AWS re:Invent

NOV. 29 – DEC. 3, 2021 | LAS VEGAS, NV

IOT317

Building digital twins using AWS services

Raj Devnath
Sr. PMT - IoT Analytics
AWS

Shyam Kariat
Manufacturing Innovation Lead
AWS

Agenda

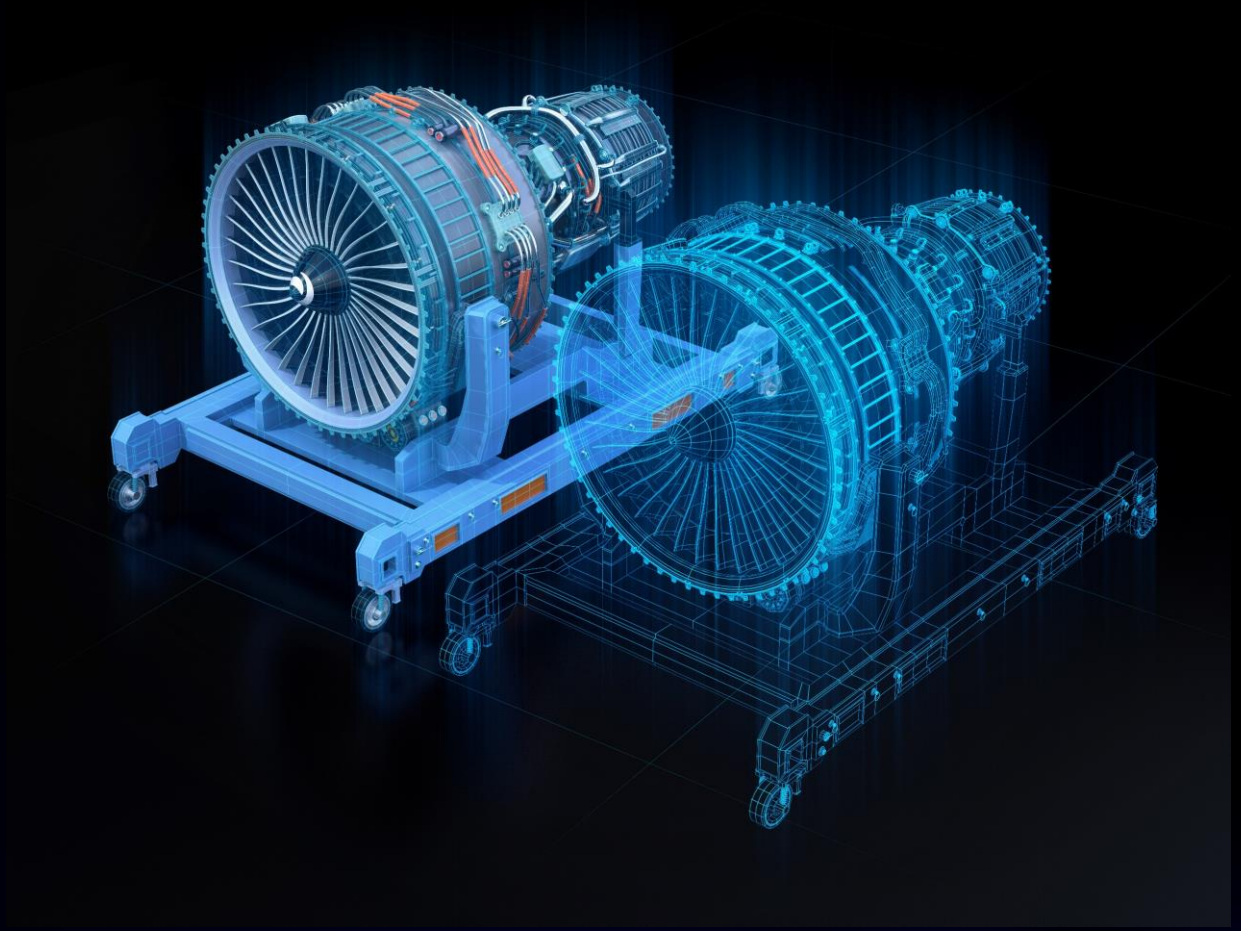
1. What is a Digital Twin?
2. Demo
3. Overall Workflow
4. Steps to build Digital twin

What is a Digital Twin

A **living digital representation** of a physical system that is

dynamically updated with data to mimic the true structure, state and behavior of the physical system and

used to **drive business outcomes**.

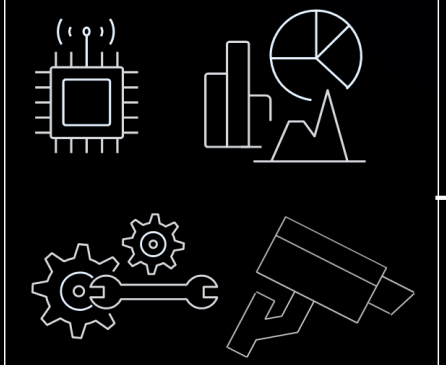


Common challenges

1. Model physical systems and express relationships between the parts
2. Bring together data from various sources without creating yet another data store
3. Add insights such as simulations using modeled data
4. Create compelling visualization using 3D
5. Publish web applications to use by operators

Workflow

1. Workspace Creation



2. Entity/Component Modeling

3. Add streaming insights

4. 3D model and Scene building

5. Create web applications

Step 1: Workspace creation

Workspace holds all resources needed to create digital twins

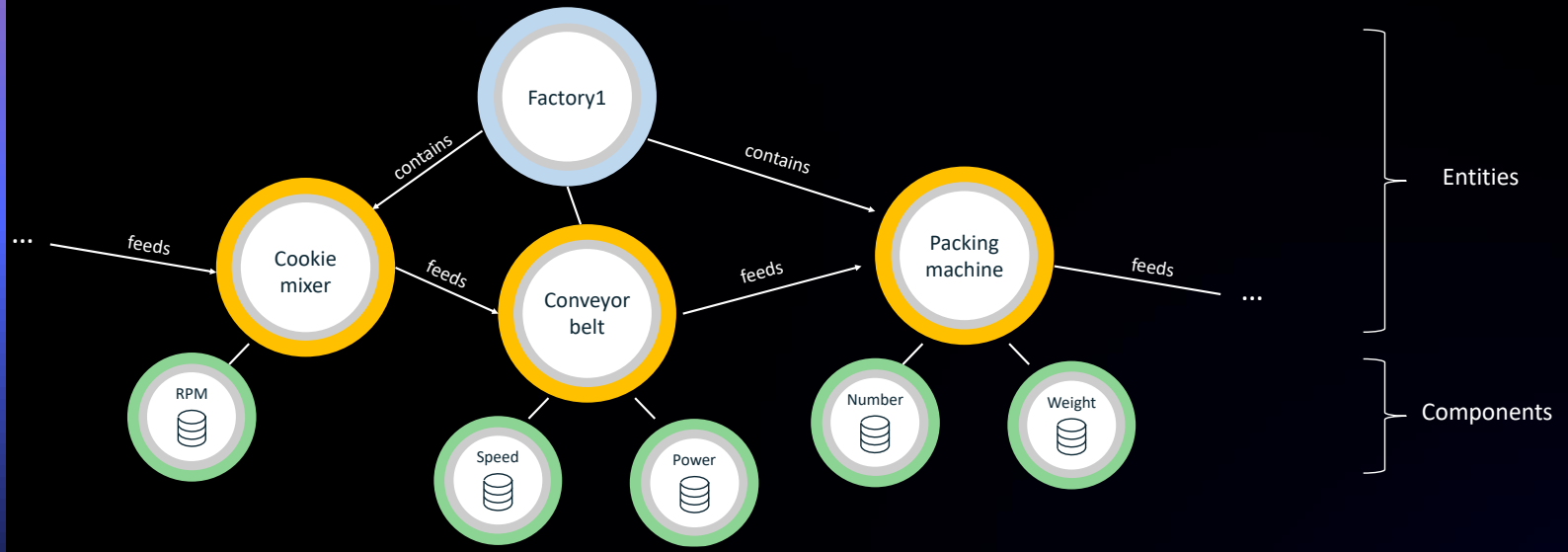
- Model of your physical systems
- Connectors to data stores
- Streaming applications
- Resources: 2D/3D files, documents, etc.
- Tags

It can also serve as a security boundary

Step 2: Modeling

Models have entities and components

- Entities: digital representations of physical system
- Components: Connections to the data stores



Express custom relationships between entities

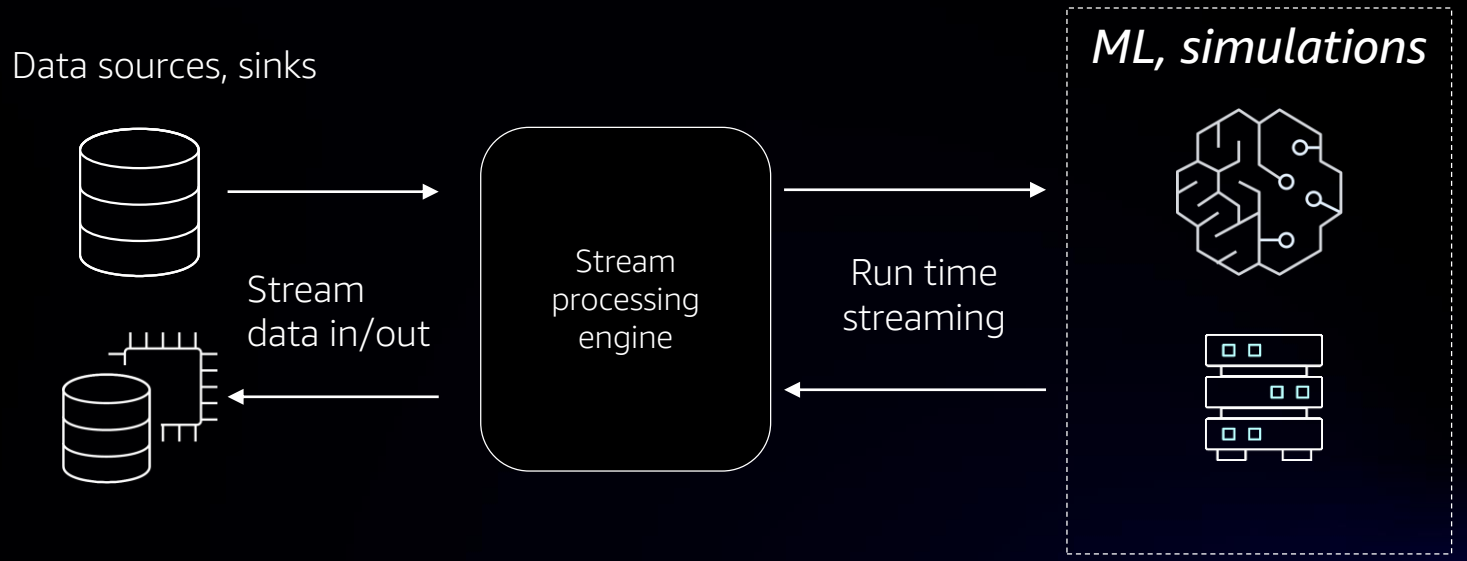
Use components: Avoid building yet another data store

Bring in schema from data stores as entity properties

Author your own components

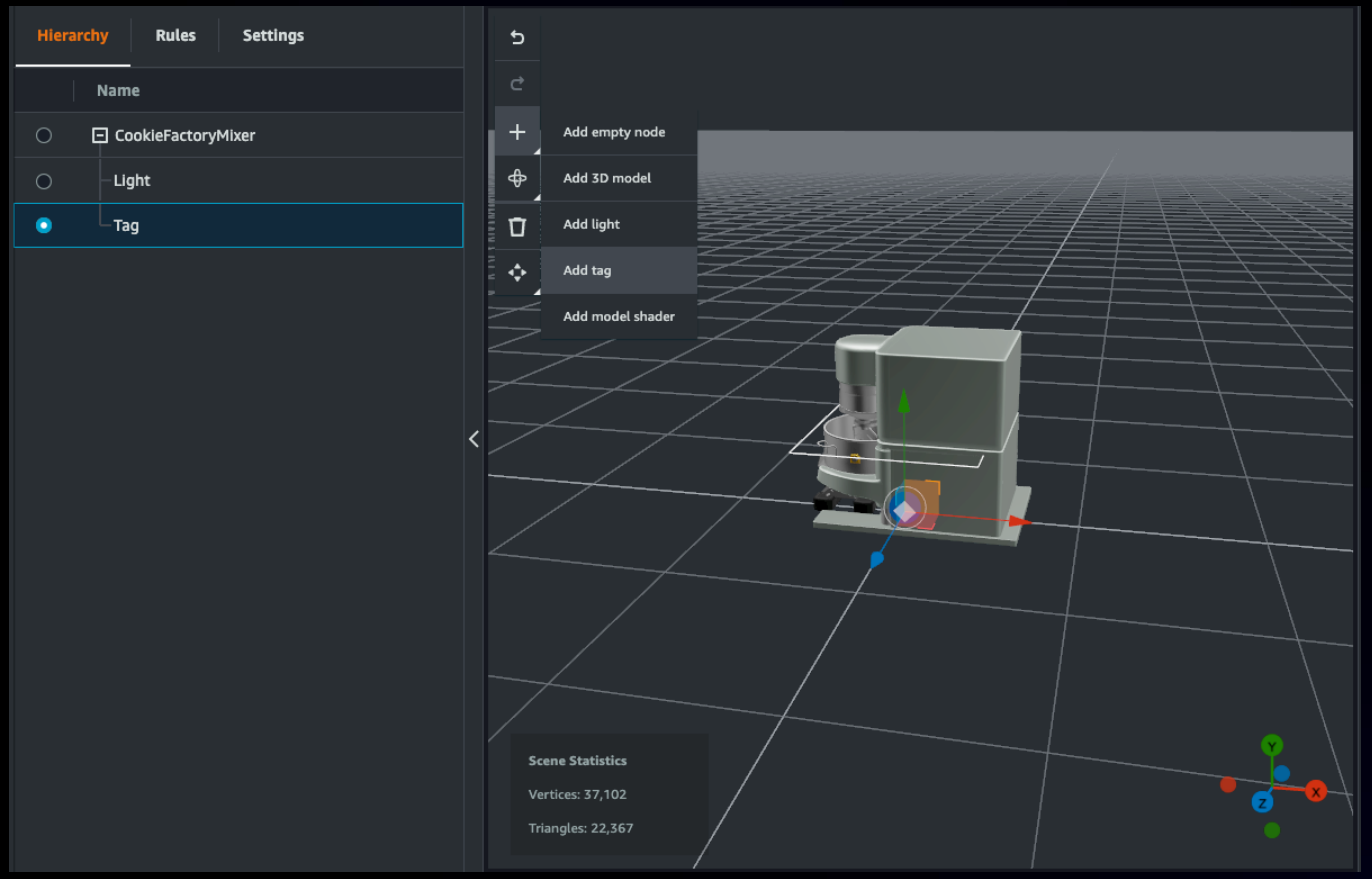
Step 3: Streaming insights

- Enrich your time series data
- Add insights using analytical engines (ML, simulations)
- Store the results back to your data store



Step 4: Create visual scenes

Scenes are visual representation of your physical system



Import 3D models via standard GLTF files

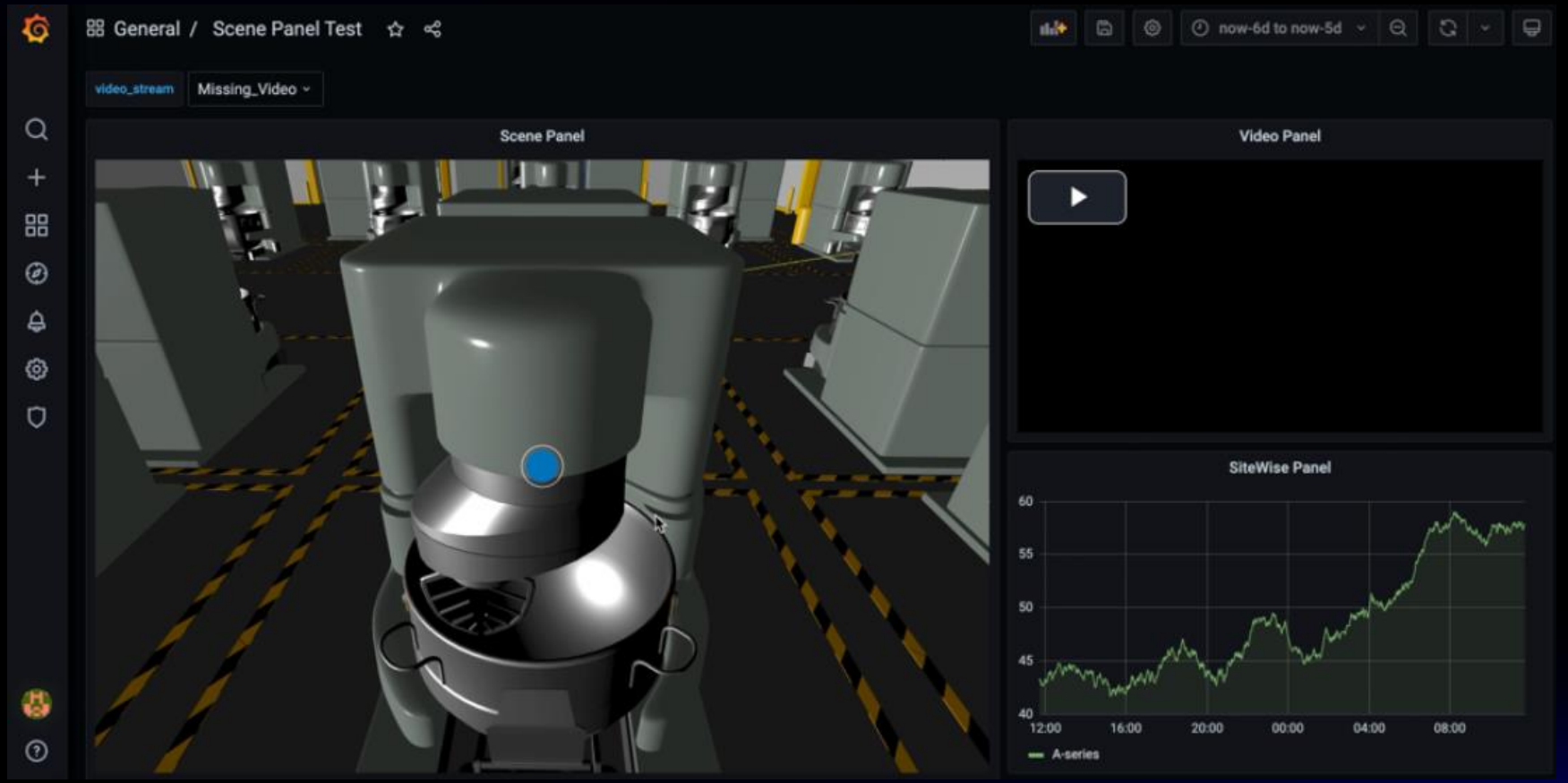
Place assets in a scene

Create bindings for dynamic instantiation, placement

Add data bound, interactive ‘widgets’

Step 5: Create web applications

Create custom dashboards for your operators



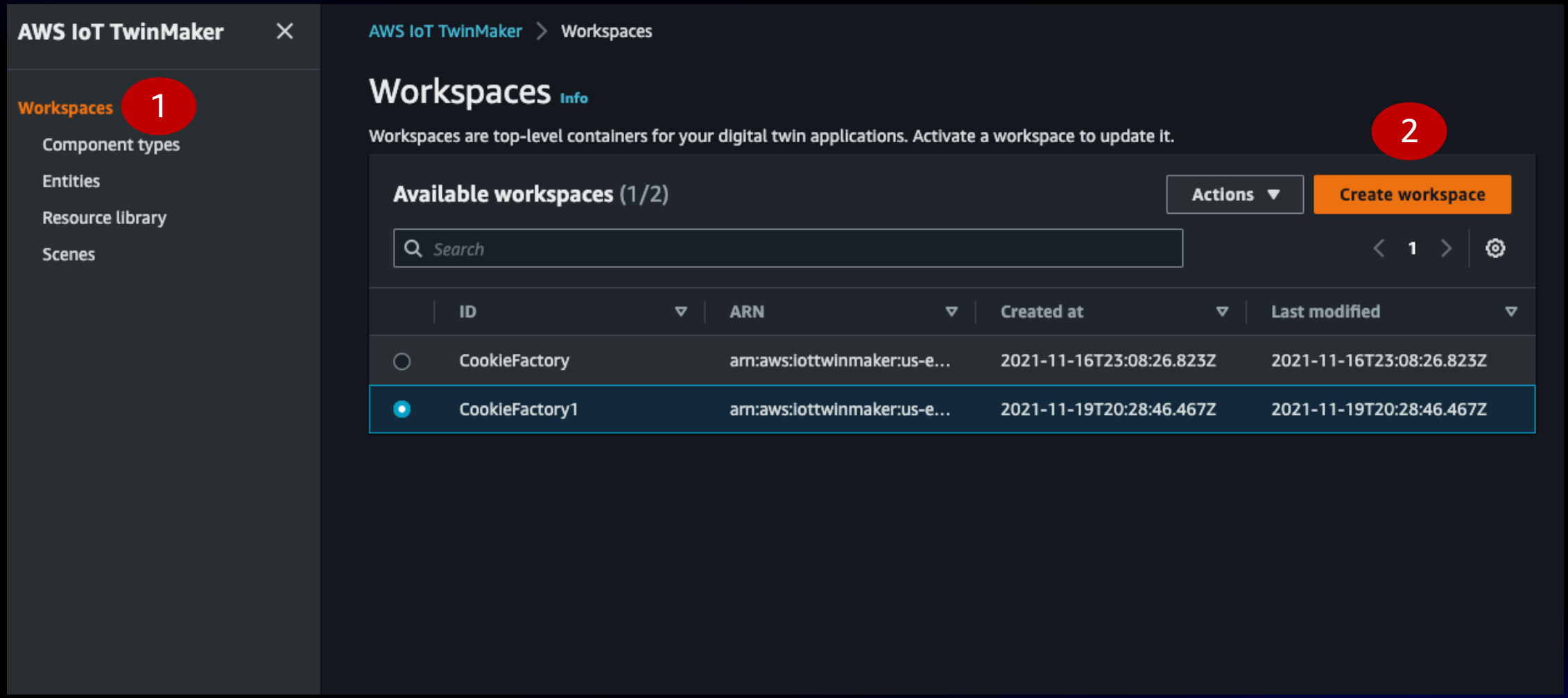
Additional details

- AWS IoT TwinMaker console
- Grafana dashboards
- Getting started sample scripts

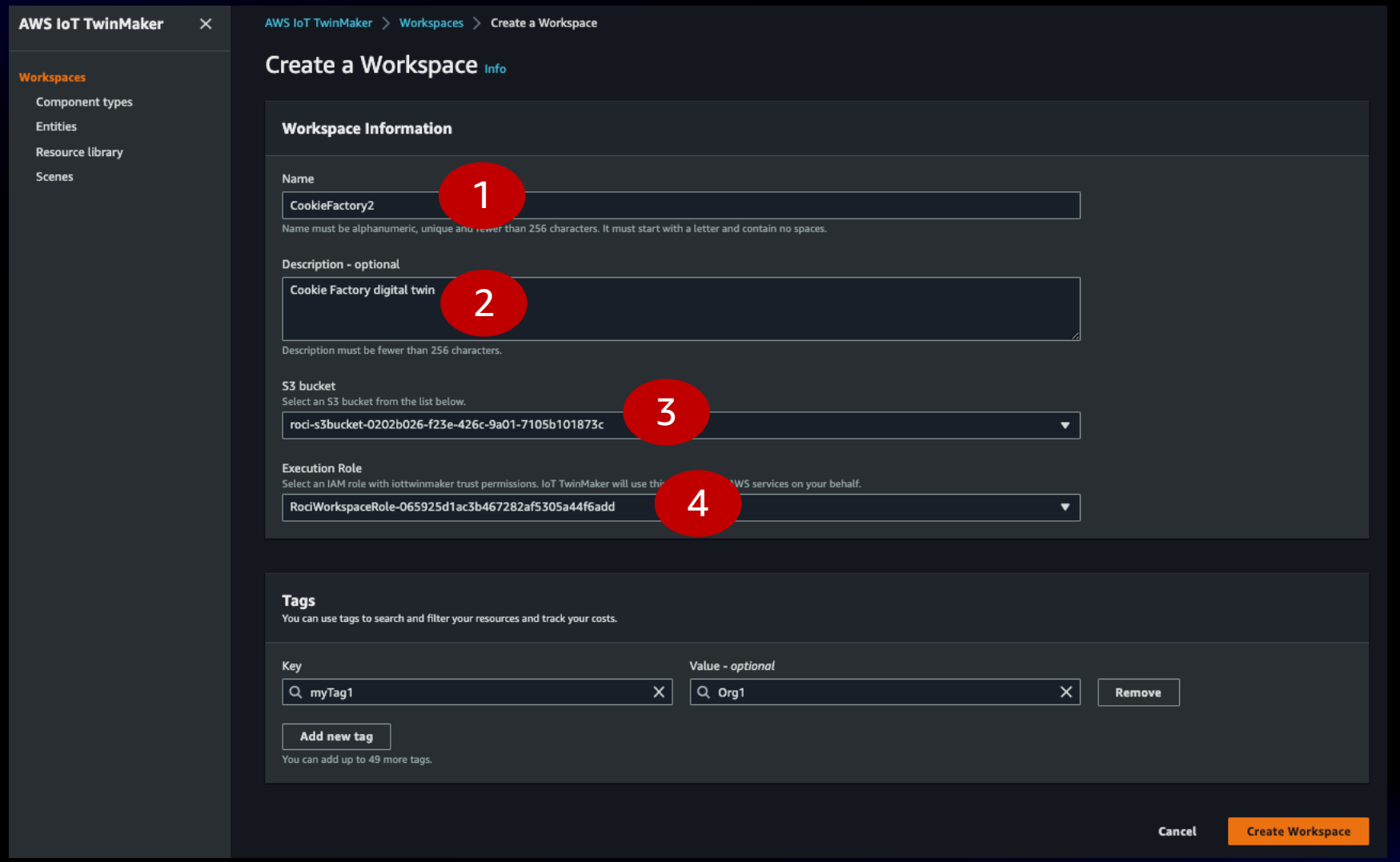
AWS IoT TwinMaker console

List workspaces

Workspace
Modeling
Insights
3D scenes
Web app

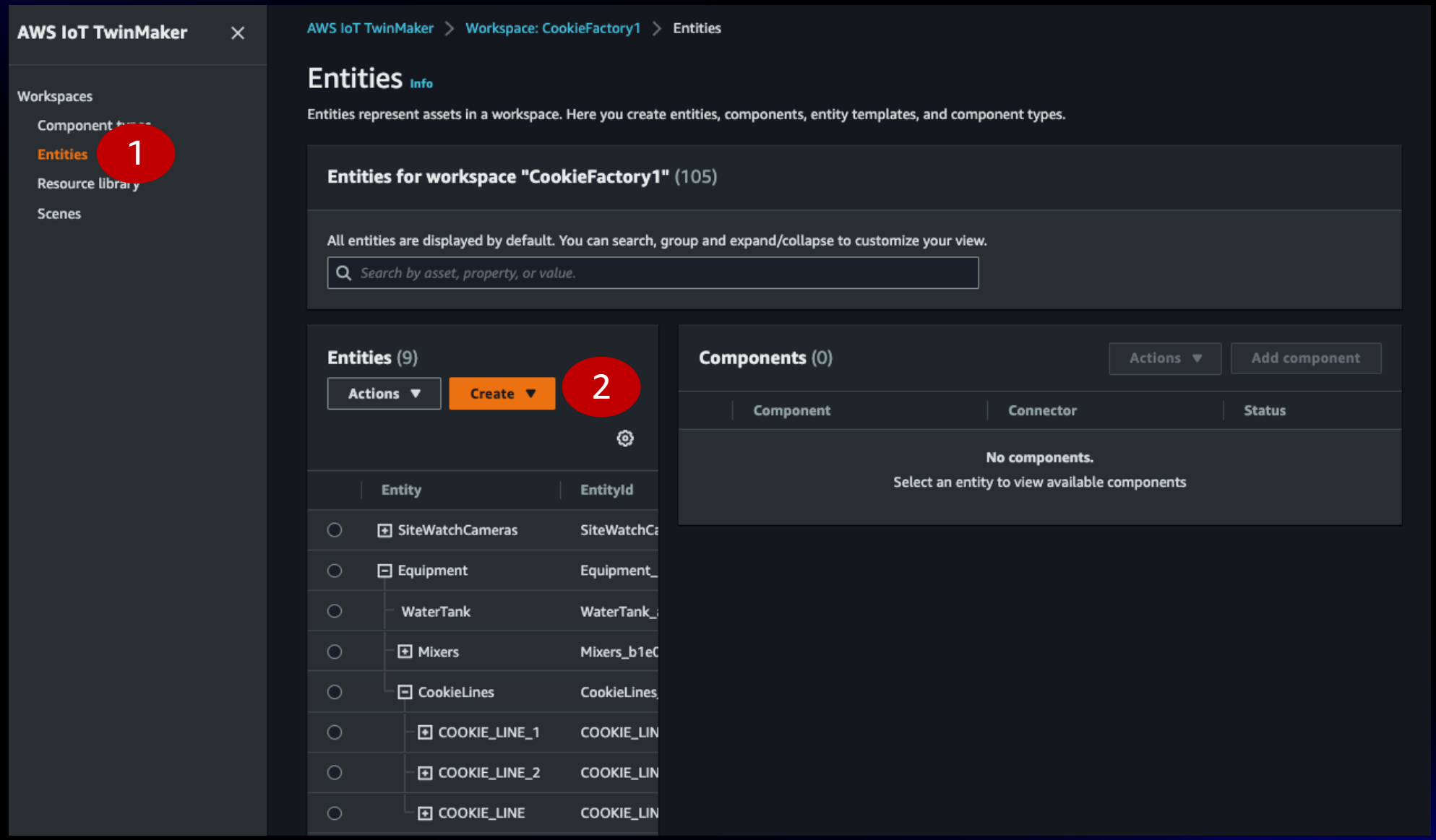


Create a new workspace

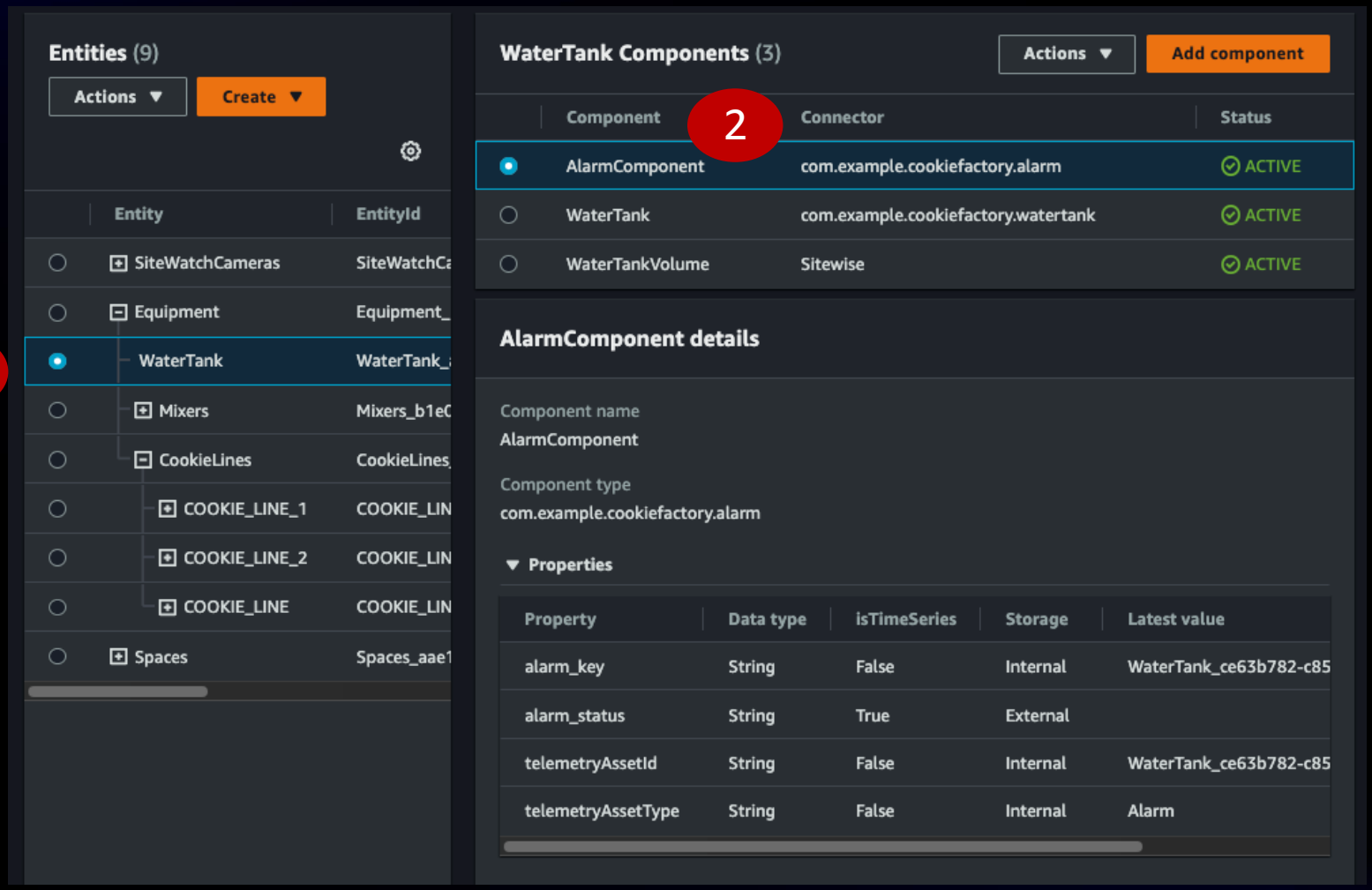


List entities

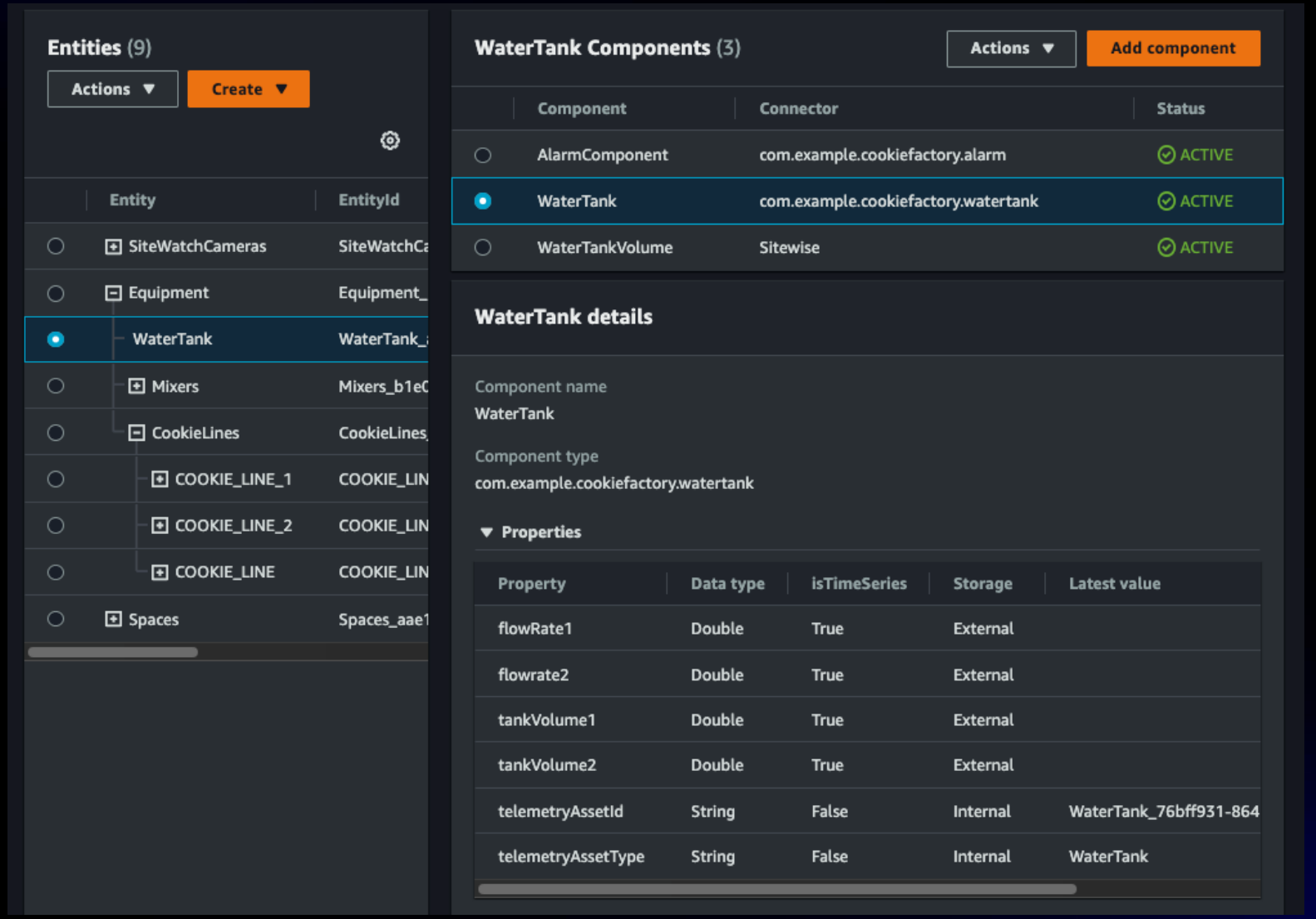
Workspace
Modeling
Insights
3D scenes
Web app



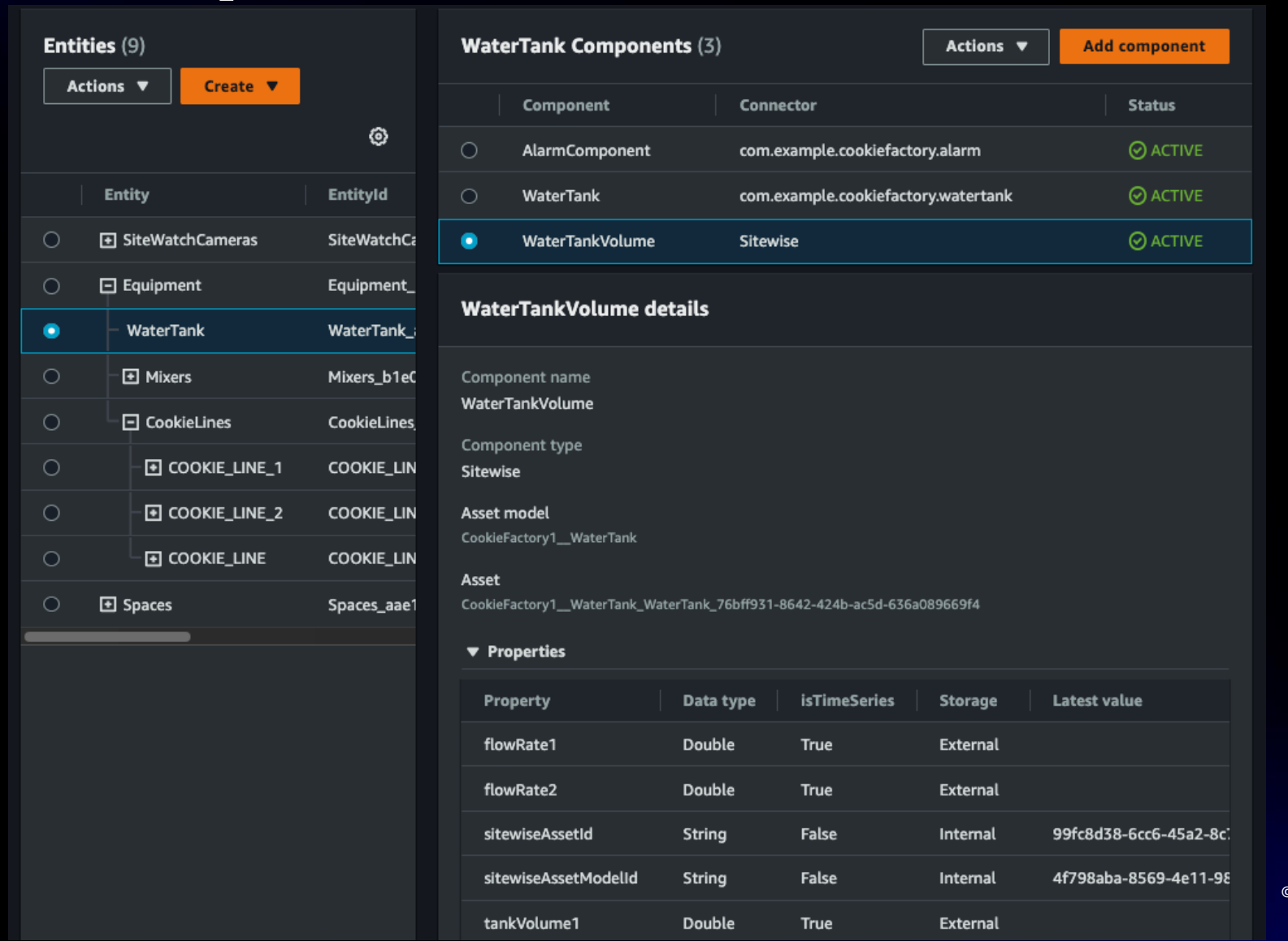
Entity with components (Alarm component)



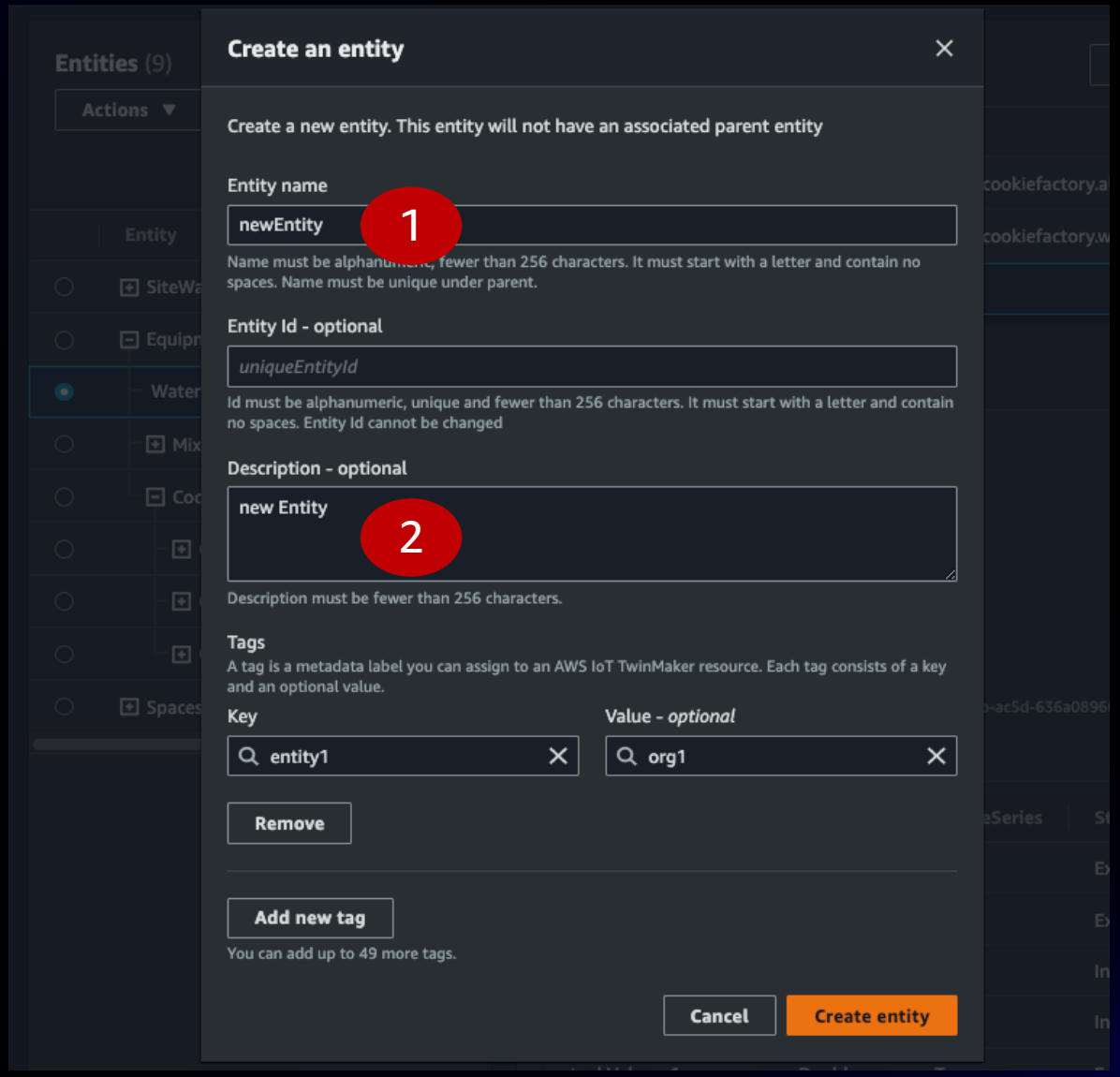
Entity with components (Amazon Timestream component)



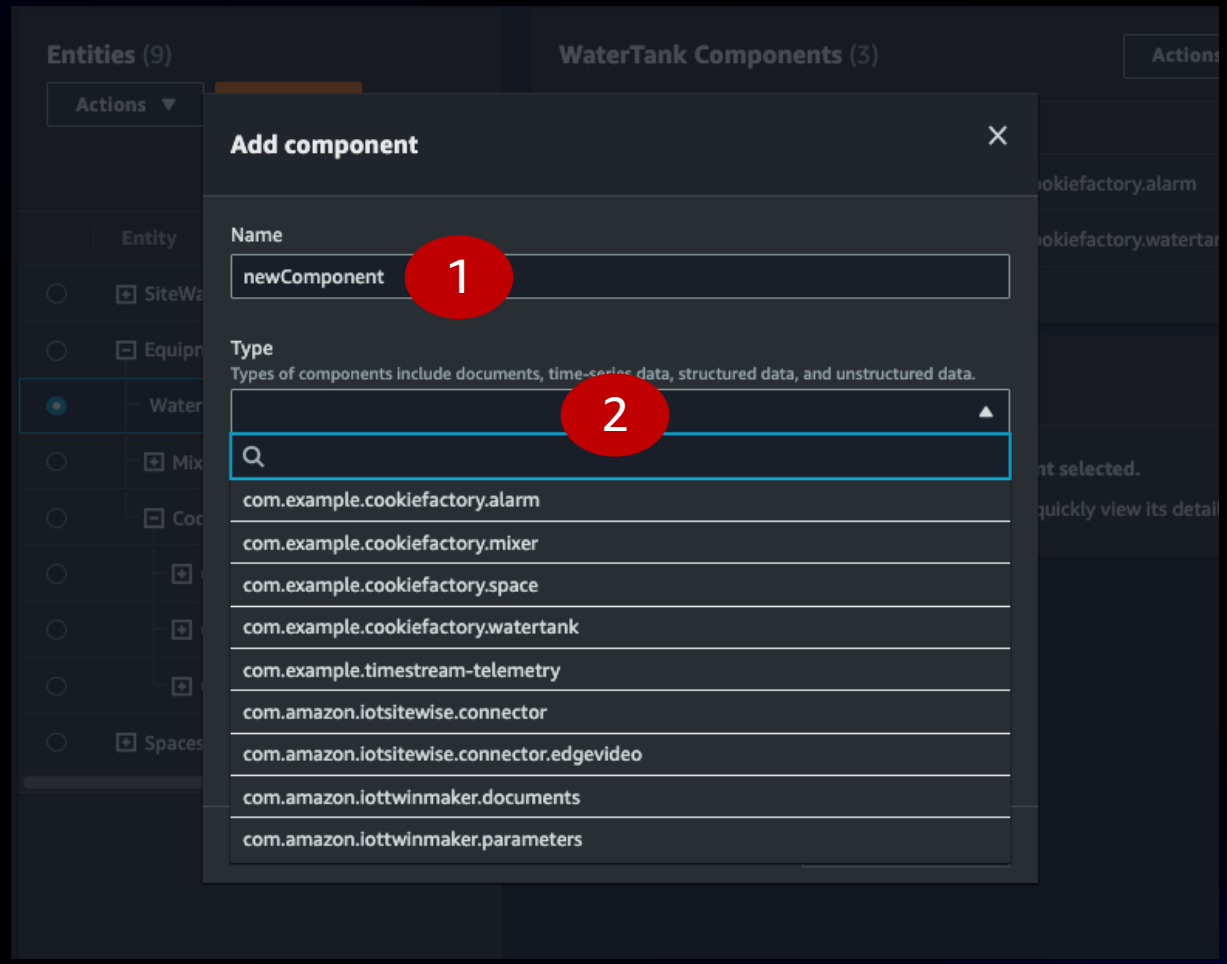
Entity with components (AWS IoT SiteWise component)



Create new entity

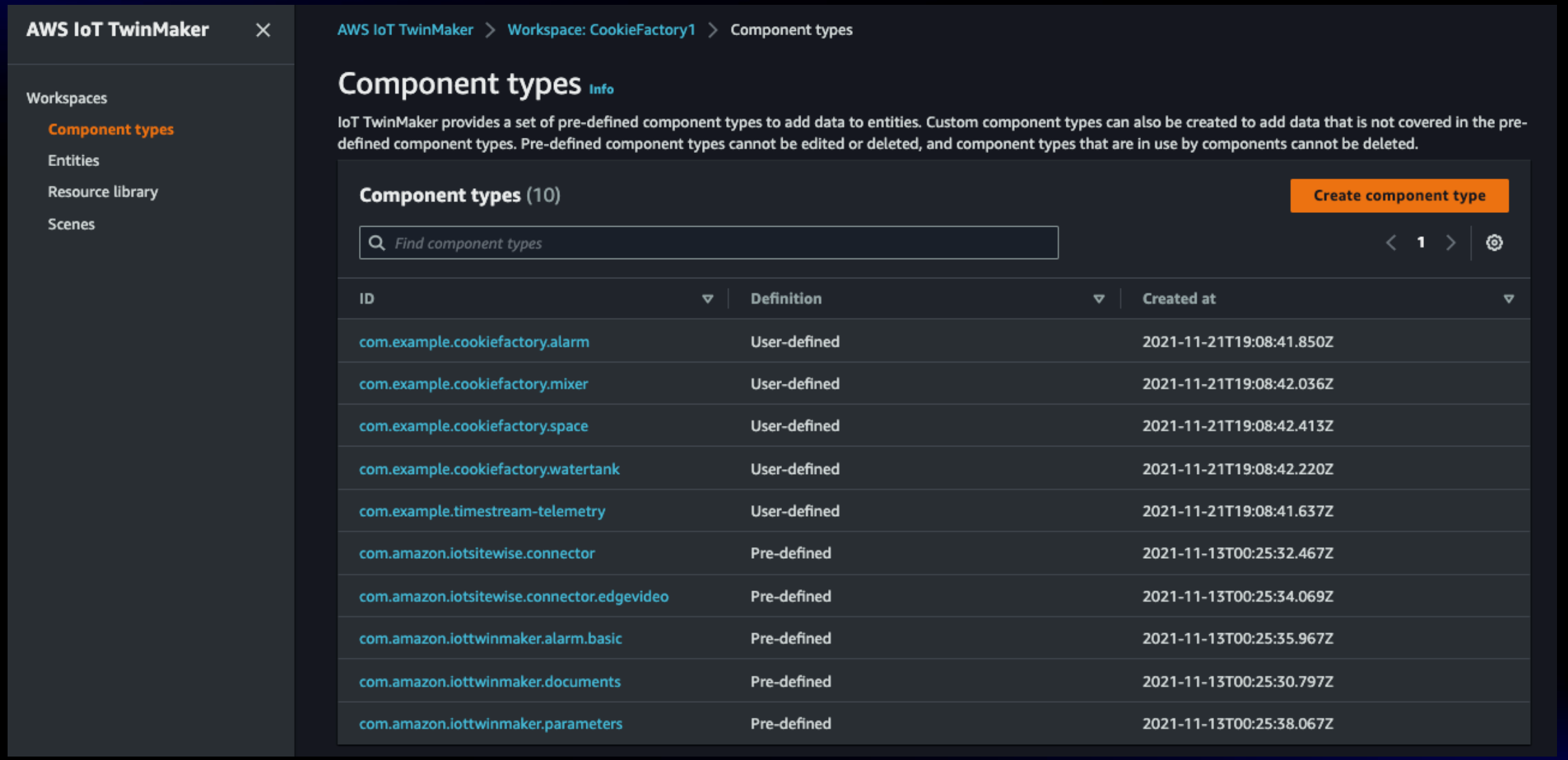


Add component to entity



List components

Workspace
Modeling
Insights
3D scenes
Web app



Create new custom component

Create component type Info

Component type information

Component type ID

com.user.custom1

Description

My custom connector

Must be less than 2048 characters

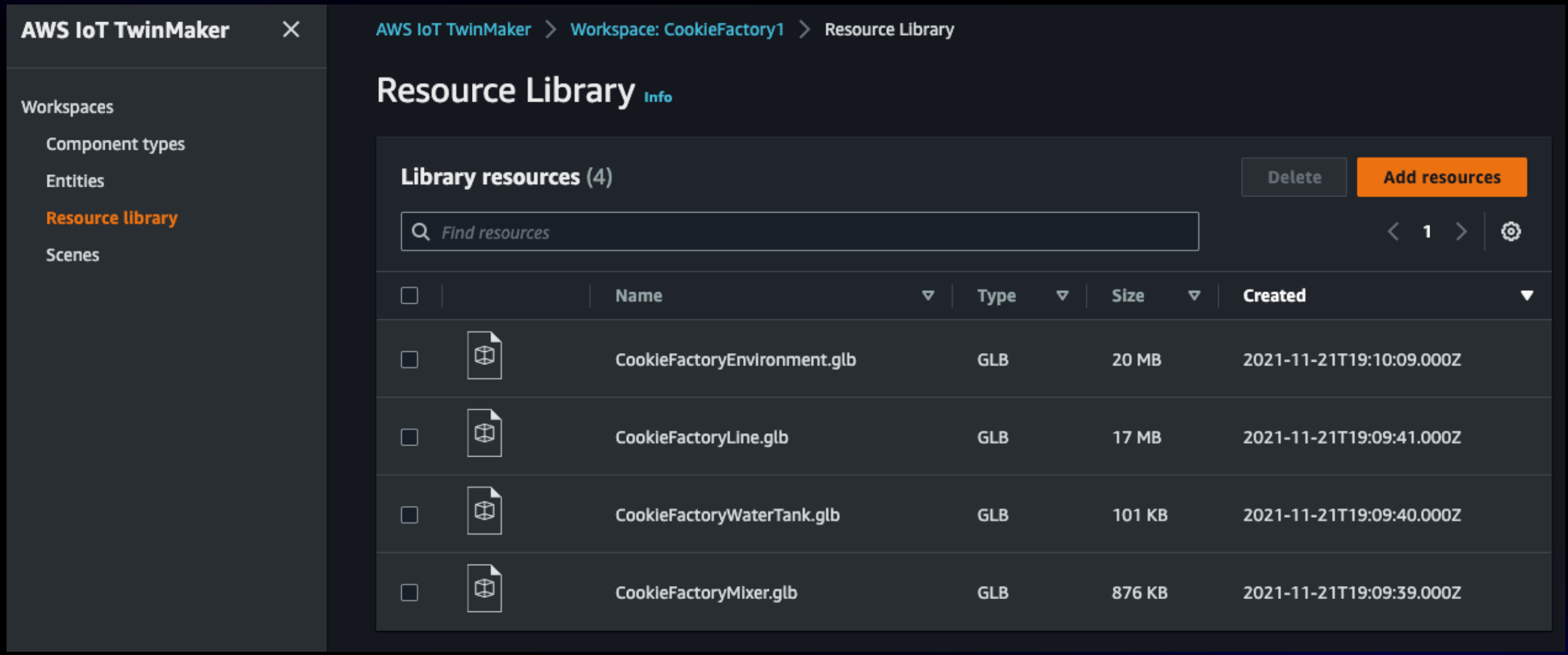
Request

```
{
  "componentTypeId": "com.user.custom1",
  "description": "My custom connector",
  "functions": {
    "dataReader": {
      "implementedBy": {
        "lambda": {
          "arn": "arn:aws:lamba:us-east-1:123456789012:function:connector"
        }
      }
    }
  },
  "propertyDefinitions": {
    "telemetryAssetType": {
      "dataType": {
        "type": "STRING"
      },
      "isExternalId": false,
      "isRequiredInEntity": true,
      "isStoredExternally": false,
      "isTimeSeries": false
    }
  }
}
```

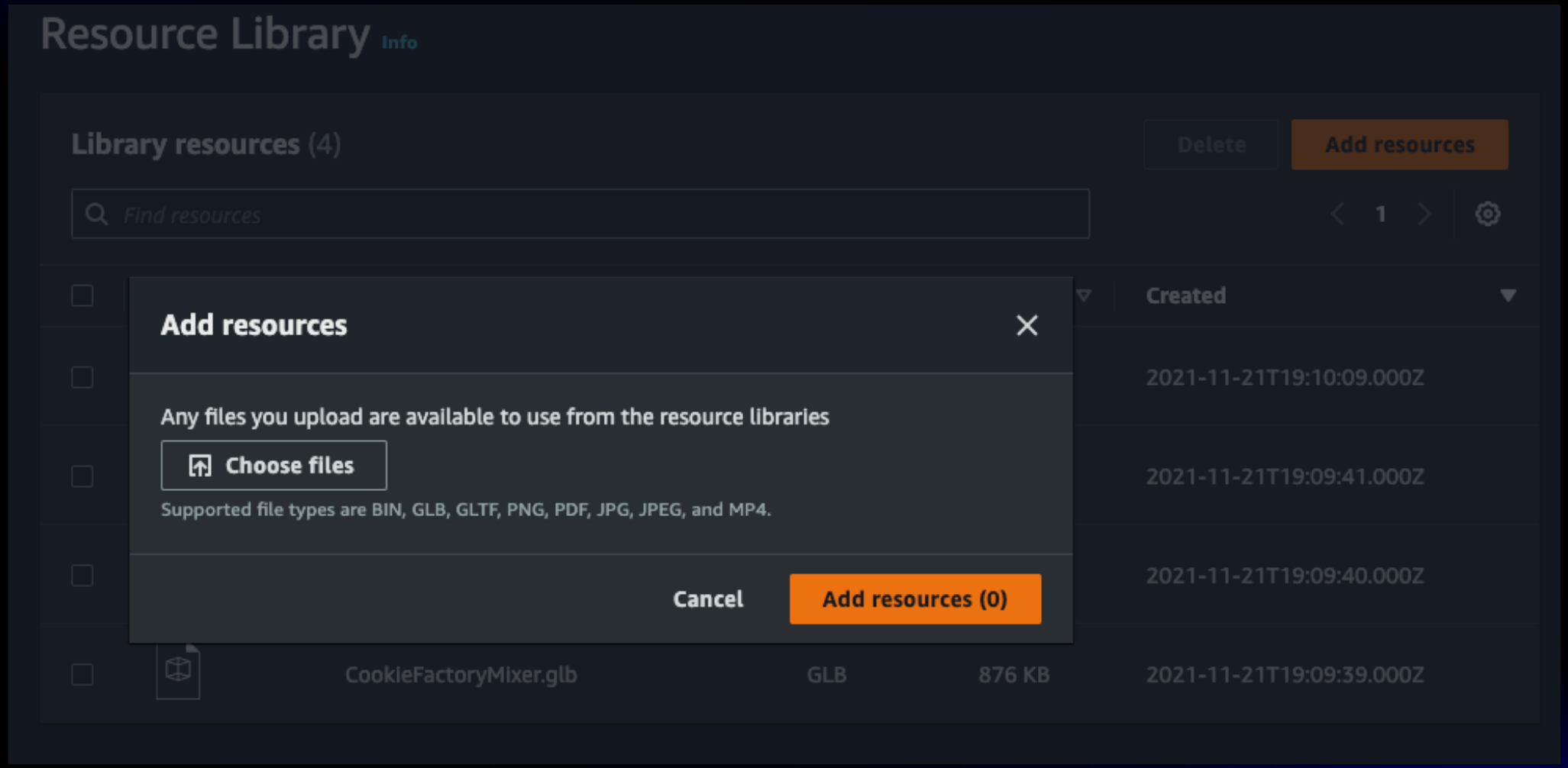
JSON Ln 8, Col 74 Errors: 0 Warnings: 0

List resources

Workspace
Modeling
Insights
3D scenes
Web app

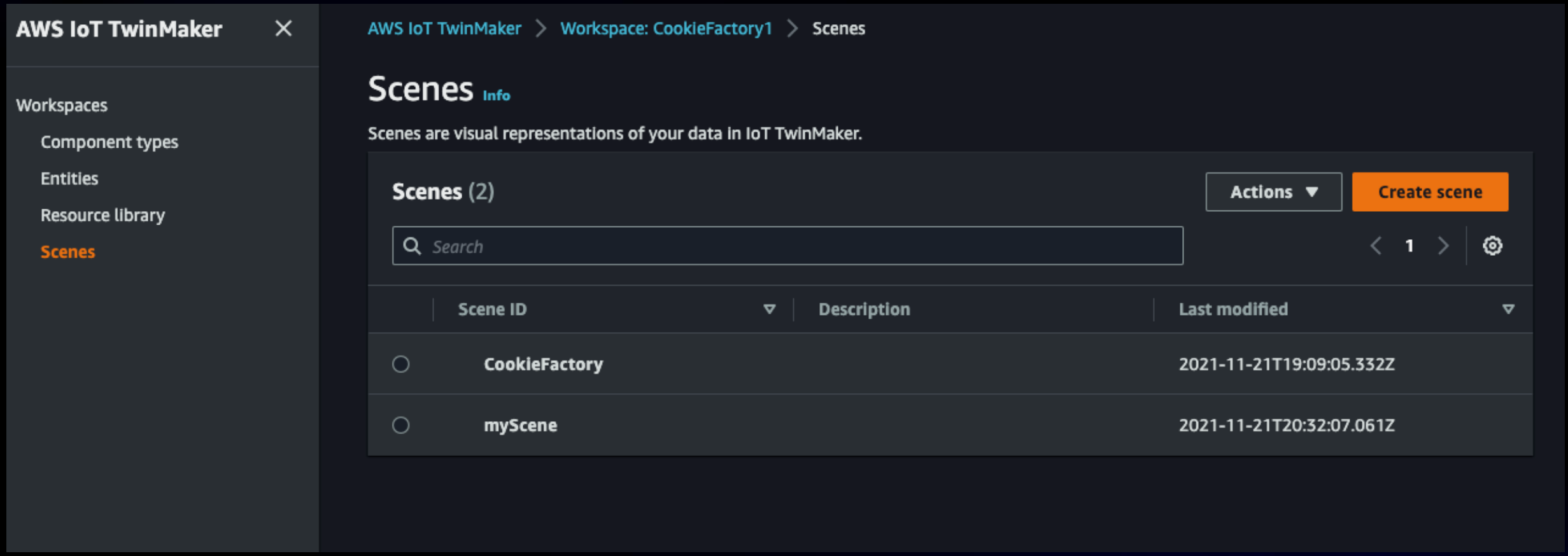


Add resource to library



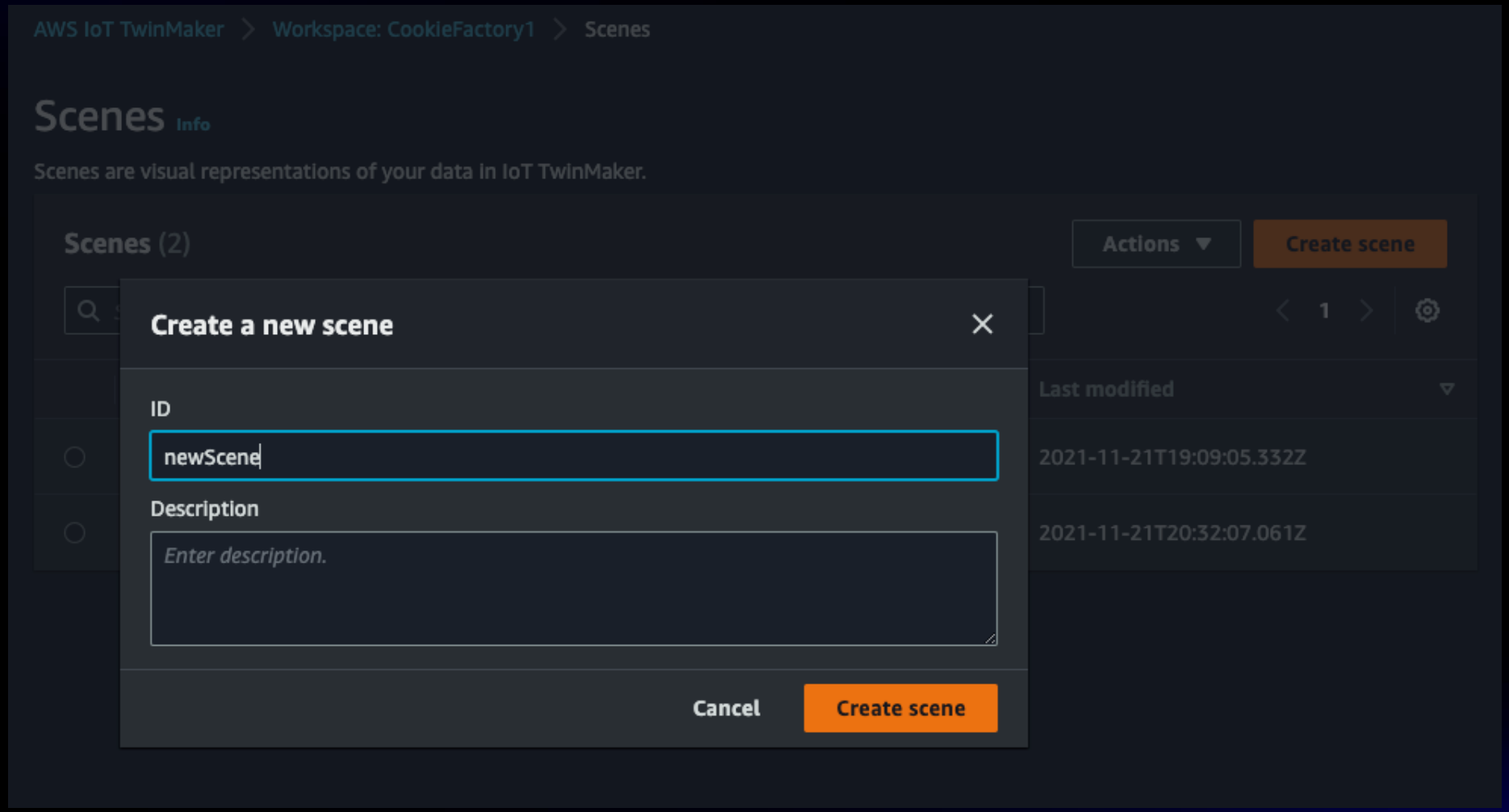
List scenes

Workspace
Modeling
Insights
3D scenes
Web app



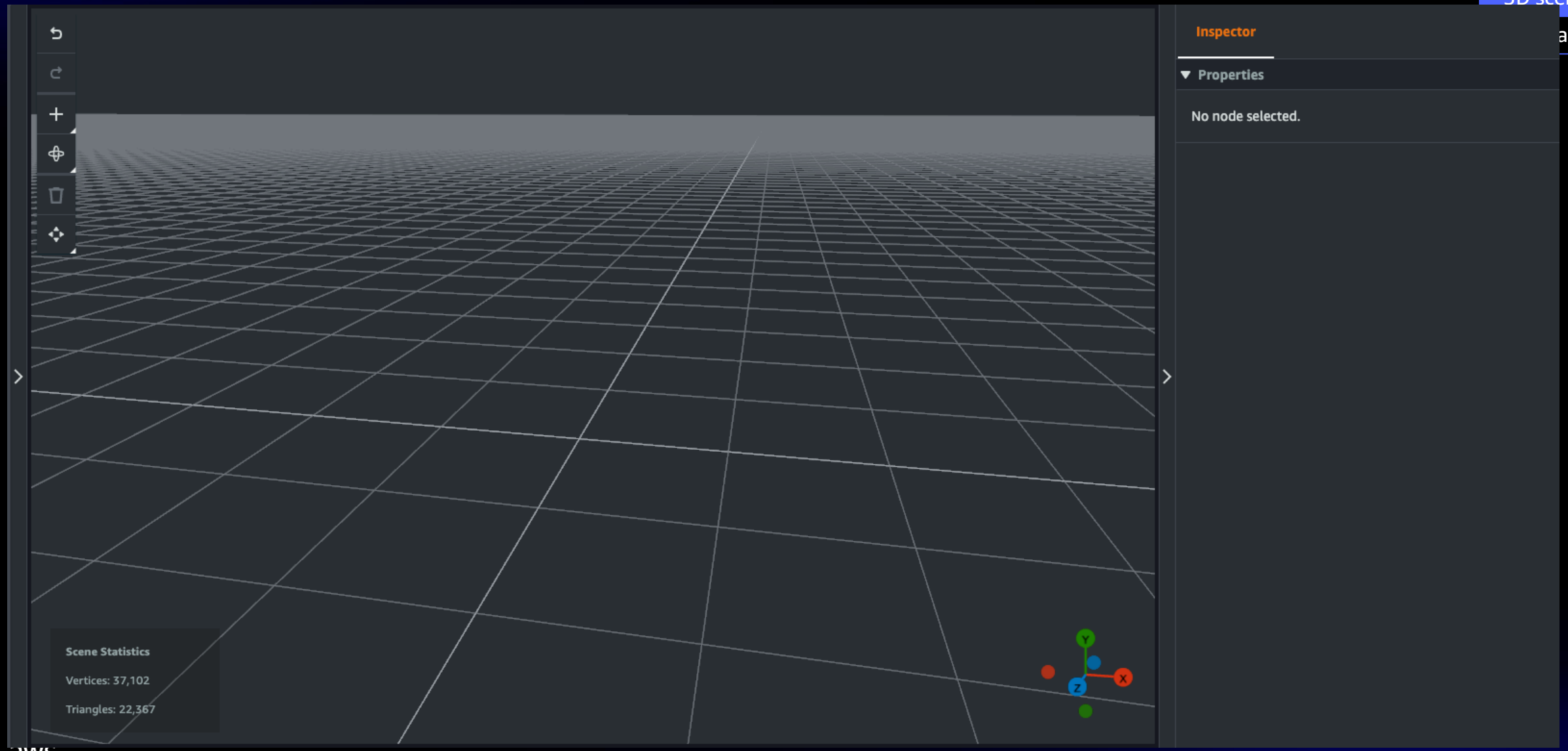
Create new scene

Workspace
Modeling
Insights
3D scenes
Web app

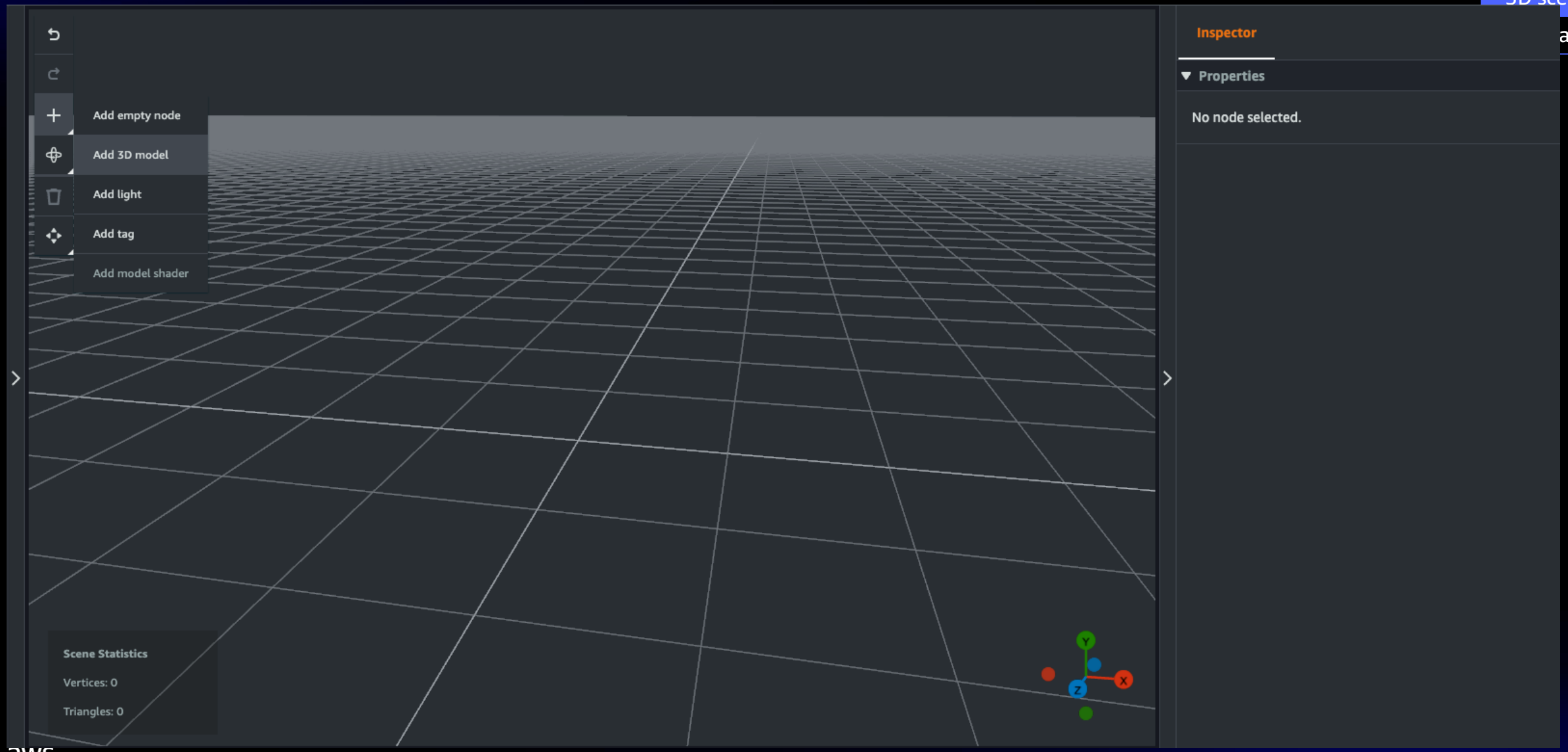


Blank scene

Workspace
Modeling
Insights
3D scenes
app

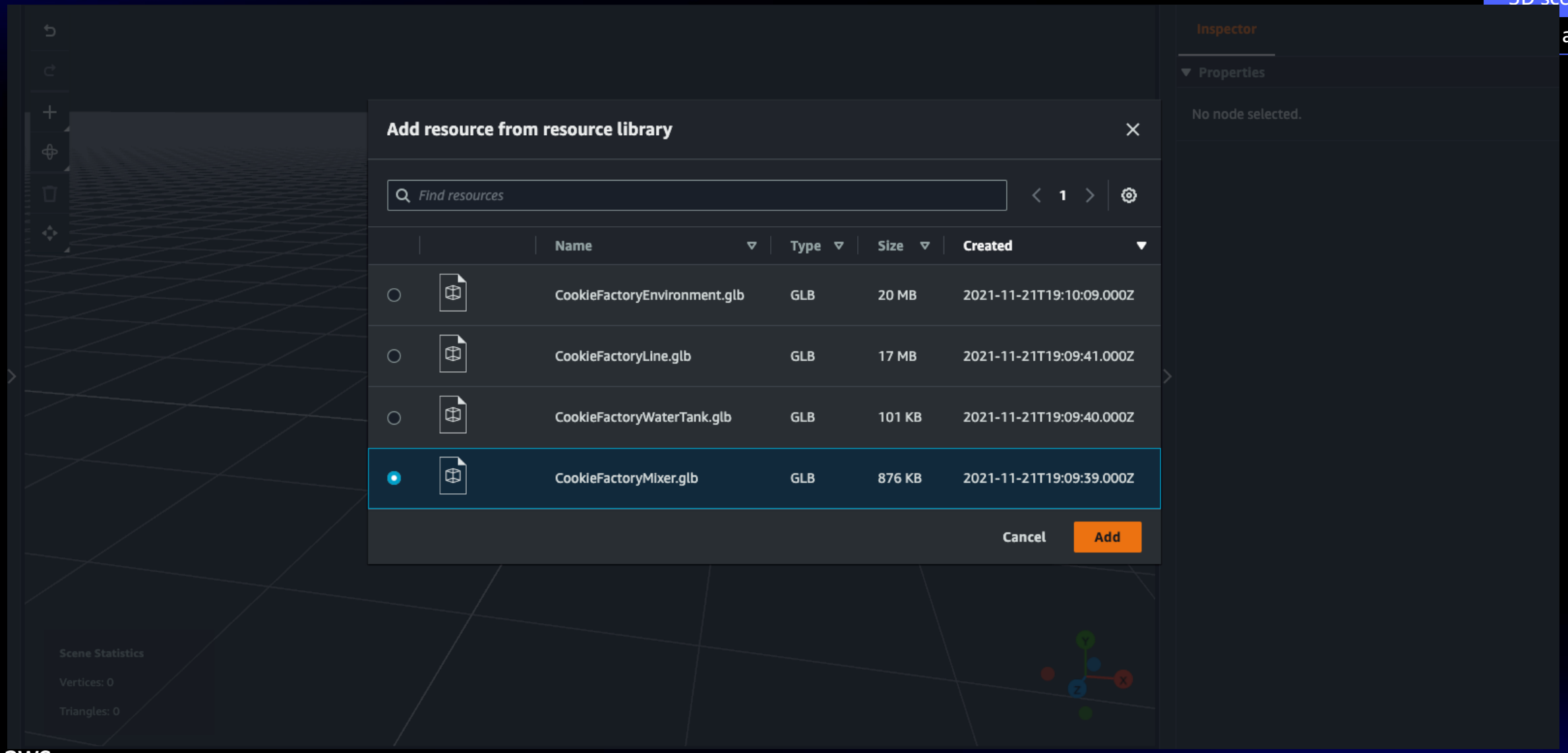


Add elements to scene

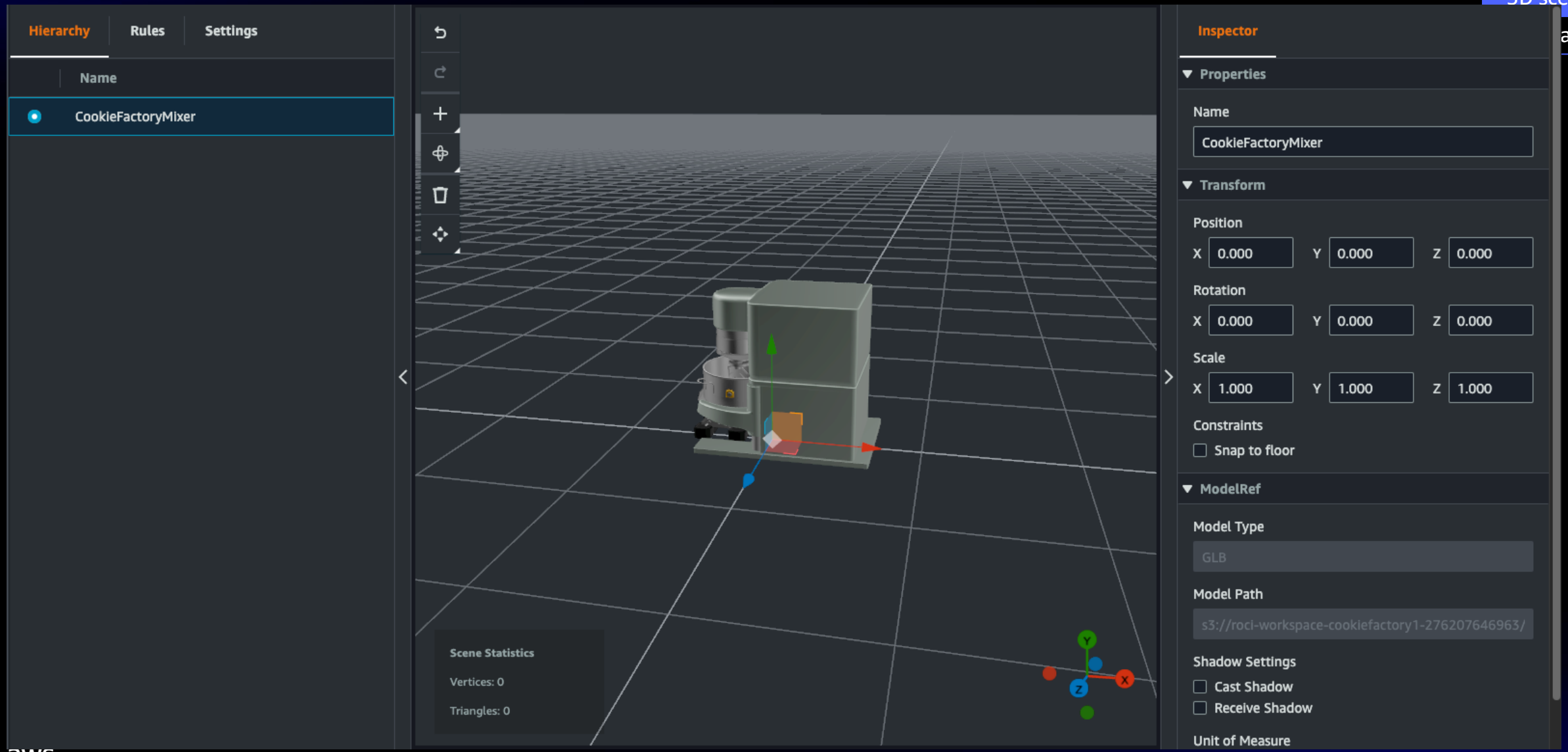


Workspace
Modeling
Insights
3D scenes
app

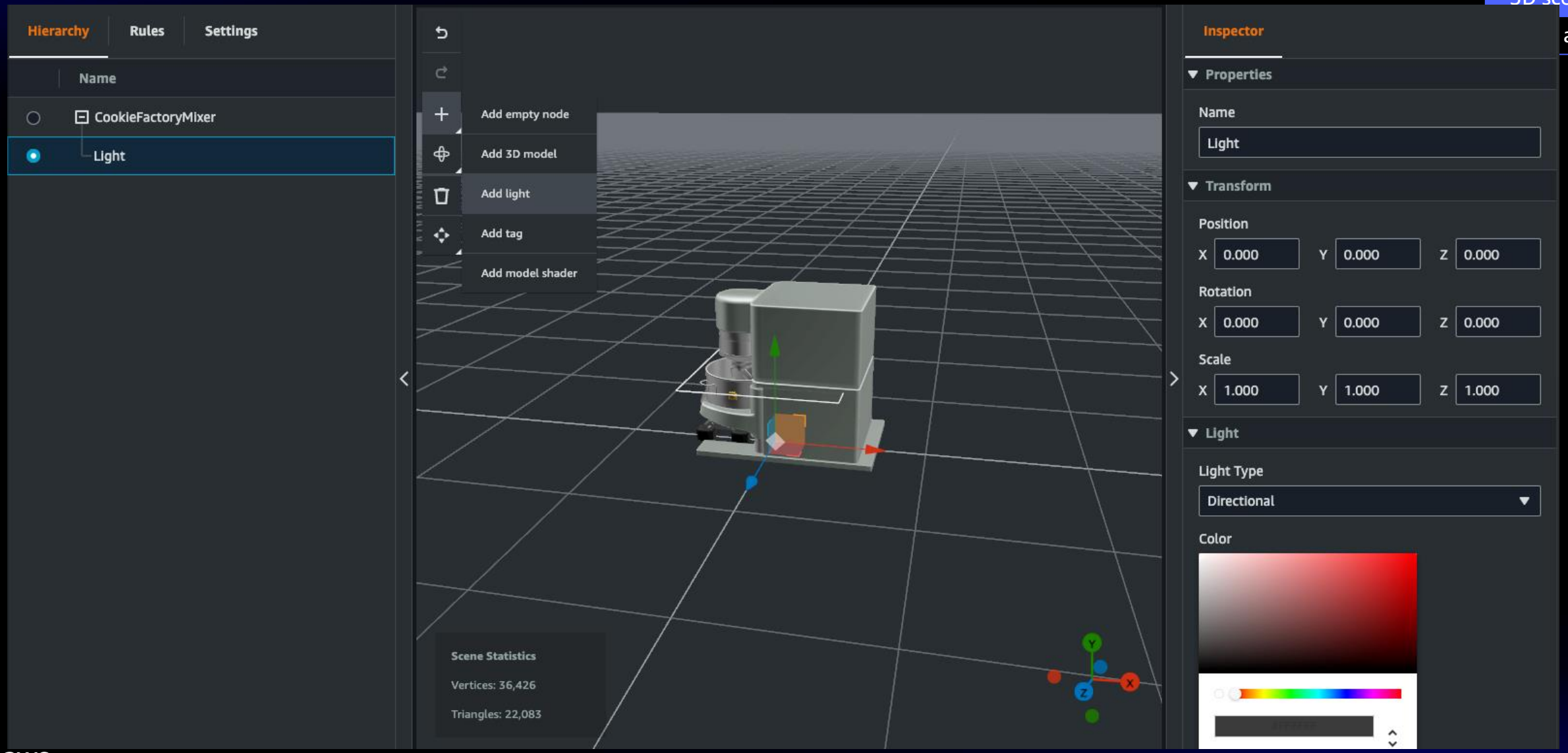
Add 3D model



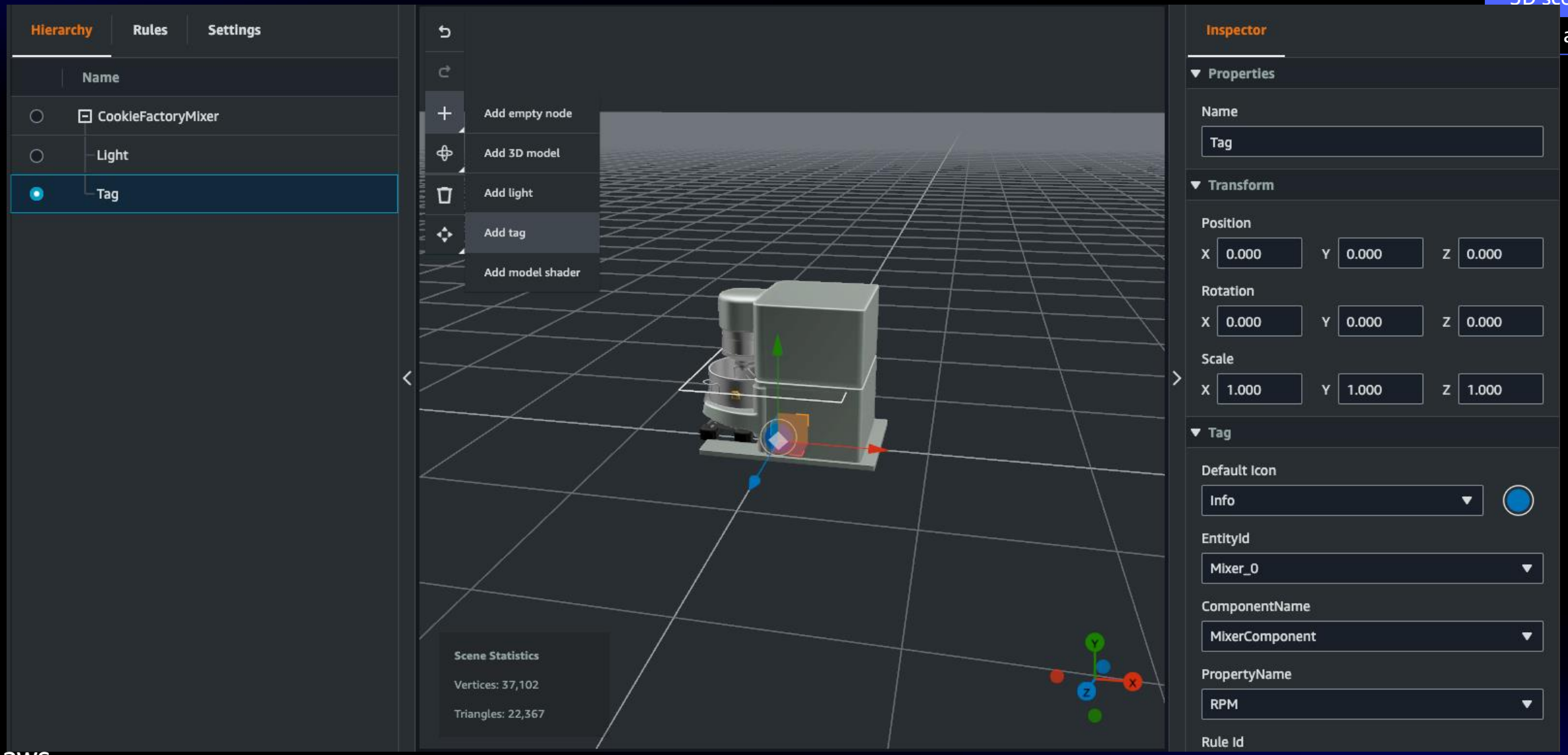
Position 3D model



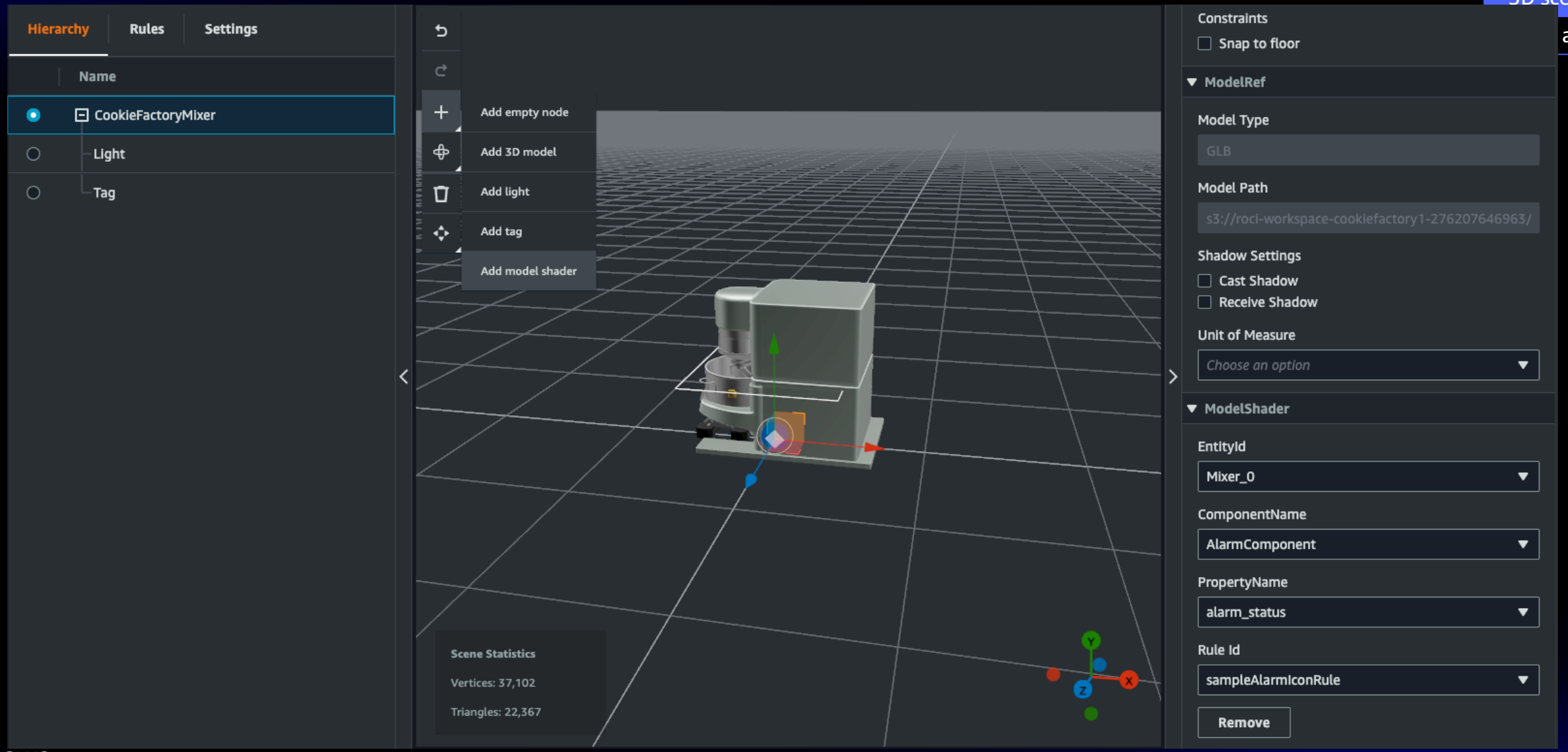
Add lighting to scene



Add tag to entity (for data binding)

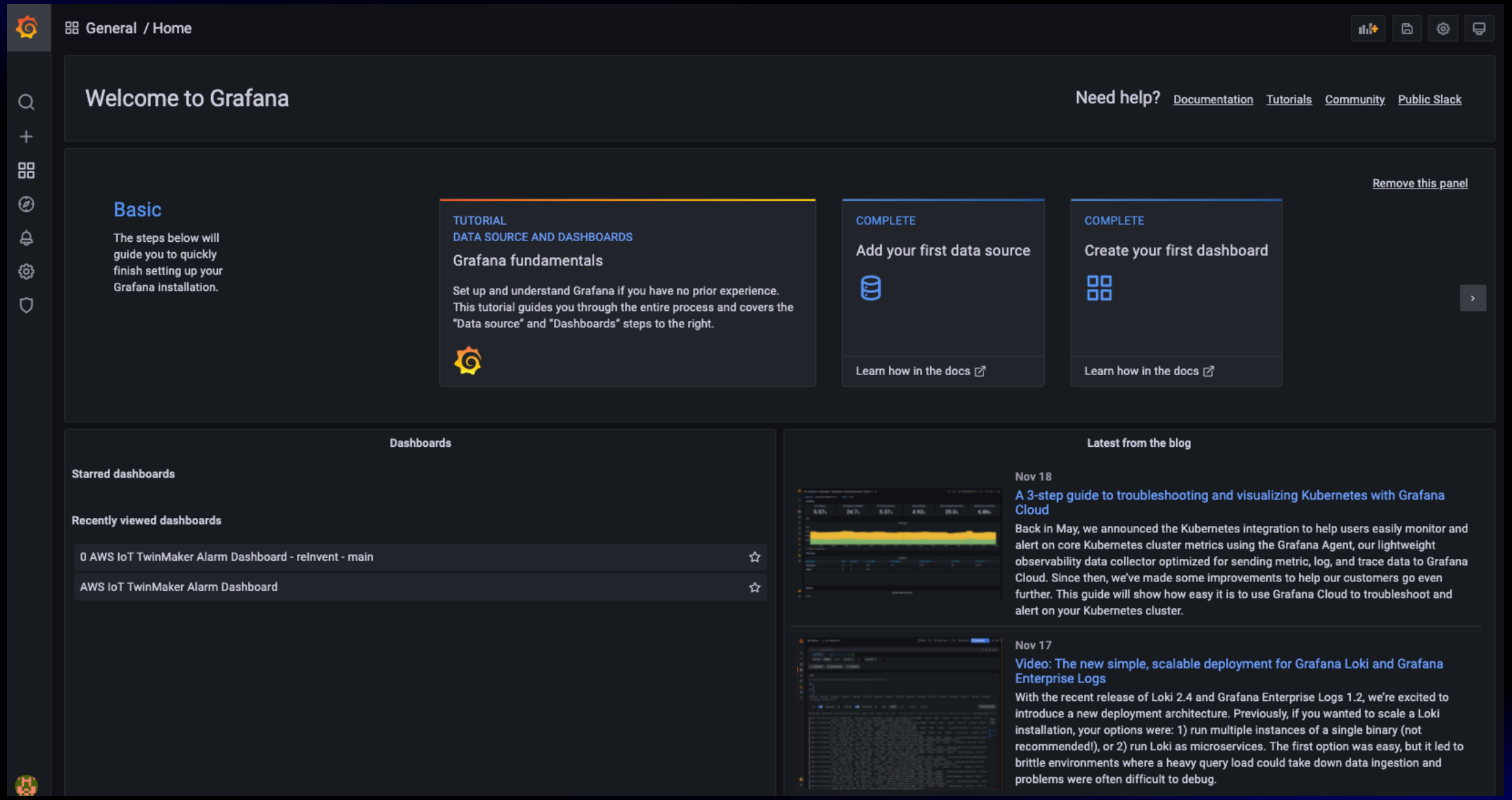


Add model shader (color entity)

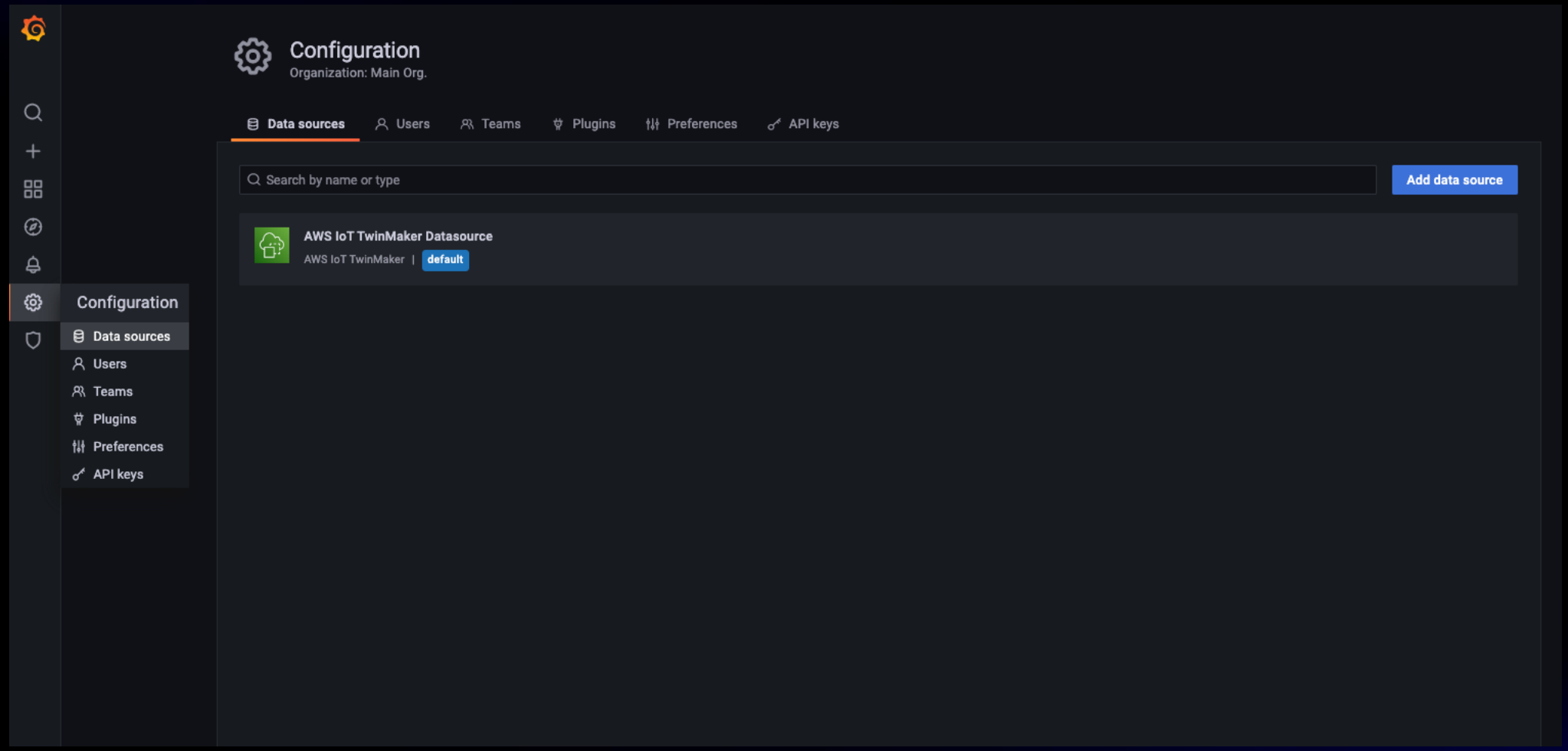


Grafana dashboards

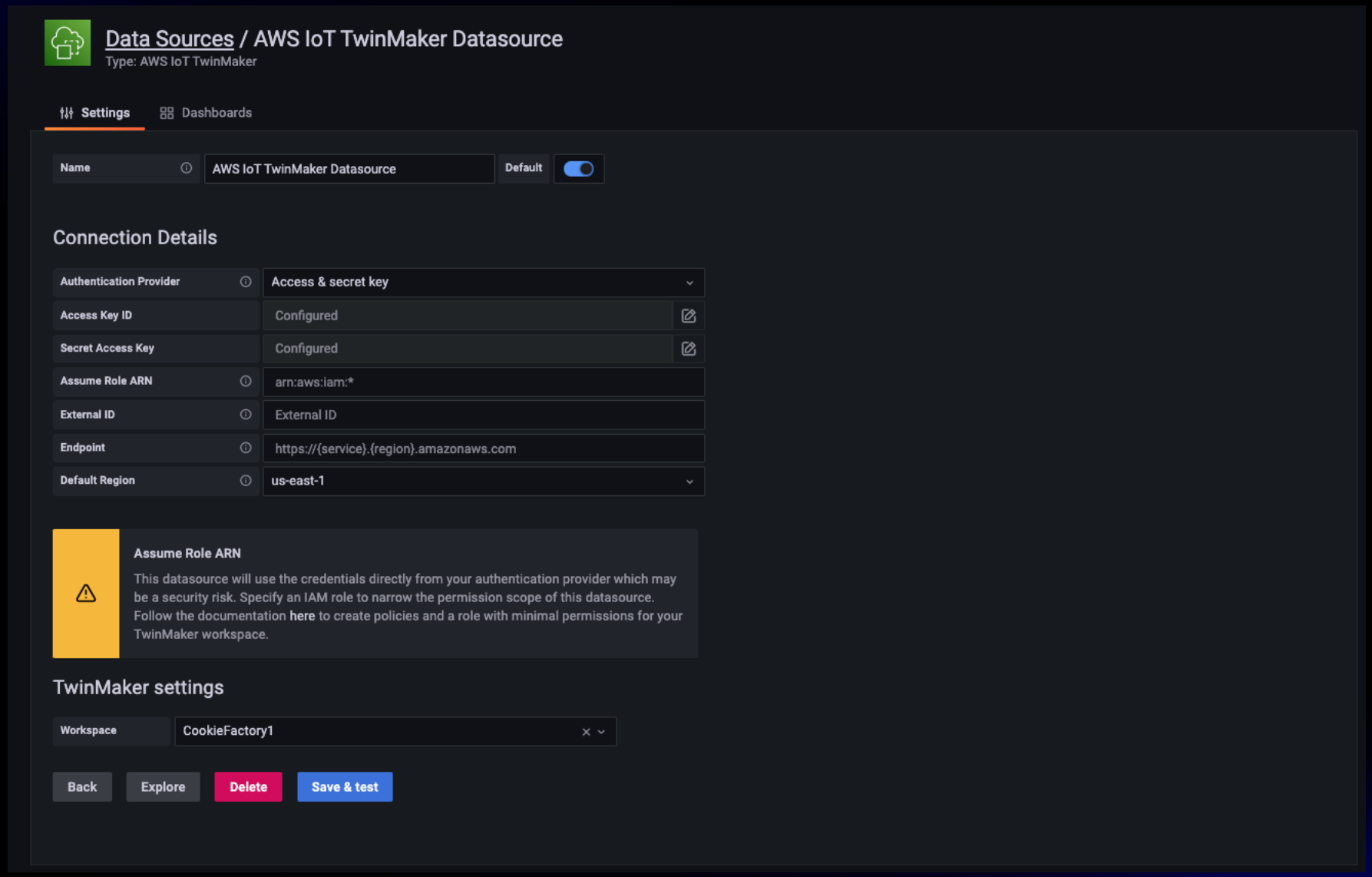
Grafana



Add TwinMaker as datasource

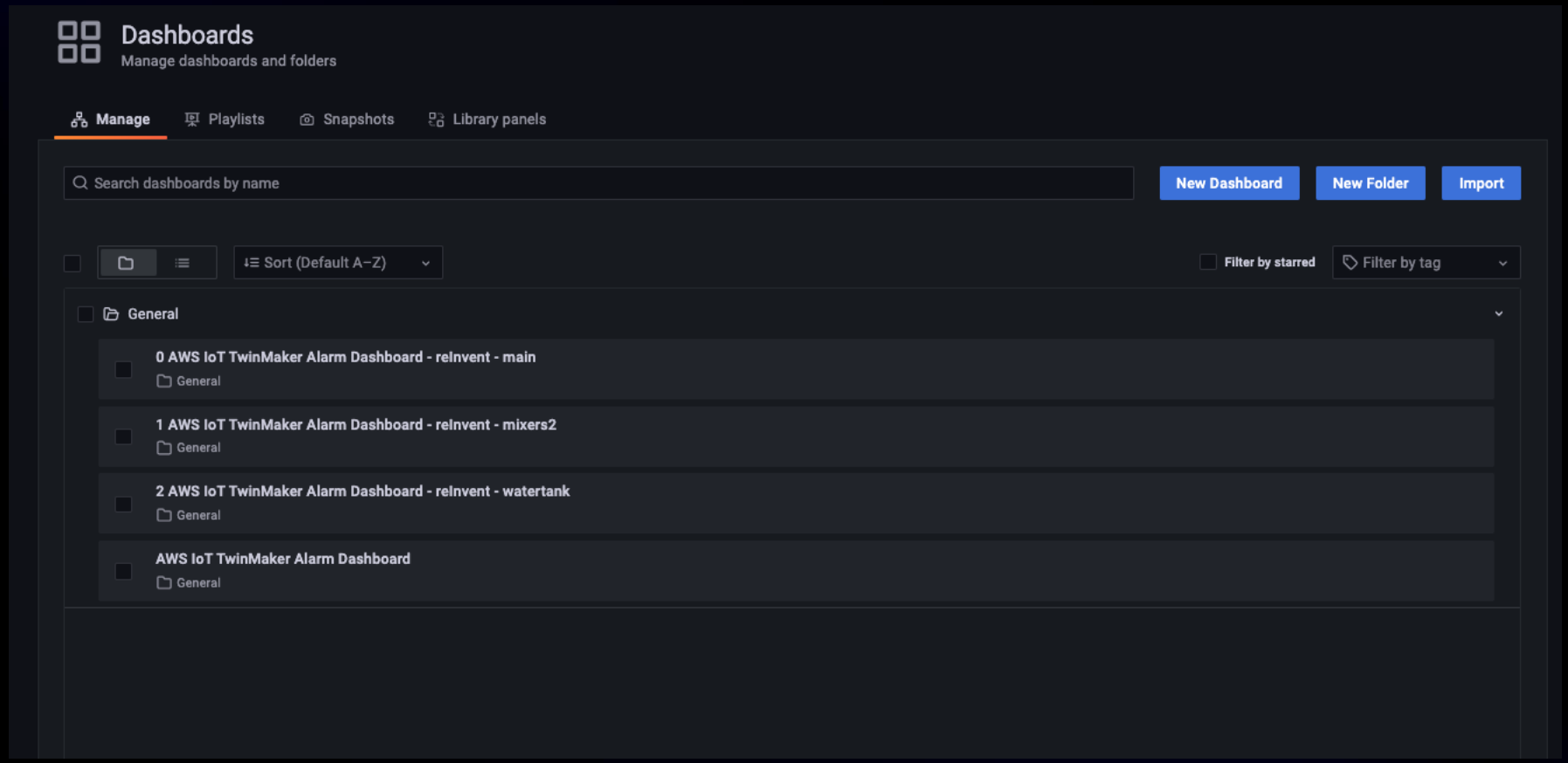


Add TwinMaker as datasource

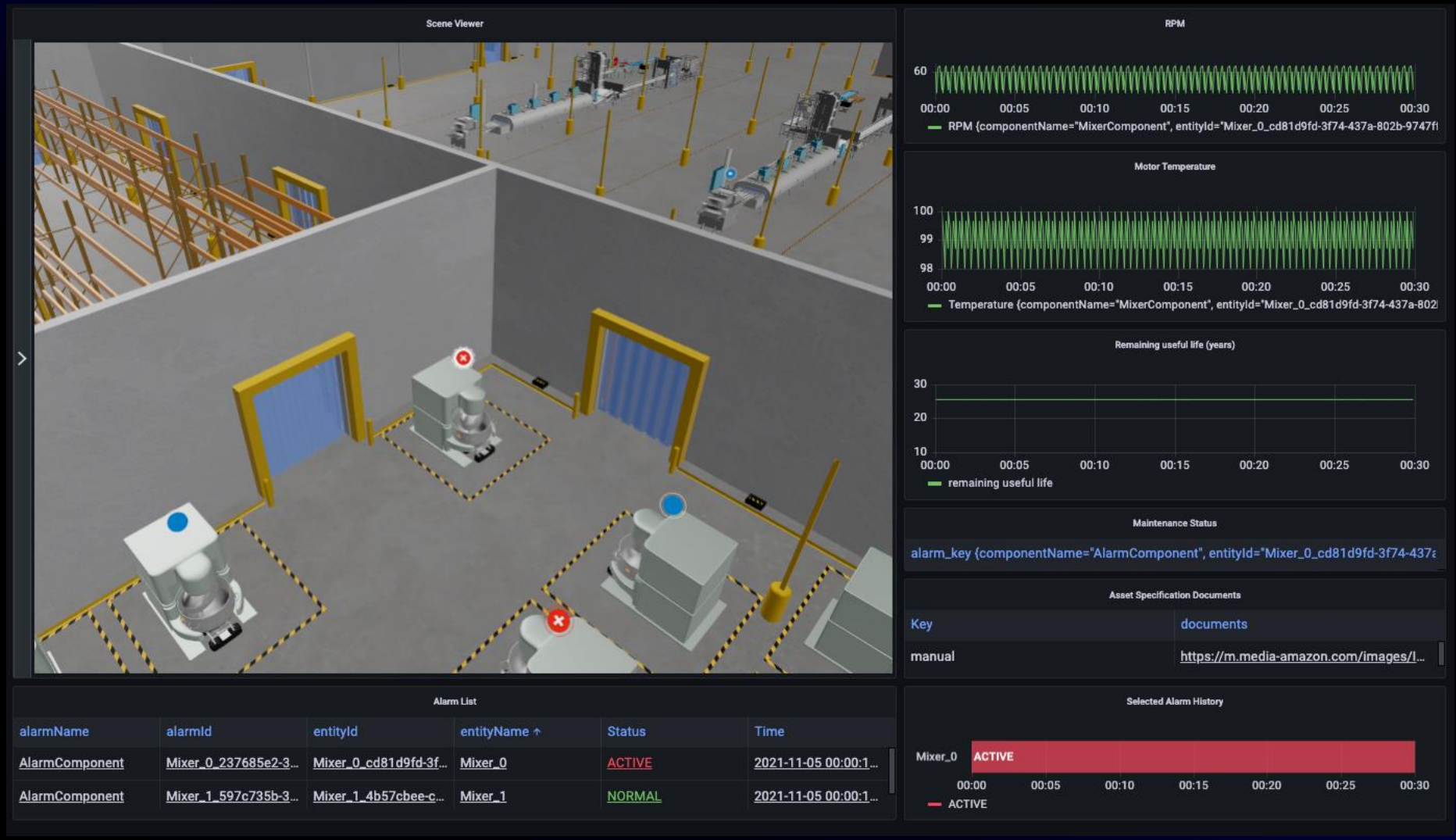


List dashboards

Workspace
Modeling
Insights
3D scenes
Web app

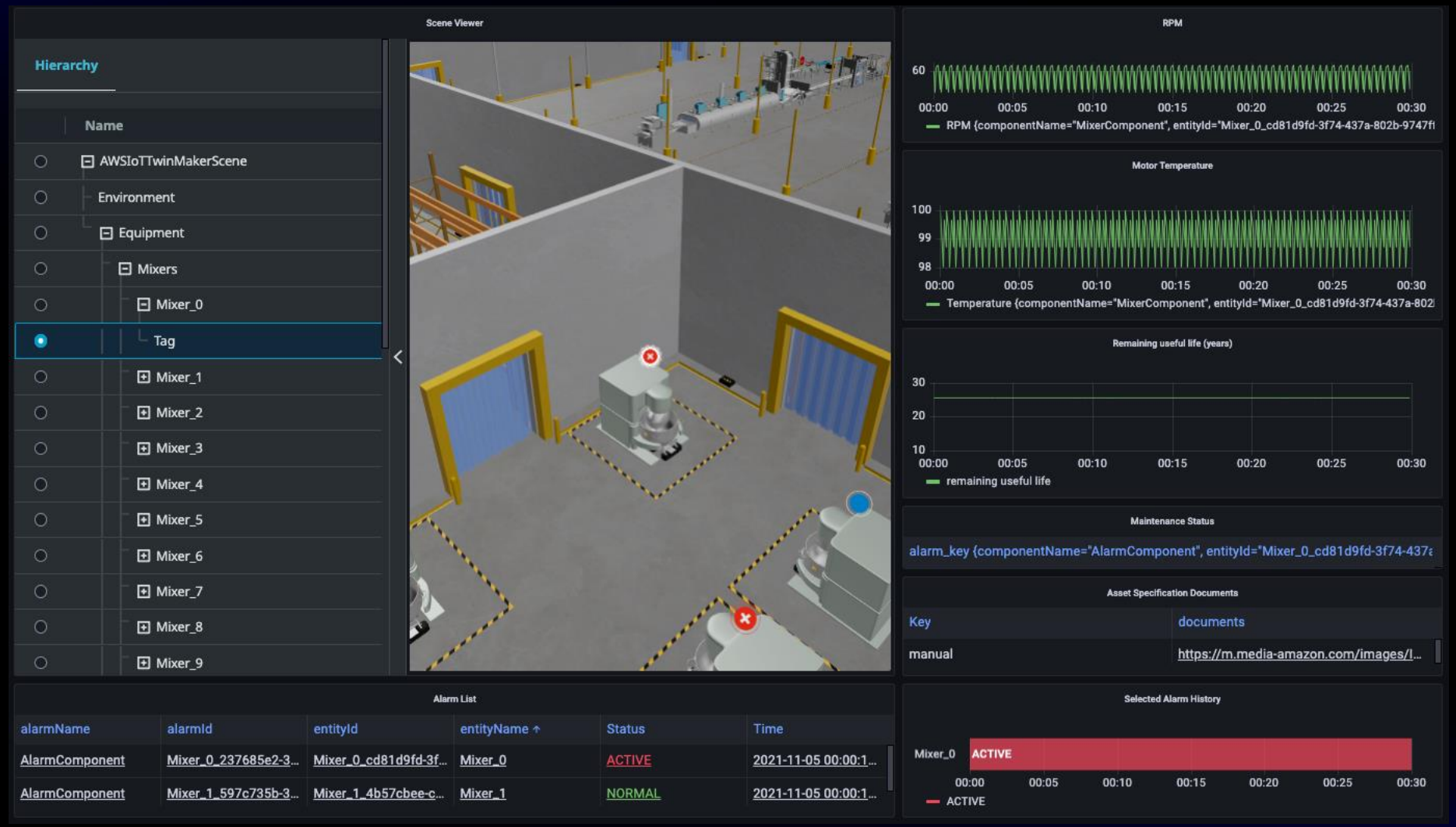


Grafana dashboard (with 3D scene, data, alarms)



Workspace
Modeling
Insights
3D scenes
Web app

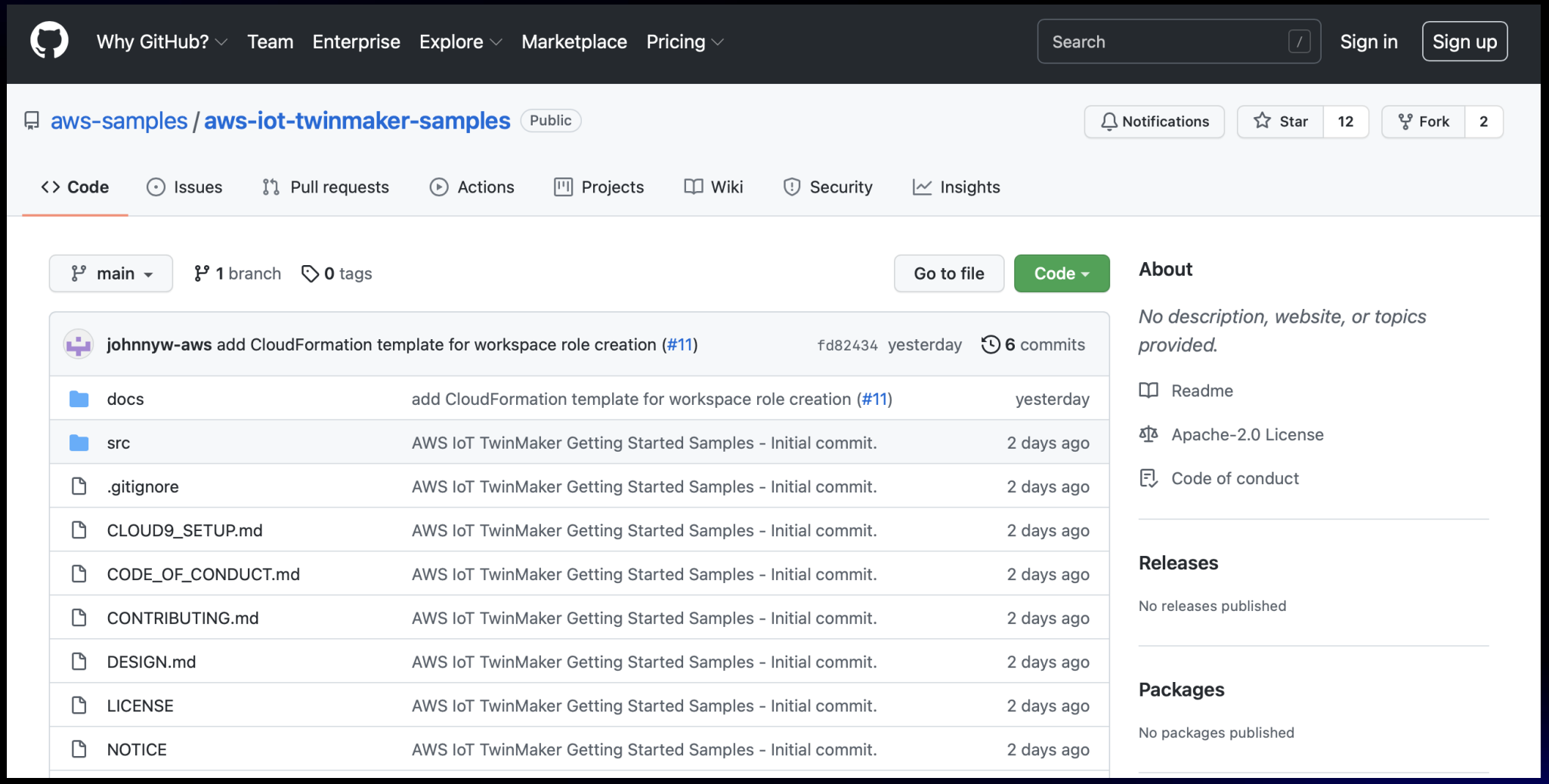
Grafana dashboard (with 3D scene, hierarchy)



Getting started samples

Getting started (1/3)

https://github.com/aws-samples/aws-iot-twinmaker-samples



Getting started (2/3)

- This project walks you through the process of building a Cookie Factory digital twin application using AWS IoT TwinMaker
- Pre-requisites
  - AWS Command Line interface
  - Python 3
  - Node JS
  - AWS CDK (cloud development kit)
  - Docker (to run Grafana locally)

Getting started (3/3)

1. Scripts create the necessary IAM roles
2. You create the workspace in console
3. Scripts create sample data in Amazon Timestream, Amazon Kinesis Video Streams
4. Scripts create all entities needed for Cookie factory and associated components
5. Scripts will load resources for scenes and create the scenes
6. You can start Docker with a pre-built Grafana image, load the TwinMaker plug-in and import the cookie factory dashboard.

You now have a working digital twin of the cookie factory

Thank you!

Raj Devnath
rajdevn@amazon.com

Shyam Kariat
skkariat@amazon.com