



Architecture diagrams for retail

# Improve Demand Forecasting to Boost Sales on Amazon.com



# **Improve Demand Forecasting to Boost Sales on Amazon.com:**

## **Architecture diagrams for retail**

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

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

# Table of Contents

**Demand forecasting ..... i**

    Demand forecasting diagram ..... 1

    Further reading ..... 2

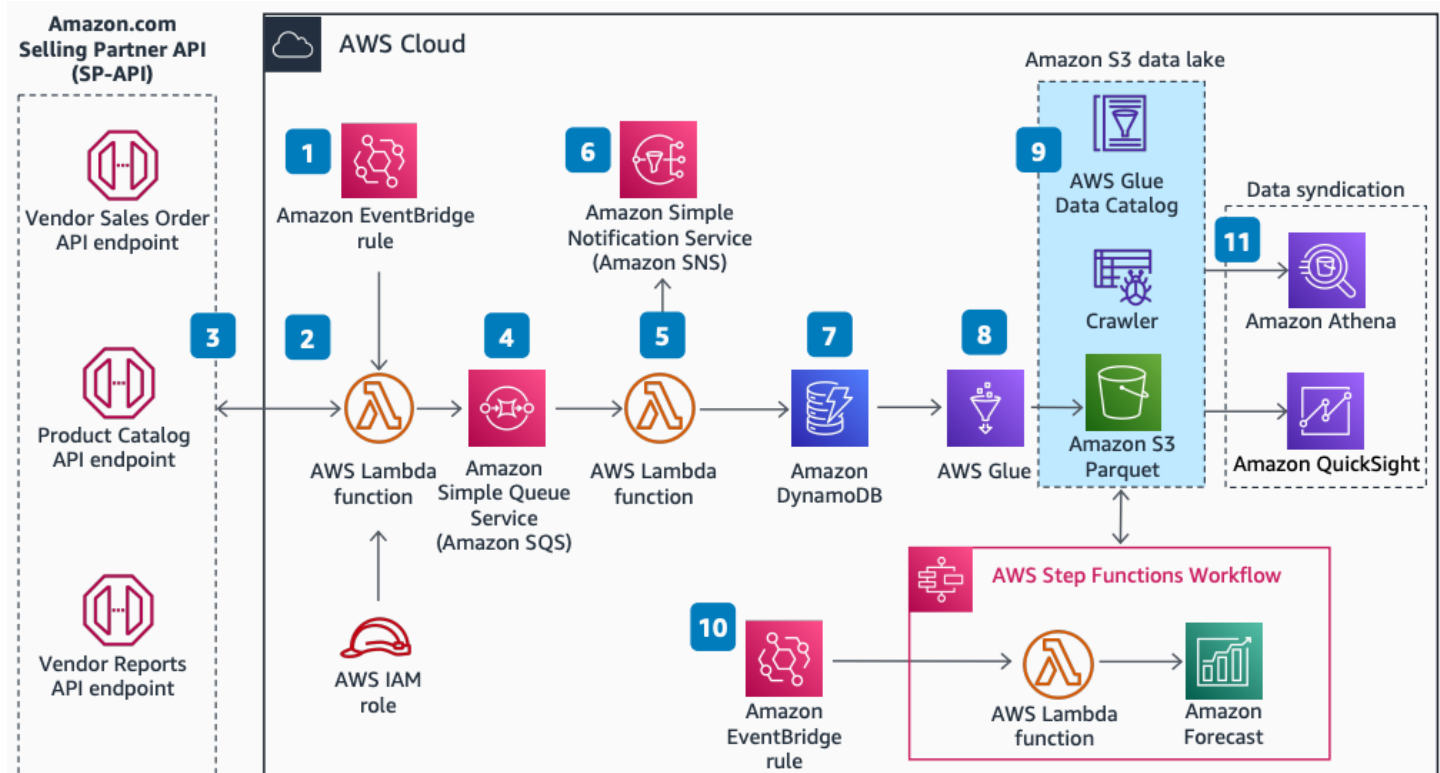
    Diagram history ..... 2

# Improve Demand Forecasting to Boost Sales on Amazon.com

Publication date: April 5, 2022 ([Diagram history](#))

With this architecture, you can understand prior demand trends and anticipate future demand. Companies that sell products on Amazon.com as first-party vendors want to improve On-Time-In-Full (OTIF) metrics and maximize revenue. You programmatically source vendor performance data from the Amazon.com Selling Partner API and use machine learning (ML) to create future demand predictions.

## Demand forecasting diagram



The following steps describe the architecture:

1. [Amazon EventBridge](#) initiates scheduled tasks that keep the data fresh with controlled latency.
2. [Lambda](#) makes API calls to the Amazon Selling Partner API. An AWS Identity and Access Management (IAM) role attached to the Amazon Vendor Central account entitles access.

3. The Selling Partner API authenticates the calling API and uses the IAM role to authorize access to a specific set of vendor codes.
4. Some API responses are stored in [Amazon Simple Queue Service](#) as part of a decoupled microservices architecture.
5. Lambda consumes messages from the Amazon SQS queue. It transforms, persists, and creates notifications.
6. [Amazon Simple Notification Service](#) receives notifications on data conditions such as product catalog changes.
7. [Amazon DynamoDB](#) acts as an intermediate database. It accepts create, read, update, and delete (CRUD) operations while maintaining data integrity.
8. A scheduled [AWS Glue](#) job publishes data to a serverless data lake for long-term persistence.
9. [Amazon Simple Storage Service](#) is the persistence tier for the serverless data lake. It serves time-series data at any scale.
- 10 EventBridge triggers a recurring [Amazon Forecast](#) routine. [AWS Step Functions](#) manages the workflow that sources demand history to generate future predictions.
- 11 Amazon S3 data is syndicated to consuming applications. [Amazon Quick Sight](#) and [Amazon Athena](#) provide business intelligence and SQL capabilities.

## Further reading

For additional information, see the following resources:

- [AWS Architecture Icons](#)
- [AWS Architecture Center](#)
- [AWS Well-Architected](#)

## Diagram history

To receive updates about this reference architecture diagram, subscribe to the RSS feed.

| Change                              | Description                                     | Date          |
|-------------------------------------|-------------------------------------------------|---------------|
| <a href="#">Initial publication</a> | Reference architecture diagram first published. | April 5, 2022 |

** RSS subscription requirement**

To subscribe to RSS updates, you must have an RSS plugin enabled for the browser you are using.