquier cola.

Una DLQ almacena mensajes que no pudieron ser consumidos con éxito.

Amazon SQS

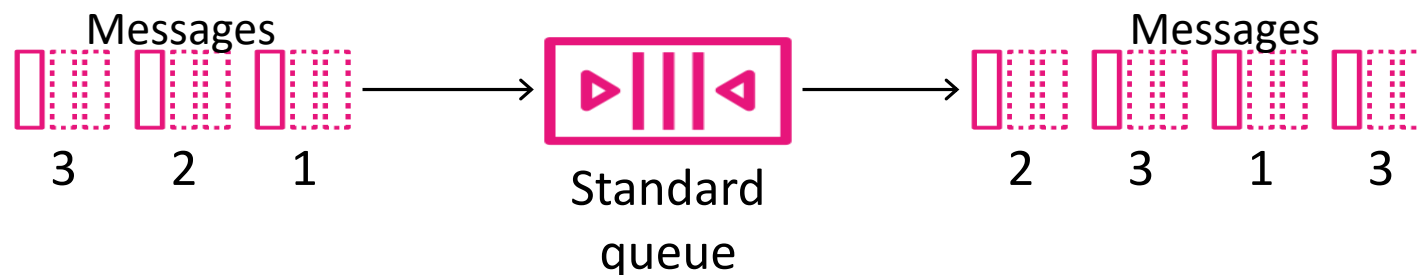
Ejemplo de desacoplamiento



Amazon SQS

Tipos de colas

Estándar



- Se entrega al menos una vez
- Se ordena según el mayor esfuerzo
- Throughput casi ilimitado

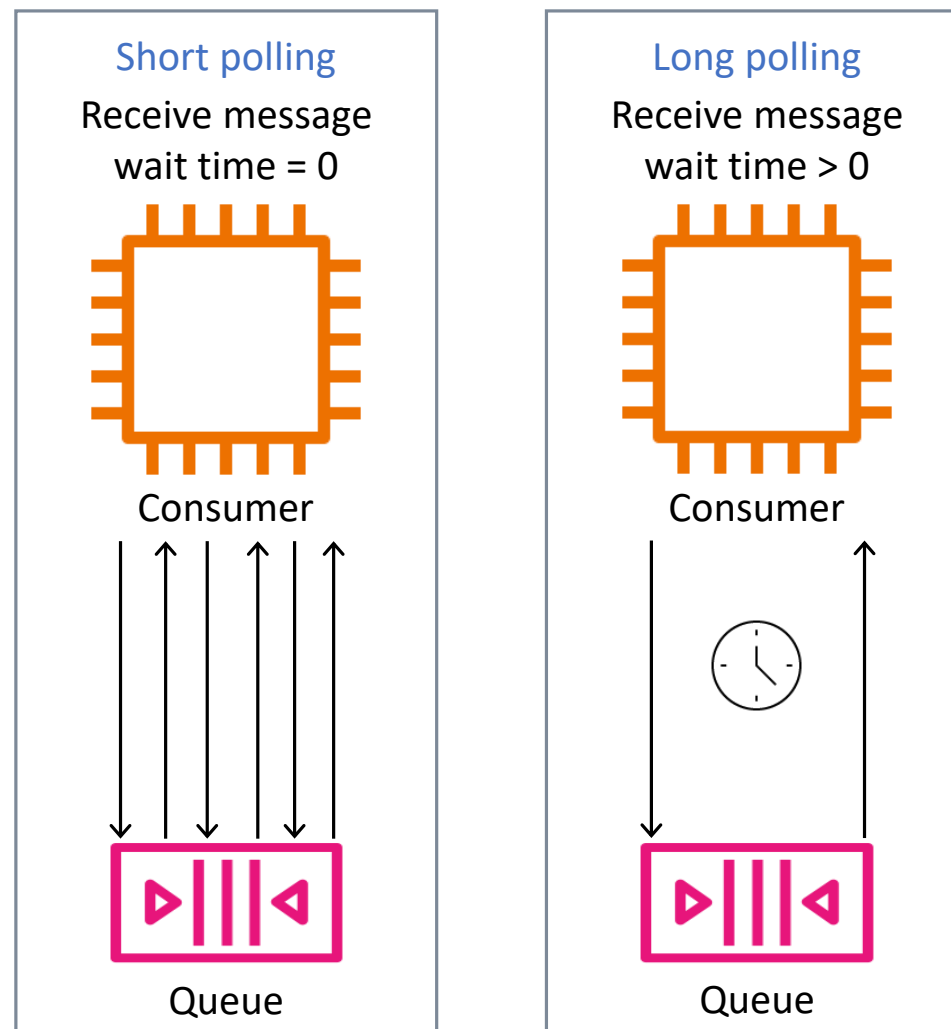
FIFO



- Entrega FIFO
- Se procesa exactamente una vez
- Throughput alto

Amazon SQS

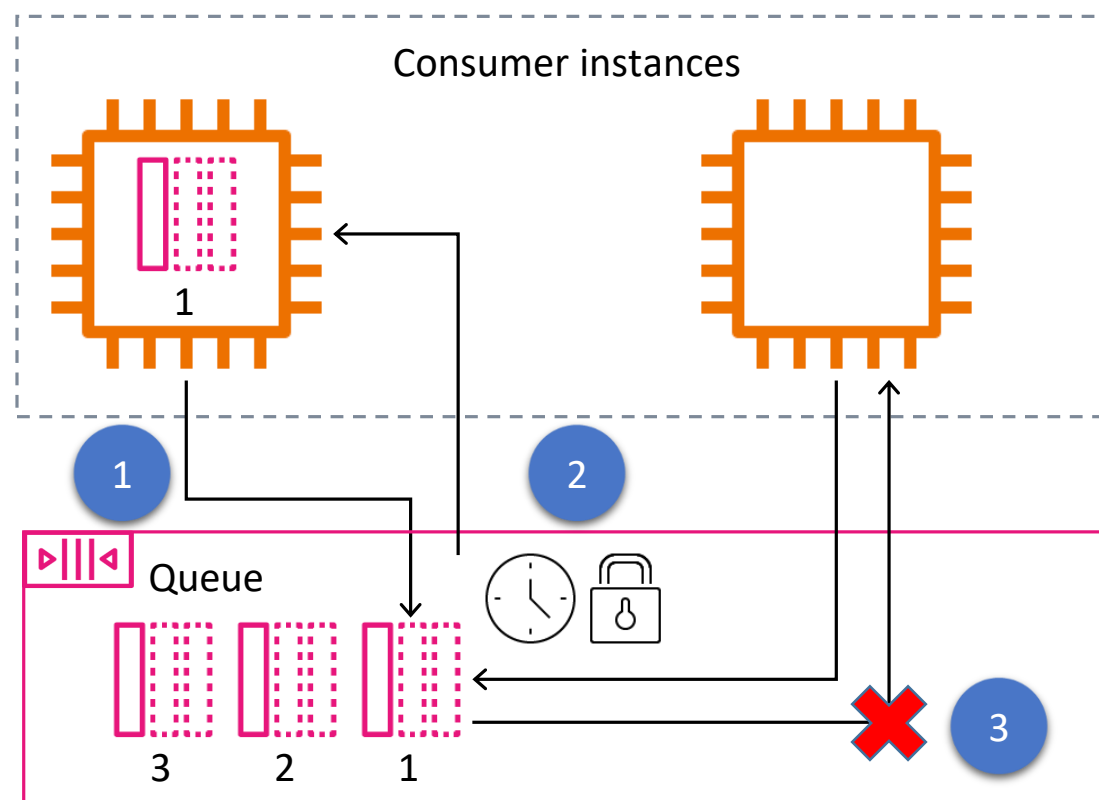
Tiempo de espera



Amazon SQS

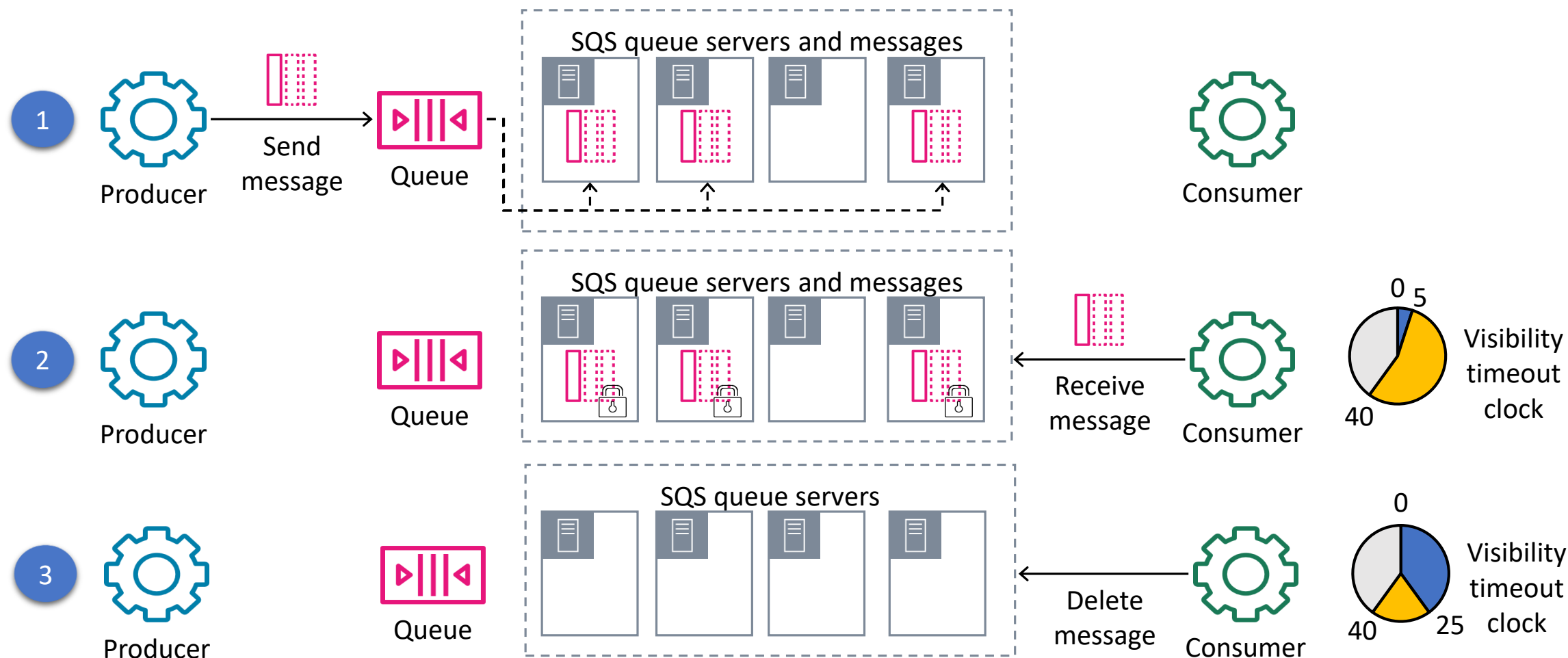
Visibilidad

Ajustar el time-out de visibilidad



Amazon SQS

Funcionamiento de las colas de mensajes



Amazon SQS

Casos de uso

Colas de trabajo

Desacoplar componentes de una aplicación distribuida que procesan la misma cantidad de trabajo a distinta velocidad

Buffers y operaciones batch

Crear un buffer para protegerse de picos de tráfico o acumular cargas para procesarlas en forma batch.

Reducir la carga de requerimientos

Encolar los requerimientos para separar las operaciones lentas de las funciones interactivas

Activar el auto scaling

Las colas SQS se pueden usar para ayudar a determinar la carga de una aplicación e invocar acciones de escalamiento.

Resumen

Amazon SQS es un servicio administrado de mensajería que puede ayudar a desacoplar componentes de una aplicación.

Amazon SQS admite colas estándar y FIFO.

Los mensajes no procesados se pueden mandar a una cola DLQ (dead-letter queue).

El long polling ayuda a reducir el costo de Amazon SQS porque disminuye la cantidad de respuestas vacías a un requerimiento.

Un “productor” envía un mensaje a la cola. Un “consumidor” lo procesa y lo borra de la cola, dentro del período de visibilidad.

Amazon SNS

Publicación y suscripción a servicios de mensajería

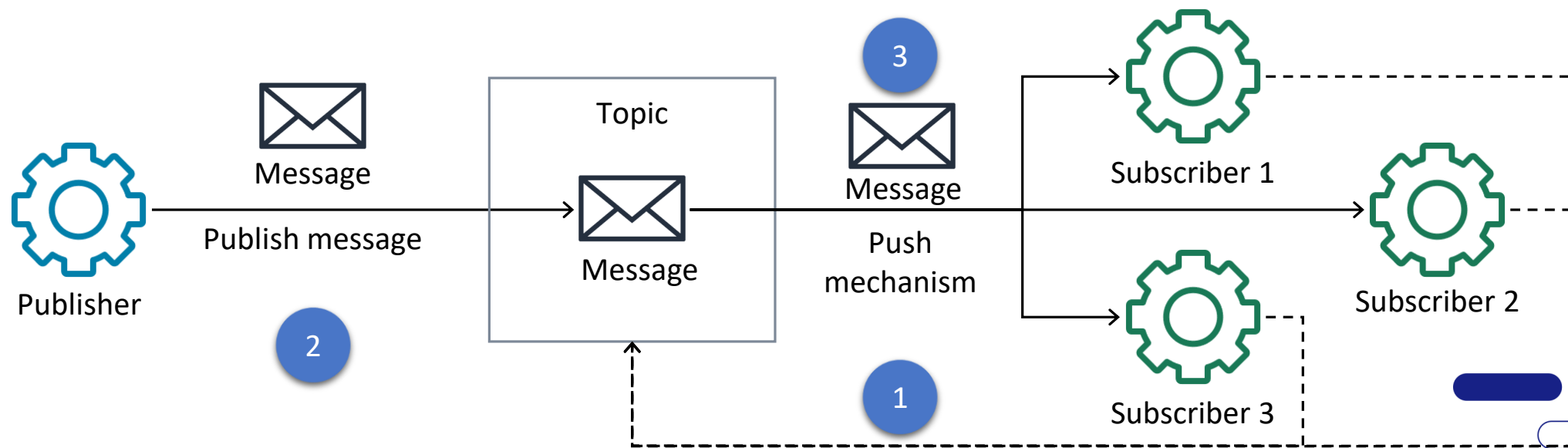
Se pueden desacoplar aplicaciones de manera asincrónica mediante mensajería de publicación/suscripción (pub/sub).

Se utiliza cuando la aplicación manda un mensaje a múltiples aplicaciones receptoras de las que no tiene conocimiento.

La aplicación que envía el mensaje es un *publisher*.

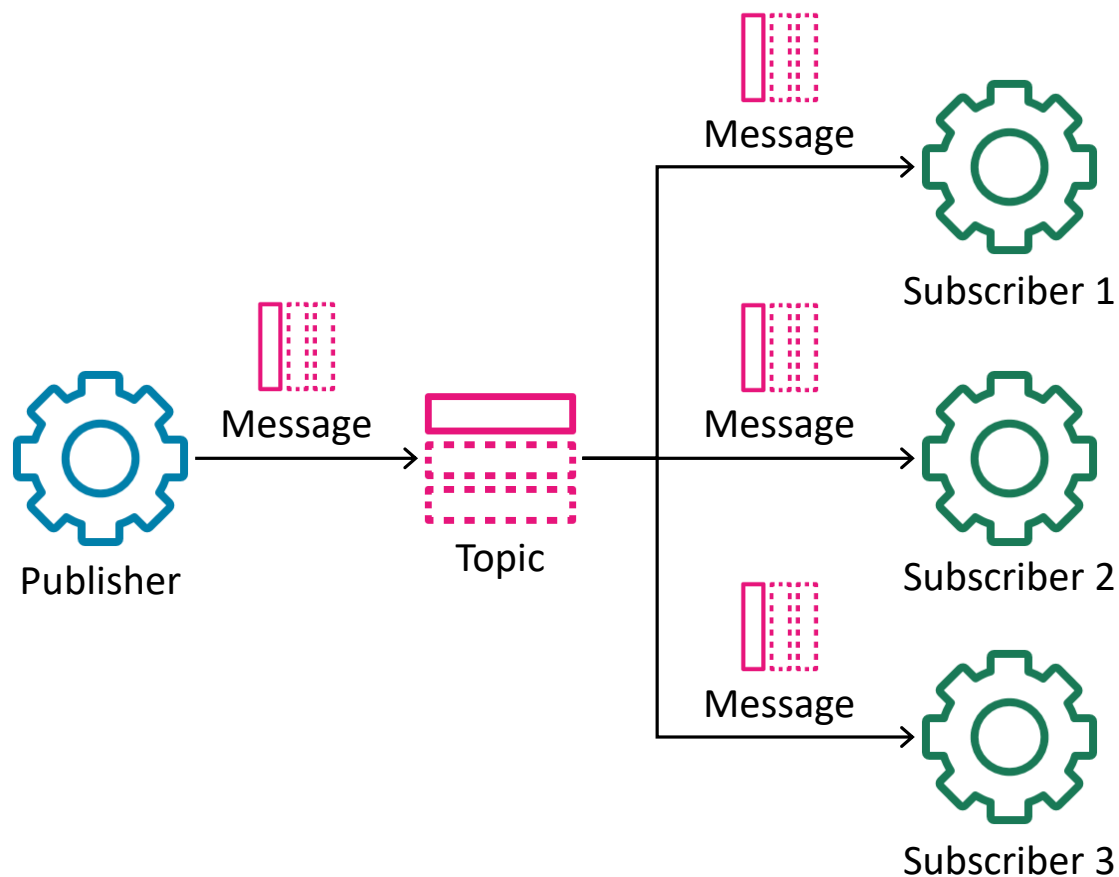
La aplicación receptora se llama *subscriber*.

La mensajería pub/sub usa un *tema (topic)* para desacoplar las aplicaciones.



Amazon SNS

Tipos de suscriptores



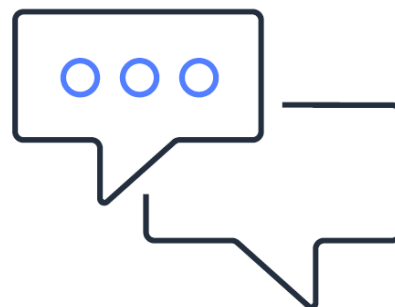
- Dirección de correo electrónico
- Receptor de mensajes de telefonía móvil
- Endpoint de notificaciones móviles de tipo *push*
- Endpoints HTTP o HTTPS
- Funciones AWS Lambda
- Cola de SQS
- Stream de Amazon Kinesis Data Firehose

Amazon SNS

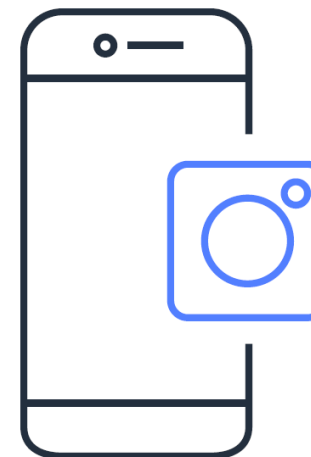
Casos de uso



Notificación de alertas
de aplicaciones y
sistemas.

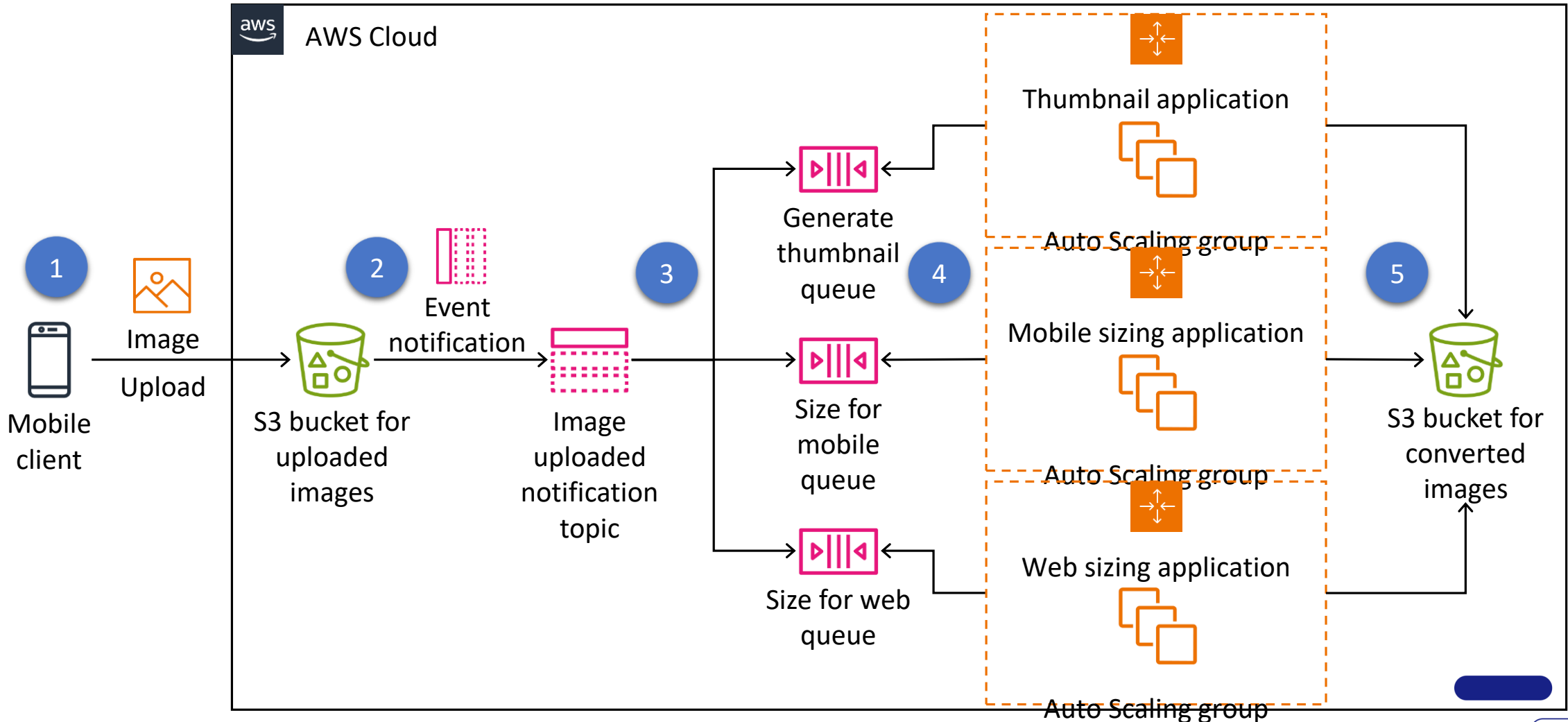


Notificación de correo
electrónico y mensaje
de texto.



Notificaciones *push* en
móviles

Uso de Amazon SQS con Amazon SNS



Amazon SNS

Consideraciones

Publicación de mensajes

Una publicación por vez
No se pueden recuperar mensajes

Entrega de mensajes

Se puede usar un tema estándar cuando no importa el orden de los mensajes.

Si hay que respetar un orden exacto, se usan temas FIFO.

La política de entrega de mensajes de un endpoint HTTP o HTTPS permite controlar el comportamiento de los reintentos.

Amazon SNS y Amazon SQS

Comparación

	Amazon SNS	Amazon SQS
Modelo de mensajería	Publisher-Subscriber	Producer-Consumer
Modelo de distribución	Uno a muchos	Uno a uno
Mecanismo de entrega	Push (pasivo)	Pull (activo)
Persistencia del mensaje	No	Yes

Resumen

Amazon SNS es un servicio web que se puede usar para configurar, operar y enviar notificaciones.

Amazon SNS aplica el paradigma de mensajería pub/sub.

Para usar Amazon SNS, se crea un tema, se *agregan* suscriptores y se publican mensajes.

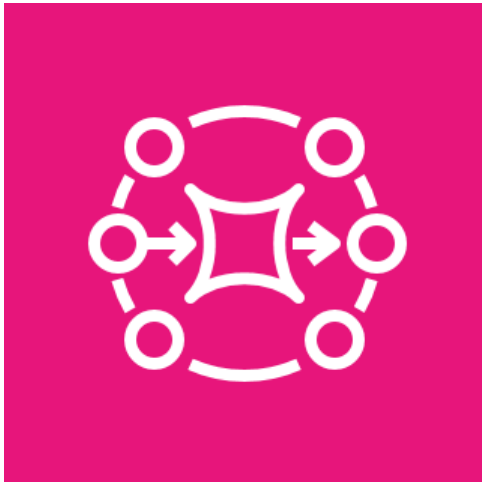
Se pueden usar temas para desacoplar los publicadores de los suscriptores y enviar mensajes a múltiples receptores a la vez.

Los servicios de AWS pueden publicar mensajes en un tema de SNS para invocar flujos manejados por eventos.

Amazon MQ

Desacoplamiento en soluciones híbridas

Amazon MQ



Amazon MQ

Servicio de mensajería administrado por AWS

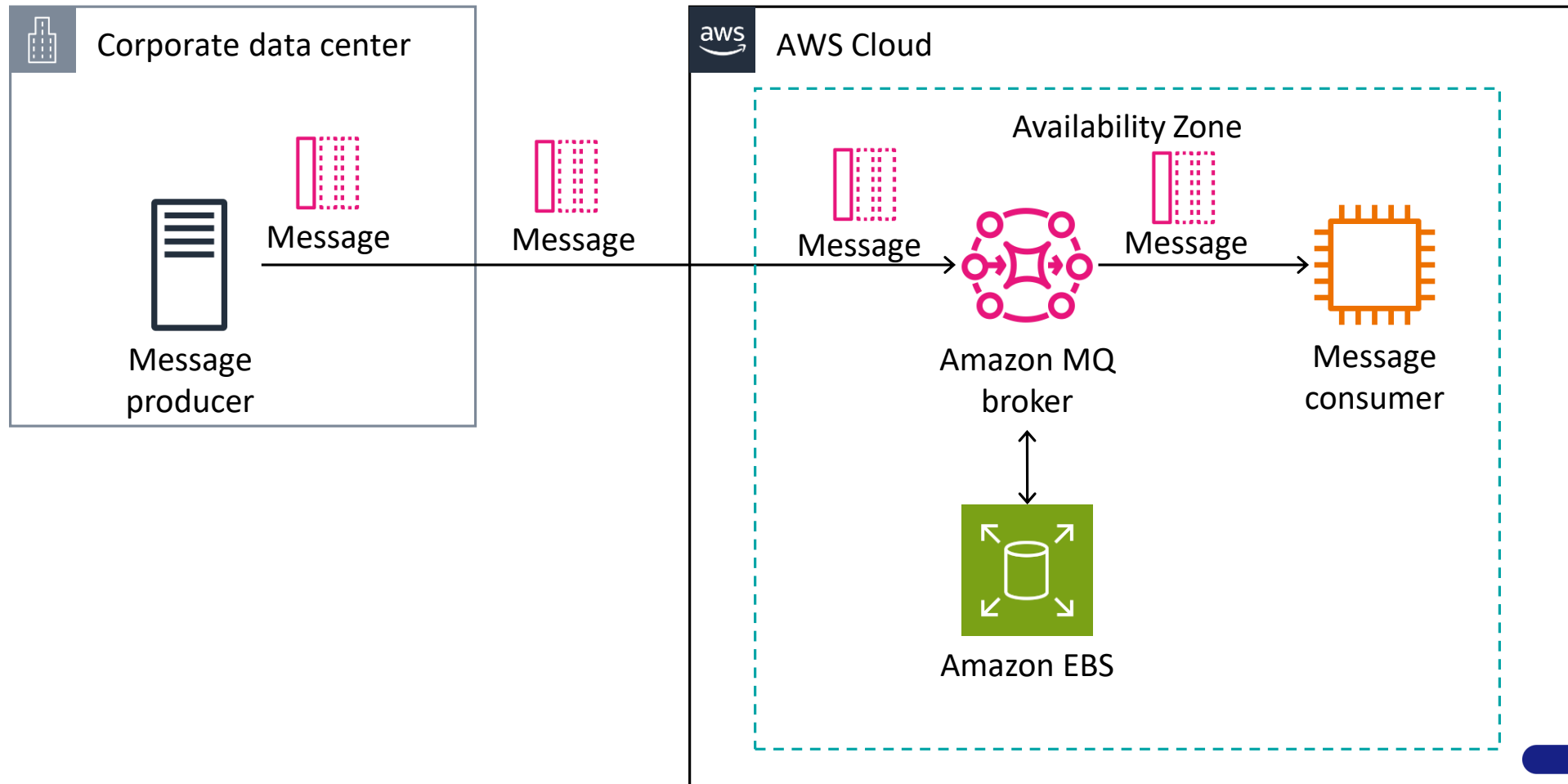
Facilita la creación, operación y gestión de una herramienta de mensajería Apache ActiveMQ o RabbitMQ en la nube de AWS

Brinda una solución basada en colas y en temas

Permite que las aplicaciones y los componentes se comuniquen mediante distintos lenguajes, sistemas operativos y protocolos de mensajería.

Amazon MQ

Caso de uso: instalación híbrida



Amazon SNS, SQS o MQ

¿Cuál es la mejor en cada caso?

	Amazon SQS	Amazon SNS	Amazon MQ
Aplicación	Aplicaciones centradas en nube	Aplicaciones nativas de nube	<ul style="list-style-type: none">• Aplicaciones híbridas• Migración de broker de mensajes
Modelo de mensajería	Producer-Consumer	Publisher-Subscriber	<ul style="list-style-type: none">• Producer-Consumer• Publisher-Subscriber
Programming API	Amazon SQS API	Amazon SNS API	APIs estándar de la industria
Modelo de precios	Por solicitud	Por solicitud	<ul style="list-style-type: none">• Por hora• Por GB

Resumen

Amazon MQ es un servicio gestionado que se puede usar para definir y operar en la nube las soluciones de mensajería Apache ActiveMQ y RabbitMQ.

Amazon MQ es compatible con los protocolos y estándares abiertos de mensajería.

Se puede usar para integrar ambientes on-premises con la nube.

Se puede usar para migrar soluciones de mensajería existentes a la nube.

Módulo 13

Resumen



Distinguir arquitecturas fuertemente y débilmente acopladas.

Identificar cómo funcionan y cuándo usar Amazon SQS y Amazon SNS.

Describir Amazon MQ.

Desacoplar cargas de trabajo usando Amazon SQS.

Módulo 13

Pregunta de práctica



A company must perform asynchronous processing and implement Amazon Simple Queue Service (Amazon SQS) as part of a decoupled architecture. The company wants to ensure that the number of empty responses from polling requests is kept to a minimum.

What should a solutions architect do to ensure that empty responses are reduced?

Identifiquemos las palabras o frases clave:

SQS queue

Empty responses kept to a minimum

Módulo 13

Pregunta de práctica

A company must perform asynchronous processing and implement Amazon Simple Queue Service (Amazon SQS) as part of a decoupled architecture. The company wants to ensure that the number of empty responses from polling requests is kept to a minimum.

What should a solutions architect do to ensure that empty responses are reduced?

Choice	Response
A	Increase the maximum message retention period for the queue.
B	Increase the maximum receives for the redrive policy for the queue.
C	Increase the default visibility timeout for the queue.
D	Increase the receive message wait time for the queue.

Módulo 13

Pregunta de práctica



A company must perform asynchronous processing and implement Amazon Simple Queue Service (Amazon SQS) as part of a decoupled architecture. The company wants to ensure that the number of empty responses from polling requests is kept to a minimum.

What should a solutions architect do to ensure that empty responses are reduced?

Choice	Response
D	Increase the receive message wait time for the queue.



Muchas gracias.

www.austral.edu.ar