



Diagramas de arquitectura

Estrategia sin servidor para precios dinámicos



Estrategia sin servidor para precios dinámicos: Diagramas de arquitectura

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon, de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

Precios dinámicos i

Estrategia sin servidor para precios dinámicos 1

Documentación adicional 2

Historial del diagrama 2

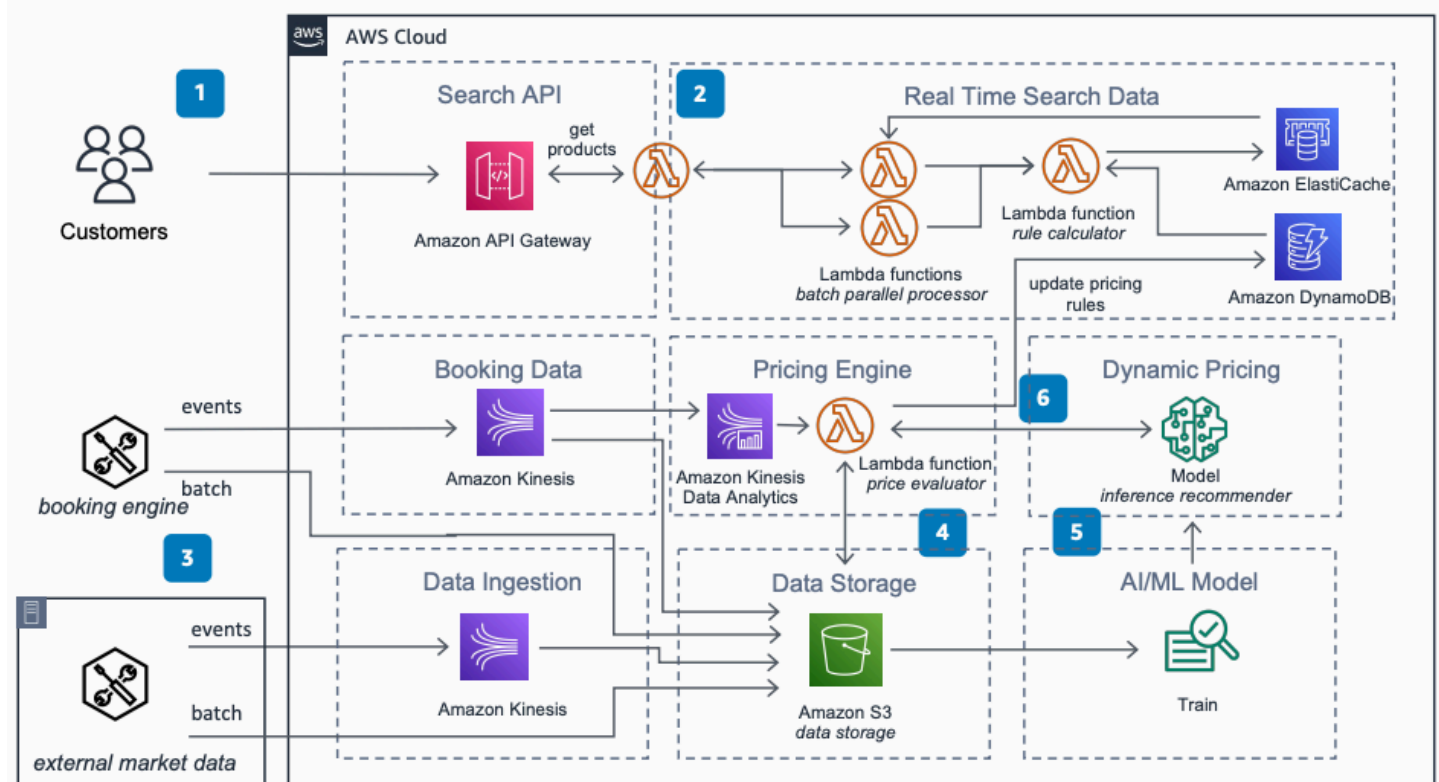
..... iv

Estrategia sin servidor para precios dinámicos

Fecha de publicación: 28 de junio de 2021 () [Historial del diagrama](#)

Esta arquitectura muestra una estrategia sin servidor para precios dinámicos. En el caso de las reservas basadas en intervalos de fechas, puedes crear precios en paralelo y calcularlos de forma dinámica en función de factores como la fecha, la duración y el número de personas. A continuación, muestra productos comparables ordenados con tiempos de respuesta inferiores a un segundo y actualiza los precios en función de las condiciones actuales del mercado.

Estrategia sin servidor para precios dinámicos



Los siguientes pasos describen la arquitectura:

1. Introduce los datos de búsqueda para obtener una lista ordenada de productos filtrados por precio y otros atributos. Los datos se envían a través de [Amazon API Gateway](#), que llama a varias [AWS Lambda](#) funciones para procesar las solicitudes de precios en paralelo y así poder consultarlos en tiempo real.

2. Las funciones Lambda del procesador paralelo por lotes invocan la función de calculadora de reglas para obtener las reglas de producto más recientes de [Amazon DynamoDB](#). El producto se calcula en función del identificador de búsqueda y se almacena en Amazon. [ElastiCache](#). Una vez completadas las funciones por lotes, todos los resultados del identificador de búsqueda se devuelven de la caché.
3. Los datos históricos del motor de reservas y los datos del mercado externo proporcionan información para que el motor de precios tenga datos suficientes para actualizar los precios en tiempo real.
4. Los eventos del motor de reservas se capturan y supervisan en [Amazon Kinesis](#). Amazon Kinesis Data Analytics consulta la transmisión de datos y activa la función Lambda del evaluador de precios si se detecta un cambio. Los eventos se almacenan en [Amazon Simple Storage Service](#) con los correspondientes datos históricos de lotes.
5. Un modelo de [Amazon SageMaker AI](#) Train utiliza datos históricos de reservas y datos históricos aumentados para preparar el modelo a la hora de recomendar precios.
6. Al utilizar un modelo de recomendación por inferencia de SageMaker inteligencia artificial, el motor de precios recibe nuevas recomendaciones de precios en función de las condiciones del mercado.

Documentación adicional

Para obtener información adicional, consulta los siguientes recursos:

- [AWS Iconos de arquitectura](#)
- [AWS Centro de arquitectura](#)
- [AWS Well-Architected](#)

Historial del diagrama

Para recibir notificaciones sobre las actualizaciones de este diagrama de arquitectura de referencia, suscríbase a la fuente RSS.

Cambio	Descripción	Fecha
Publicación inicial	El diagrama de arquitectura de referencia se publicó por primera vez.	28 de junio de 2021

 Note

Para suscribirse a las actualizaciones de RSS, debe tener un complemento de RSS habilitado para el navegador que esté utilizando.

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.