



Architecture Diagrams

Spatial Computing 3D Content Management



Spatial Computing 3D Content Management: Architecture Diagrams

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

3D content management

Spatial computing 3D content management diagram

Further reading

Diagram history

i

1

2

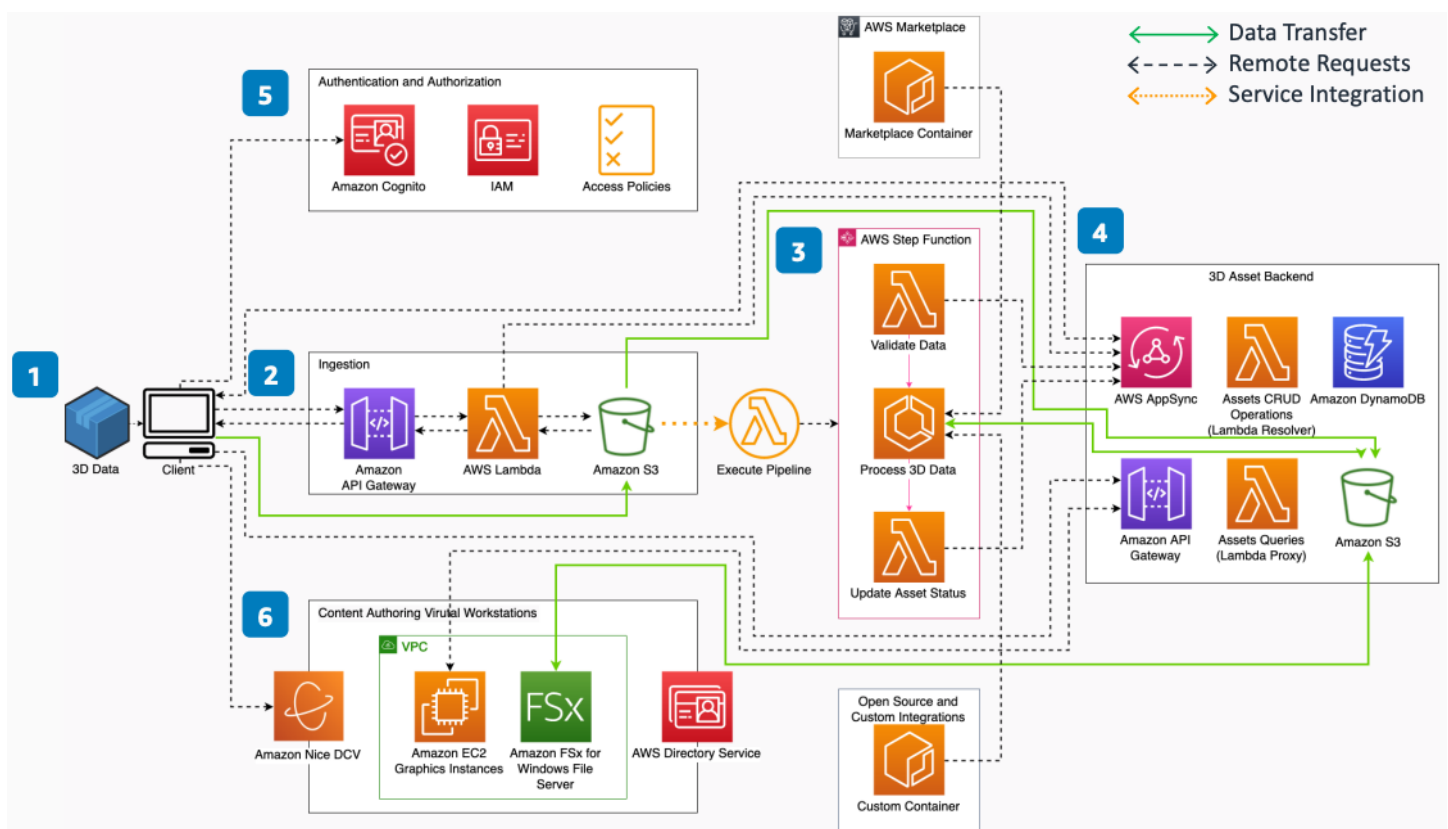
2

Spatial Computing 3D Content Management

Publication date: **August 27, 2021** ([Diagram history](#))

With this architecture, you can create a single source of truth for 3D assets. Automate ingestion, transformation, and versioning of 3D assets. Process computer-aided design (CAD), photogrammetry, light detection and ranging (LiDAR), or other existing 3D data. The solution uses [Amazon Simple Storage Service](#) (Amazon S3), [Amazon DynamoDB](#), [AWS Lambda](#), and [AWS Step Functions](#).

Spatial computing 3D content management diagram



The following steps describe the architecture:

1. With a graphical user interface (GUI) or command-line interface (CLI) client, users upload 3D data, make queries against existing data, and connect to virtual workstations.
2. On initial upload, the system creates an asset entry in a DynamoDB table. Uploading source data to Amazon S3 triggers an event that invokes a Lambda function to initiate a Step Functions workflow.

3. The workflow has three states: inspect the validity of uploaded source data, process the data with AWS Marketplace or open source tools, and record the results in the asset database.
4. Use [AWS AppSync](#) to manage create, read, write, and delete operations to asset entries with GraphQL mutations. Consuming applications subscribe to GraphQL subscriptions for real-time updates. Implement a REST API with [Amazon API Gateway](#) for request and response communication.
5. [Amazon Cognito](#) handles authentication. [AWS Identity and Access Management](#) user group policies control fine-grained access over asset resources.
6. Virtual workstation desktops stream over the network with [Amazon EC2](#) graphics instances backed by [Amazon DCV](#) for manual content authoring.

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 27, 2021

RSS subscription requirement

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