



Architecture diagrams for hospitality

Connected Lodging Properties Using IoT and AI/ML



Connected Lodging Properties Using IoT and AI/ML: Architecture diagrams for hospitality

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

Connected lodging IoT **i**

Connected lodging properties diagram **1**

Further reading **2**

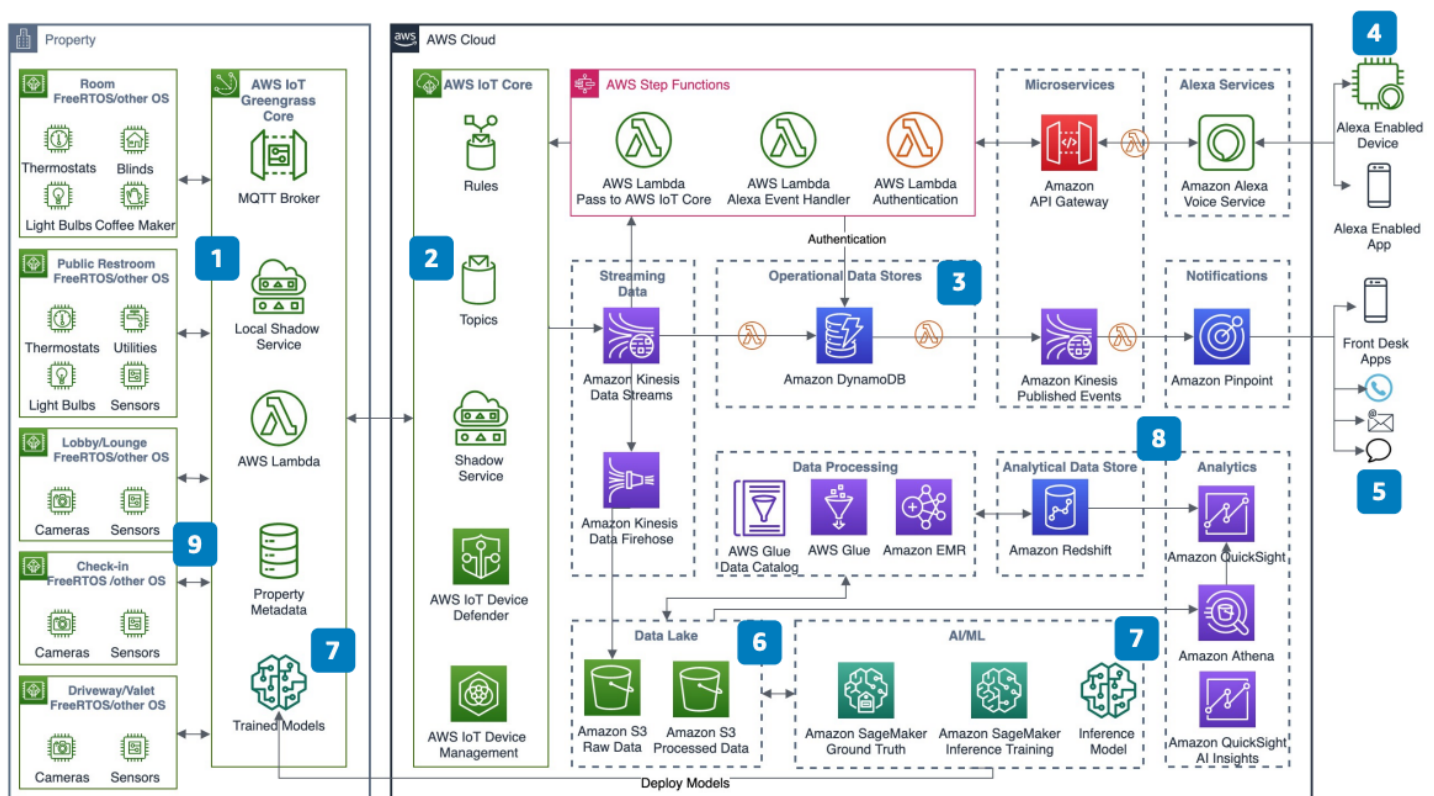
Diagram history **2**

Connected Lodging Properties Using IoT and AI/ML

Publication date: August 28, 2020 ([Diagram history](#))

With this architecture, you can build smart, connected lodging properties. Use IoT and artificial intelligence/machine learning (AI/ML) to provide touchless room personalization. The solution monitors queue depths, tracks foot and vehicle traffic, and maintains sanitation compliance. It uses [AWS IoT Greengrass](#) at the edge and [Amazon SageMaker AI](#) for inference models.

Connected lodging properties diagram



The following steps describe the architecture:

1. Use [AWS IoT Greengrass](#) Core to connect, publish, and subscribe to data. Communicate by using the Message Queuing Telemetry Transport (MQTT) protocol with IoT devices that run on FreeRTOS.
2. Use [AWS IoT Core](#) to maintain device shadows for all IoT devices. Connect to the AWS Cloud, manage devices, update over-the-air (OTA), and secure the devices.

3. Use purpose-built databases such as [Amazon DynamoDB](#) and serverless architecture. Store events, deliver microservices, and generate events for an operational data store.
4. Use Alexa Voice Service to personalize guest rooms with voice-enabled controls.
5. Use [Amazon Pinpoint](#) to deliver alerts to multiple channels. Send notifications for operational events and property management.
6. Build a data lake to store raw data and create curated datasets in [Amazon Simple Storage Service](#). Use [AWS Glue](#) and [Amazon EMR](#) for data processing.
7. Use SageMaker AI to build, train, and deploy inference models. (Optional) Deploy edge models on AWS IoT Greengrass Core.
8. Use [Amazon Redshift](#), [Amazon Athena](#), and [Amazon Quick Sight](#) for analytics. (Optional) Build data marts in Amazon Redshift for heavily used analytics.
9. Use social distancing and queue depth management solutions for compliance and enhanced guest experience.

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
Initial publication	Reference architecture diagram first published.	August 28, 2020

** RSS subscription requirement**

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